



Universidade do Minho
Escola de Engenharia

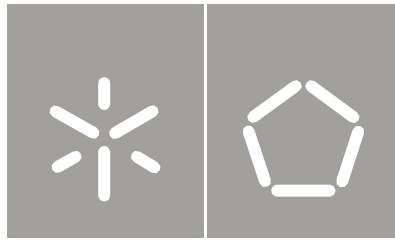
Andreia da Costa Veloso

Virtual Innovation Lab

Andreia da Costa Veloso **Virtual Innovation Lab**

UMinho | 2020

outubro de 2020



Universidade do Minho
Escola de Engenharia

Andreia da Costa Veloso

Virtual Innovation Lab

Dissertação de Mestrado
Mestrado em Engenharia e Gestão de
Sistemas de Informação

Trabalho efetuado sob a orientação da
Professora Doutora Isabel Ramos

outubro de 2020

DIREITOS DE AUTOR E CONDIÇÕES DE UTILIZAÇÃO DO TRABALHO POR TERCEIROS

Este é um trabalho académico que pode ser utilizado por terceiros desde que respeitadas as regras e boas práticas internacionalmente aceites, no que concerne aos direitos de autor e direitos conexos.

Assim, o presente trabalho pode ser utilizado nos termos previstos na licença indicada.

Caso o utilizador necessite de permissão para poder fazer um uso do trabalho em condições não previstas no licenciamento indicado, deverá contactar o autor, através do RepositóriUM da Universidade do Minho.

Agradecimentos

A realização deste trabalho foi uma constante aprendizagem para a qual a contribuição de várias pessoas foi importante.

Agradeço à minha orientadora, Professora Isabel Ramos, que me ajudou durante todo este processo, por me acompanhar e me apoiar em todas as fases e por todas as constantes revisões e orientações.

Agradeço também ao Vítor Barros por todo o auxílio ao longo deste processo e pela partilha de ideias.

Não posso deixar de agradecer aos meus amigos e à minha família pela motivação constante e apoio em todos os momentos.

Obrigada.

DECLARAÇÃO DE INTEGRIDADE

Declaro ter atuado com integridade na elaboração do presente trabalho acadêmico e confirmo que não recorri à prática de plágio nem a qualquer forma de utilização indevida ou falsificação de informações ou resultados em nenhuma das etapas conducente à sua elaboração.

Mais declaro que conheço e que respeitei o Código de Conduta Ética da Universidade do Minho.

Resumo

O tema que se propõe estudar é a maneira como as pequenas e médias empresas (PMEs) podem aprender sobre o potencial transformador das tecnologias emergentes, adquirindo uma melhor preparação para a Transformação Digital de produtos/serviços, processos e modelos de negócio.

As PMEs têm maior dificuldade em avaliar o potencial transformador das tecnologias emergentes. Esta dificuldade advém da complexidade destas tecnologias, as quais estão frequentemente em estado muito imaturo de desenvolvimento. Assim a experimentação com estas tecnologias num ambiente seguro torna-se fundamental para uma decisão de investimento bem fundamentada.

O primeiro objetivo deste trabalho é explorar os desafios da transformação digital com os quais os gestores das PMEs se deparam, em particular o desafio de compreender o potencial transformador das tecnologias emergentes. Este trabalho parte do pressuposto que a aprendizagem por simulação é um meio eficaz para atingir esta compreensão. Neste trabalho apresentamos uma visão geral sobre como a simulação tem vindo a ser aplicada para promover a aprendizagem de processos e tecnologias complexas, nomeadamente em áreas como enfermagem e aviação. A experiência relatada nestas áreas é importante para informar a utilização da simulação na área dos Sistemas de Informação bem como fundamentar as decisões deste trabalho.

Assim, o nosso trabalho foca três áreas centrais: (1) a implementação de estratégias de transformação digital nas PMEs, (2) as tecnologias emergentes com potencial transformador dos modelos de negócio das PMEs, e (3) a aprendizagem baseada na simulação da utilização de tecnologias emergentes.

O nosso trabalho fornece recomendações para o desenvolvimento de competências digitais avançadas nas PMEs e apresenta um ambiente virtual de apoio à transformação digital nestas empresas. As recomendações e ambiente virtual são úteis aos gestores das PMEs e apontam caminhos para futura investigação na área da aprendizagem por simulação no contexto da transformação digital das PMEs.

Palavras-chave: PMEs, simulação, tecnologias emergentes

Abstract

The theme we propose to study is the way in which small and medium-sized companies (SMEs) can learn about the transformative potential of emerging technologies, acquiring a better preparation for the Digital Transformation of products/services, processes and business models.

SMEs find it more difficult to assess the transformative potential of emerging technologies. This difficulty comes from the complexity of these technologies, which are often in a very immature state of development. Thus, experimentation with these technologies in a safe environment becomes essential for a well-founded investment decision.

The first objective of this work is to explore the challenges of digital transformation that SME managers face, in particular the challenge of understanding the transformative potential of emerging technologies. This work assumes that simulation learning is an effective way to achieve this understanding. In this work we present an overview of how simulation has been applied to promote the learning of complex processes and technologies, namely in areas such as nursing and aviation. The experience reported in these areas is important to inform the use of simulation in the area of Information Systems as well as to base the decisions of this work.

Thus, our work focuses on three central areas: (1) the implementation of digital transformation strategies in SMEs, (2) the emerging technologies with the transforming potential of SME business models, and (3) learning based on the simulation of use emerging technologies.

Our work provides recommendations for the development of advanced digital skills in SMEs and presents a virtual environment to support digital transformation in these companies. The recommendations and virtual environment are useful to SME managers and point out ways for future research in the area of learning by simulation in the context of the digital transformation of SMEs.

Keyword: SMEs, simulation, emerging technologies

Índice

<i>Agradecimentos</i>	4
<i>Resumo</i>	6
<i>Abstract</i>	7
<i>Índice</i>	8
<i>Lista de Abreviaturas</i>	9
<i>Índice de Figuras</i>	9
<i>Índice de Tabelas</i>	9
1. <i>Introdução</i>	11
2. <i>Transformação Digital</i>	15
3. <i>Simulação</i>	26
4. <i>Tecnologias Emergentes</i>	33
5. <i>Metodologia de Trabalho</i>	46
6. <i>Resultados</i>	50
7. <i>Discussão</i>	73
8. <i>Conclusões</i>	78
<i>Bibliografia</i>	80
<i>Anexo</i>	90

Lista de Abreviaturas

PMEs – Pequenas e Médias Empresas
TI – Tecnologias de Informação
TIC – Tecnologias da Informação e Comunicação
IOT – Internet of Things
RPA – Robotic Process Automation
VOIL – Virtual Open Innovation Lab

Índice de Figuras

<i>Figura 1 - Framework de Transformação Digital</i>	18
<i>Figura 2 - Factores que afetam a transformação digital</i>	24
<i>Figura 4 - Blockchain</i>	36

Índice de Tabelas

<i>Tabela 1 - Definições de Transformação Digital</i>	16
<i>Tabela 2 - Critérios de Seleção</i>	48
<i>Tabela 3 - Simuladores Blockchain</i>	52
<i>Tabela 4 – Simuladores Finais Blockchain</i>	52
<i>Tabela 5 - Simuladores Cloud&Edge Computing</i>	54
<i>Tabela 6 – Simuladores Finais Cloud&Edge Computing</i>	54
<i>Tabela 7 - Simuladores Digital Twins</i>	55
<i>Tabela 8 - Simuladores Finais Digital Twins</i>	56
<i>Tabela 9 - Simuladores Internet of Things</i>	59
<i>Tabela 10 - Simuladores Finais Internet of Things</i>	60
<i>Tabela 11 - Simuladores Robotic Process Automation</i>	62
<i>Tabela 12 – Simuladores Finais Robotic Process Automation</i>	63
<i>Tabela 13 - Simulador Final CupCarbon</i>	65
<i>Tabela 14 - Simulador Final Anylogic</i>	67
<i>Tabela 15 - Simulador Final EyeSim</i>	69
<i>Tabela 16 - Simulador Final Elemental Battles</i>	71
<i>Tabela 17 - Resumo das Vantagens e Desafios da Transformação Digital</i>	74
<i>Tabela 18 - Resumo das Vantagens e Desvantagens da Simulação</i>	75
<i>Tabela 19 - Lista Final de Simuladores</i>	76
<i>Tabela 20 - Simulador ENERPORT</i>	91
<i>Tabela 21 - Simulador IBM Hyperledger</i>	92
<i>Tabela 22 - Simulador Multichain</i>	94
<i>Tabela 23 - Simulador Bitcoin Testnet</i>	95
<i>Tabela 24 - Ethereum Testnet</i>	97
<i>Tabela 25- Simulador SIMBlock</i>	98
<i>Tabela 26 - Simulador The Principles of Blockchain</i>	99
<i>Tabela 27 - Simulador Elemental Battles</i>	101
<i>Tabela 28 - Simulador EdgeCloudSim</i>	103
<i>Tabela 29 - Simulador CloudSim</i>	104
<i>Tabela 30 - Simulador iCanCloud</i>	105
<i>Tabela 31 - Simulador EMUSIM</i>	106
<i>Tabela 32 - Simulador GROUDSIM</i>	107

<i>Tabela 33 - Simulador GreenCloud</i>	109
<i>Tabela 34 - Simulador CloudAnalyst</i>	110
<i>Tabela 35 - Simulador Anylogic</i>	112
<i>Tabela 36 - Simulador SIMIO</i>	113
<i>Tabela 37 - Simulador SimCad</i>	115
<i>Tabela 38 - Simulador TOSSIM</i>	115
<i>Tabela 39 - Simulador DPWSim</i>	117
<i>Tabela 40 - Simulador iFogSim</i>	118
<i>Tabela 41 - Simulador Cooja</i>	120
<i>Tabela 42 - Simulador WSNet</i>	120
<i>Tabela 43 - Simulador CupCarbon</i>	122
<i>Tabela 44 - Simulador SenseSim</i>	123
<i>Tabela 45 - Simulador OMNeT++</i>	124
<i>Tabela 46 - Simulador Bevywise</i>	127
<i>Tabela 47 - Simulador IOTify</i>	128
<i>Tabela 48 - Simulador IOTSim</i>	130
<i>Tabela 49 - Simulador NetSim</i>	131
<i>Tabela 50 - Simulador Ansys</i>	132
<i>Tabela 51 - Simulador QualNet</i>	133
<i>Tabela 52 - Simulador SimpleIoT Simulator</i>	134
<i>Tabela 53 - Simulador Gazebo</i>	137
<i>Tabela 54 - Simulador USARSim</i>	139
<i>Tabela 55 - Simulador ARGoS</i>	141
<i>Tabela 56 - UberSim</i>	142
<i>Tabela 57 - Simulador EyeSim</i>	144
<i>Tabela 58 - Simulador SimRobot</i>	145
<i>Tabela 59 - Simulador WeBots</i>	147
<i>Tabela 60 - Simulador HEEDS</i>	149
<i>Tabela 61 - Simulador SimCAD</i>	150
<i>Tabela 62 - Simulador ANSYS ACT</i>	153
<i>Tabela 63 - Simulador Leapwork</i>	157

1. Introdução

1.1. Enquadramento

O desenvolvimento da economia e dos mercados implica uma forte competitividade, o que leva a que as organizações sintam necessidade de antecipar e se adaptar às mudanças no mercado. A economia digital coloca uma forte pressão para as empresas inovarem e transformarem os seus modelos de negócios, adotando tecnologias emergentes que podem resultar em vantagens competitivas.

No entanto, para alcançar as vantagens competitivas que podem resultar da adoção de tecnologias emergentes ainda pouco estabelecidas no mercado, os empreendedores têm de compreender os desafios desta adoção. É também importante que os mesmos implementem estratégias que assegurem o sucesso dos processos de adoção. As tecnologias emergentes são complexas, existe ainda pouca experiência sobre os desafios que se colocam à sua adoção, as empresas não têm as competências e conhecimentos necessários à sua integração, é difícil dar apoio a clientes para os produtos e serviços desenvolvidos a partir de tecnologias ainda pouco conhecidas.

À complexidade das tecnologias emergentes há ainda acrescentar a dificuldade em imaginar transformações digitais eficazes, tanto ao nível do modelo de negócio como de produtos e serviços. O esforço de transformação deve conduzir a claros benefícios económicos e organizacionais. Para as PME's pode ainda ser mais complicado já que grande parte destas empresas luta com recursos limitados de conhecimento, financeiros e humanos (Raffaella Manzini, Valentina Lazzarotti, Milena Motta, Silvia Fossati, 2016).

De modo a facilitar e acelerar este processo de transformação digital das empresas, as universidades precisam desenvolver capacidades digitais avançadas para os seus alunos, os futuros funcionários das PME's (Woods & Dennis, 2009).

A complexidade, imaturidade e custo das tecnologias emergentes faz com que a experimentação com estas tecnologias num ambiente seguro seja fundamental para uma decisão de investimento bem fundamentada. A simulação de alterações dos negócios, processos e produtos/serviços apoiam a aprendizagem sobre os impactos de mudanças sem riscos das alterações em contextos reais (Leemkuil, de Jong, & Ootes, 2000).

Este trabalho vai acompanhar o desenvolvimento de um laboratório virtual para experimentação com tecnologias emergentes, assim como o desenvolvimento de guias

práticos para promover a aprendizagem no contexto de iniciativas de transformação digital.

Este trabalho está inserido no contexto do Projeto VOIL – Virtual Open Innovation Lab, ERASMUS+, um projeto que conta com vários eventos e vários parceiros pertencentes à União Europeia. O projeto tem dois objetivos estratégicos: aumentar a conscientização sobre o potencial das tecnologias emergentes para a transformação digital das PMEs, e delinear um modelo pedagógico para a aprendizagem baseada em simulação de competências digitais avançadas. O seu principal impacto será a nível académico, capacitando as universidades a desenvolver competências avançadas de transformação digital nos seus alunos e professores, assim como a nível empresarial mais propriamente a PMEs.

1.2. Objetivos da investigação

A transformação digital é cada vez mais uma realidade e uma necessidade para sobreviver no mercado competitivo. No entanto, as PMEs têm grandes dificuldades em transformar o seu negócio, pois têm recursos limitados.

Cabe às universidades formar os profissionais de modo que quando entrem no mercado sejam capazes de auxiliar as empresas a implementar iniciativas de transformação digital.

É por isso que esta dissertação pretende responder à seguinte questão:

- Como aplicar a aprendizagem baseada em simuladores de tecnologia emergente nos esforços de transformação digital das PMEs?

Para responder à questão acima mencionada foi necessário atingir os seguintes objetivos específicos:

- Definir o conceito de transformação digital: nas próximas secções é apresentada uma síntese dos principais conceitos assim como estudos e aplicações em torno da transformação digital.
- Definir as vantagens e desafios da transformação digital.
- Explicar os desafios da Transformação Digital nas PMEs.
- Explicar a importância da simulação: explicação das vantagens e desafios da aprendizagem baseada na simulação de tecnologias emergentes em situações organizacionais próximo do real.

- Definição do conceito de tecnologias emergentes: apresentação do conceito e dos desafios da aplicação destas tecnologias.
- Seleção e utilização de simuladores em ambiente digital de aprendizagem: acompanhamento da criação e utilização de um ambiente virtual de simulação de utilização de tecnologias emergentes.

Para alcançar os objetivos acima de forma rigorosa é necessário adotar uma abordagem de trabalho bem delineada.

1.3. Metodologia de trabalho

O nosso estudo começa com a realização de uma revisão de literatura focada em nas 3 áreas relevantes para a esta dissertação: transformação digital nas PMEs, aprendizagem baseada na simulação de situações reais, aplicação da aprendizagem baseada em simulação no desenvolvimento de competências digitais avançadas.

A revisão de literatura implica algum tempo de dedicação e uma pesquisa abrangente. Para esse efeito foram utilizadas bases de dados como o Google Scholar (<https://scholar.google.com/>), Scopus (www.scopus.com). Foram também estabelecidos limites temporais para as publicações a recolher, nomeadamente 2016-2020; dada a rápida evolução das tecnologias e simuladores considera-se que o conhecimento desenvolvido nos últimos 5 anos é o mais relevante. Como critério de seleção foram usados os termos “digital transformation”, “emerging technologies”, “technology simulators”, “simulation based learning”. Foram encontrados 99 artigos e da análise do resumo, introdução e conclusões resultou que 49 desses artigos eram relevantes para o trabalho a realizar. Os artigos selecionados contêm uma bibliografia que também ela foi utilizada como referência.

Para apoiar o desenvolvimento do ambiente virtual do projeto VOIL fez-se uma análise inicial de 44 simuladores de tecnologias emergentes. Esta lista de simuladores resultou de uma dissertação anterior (“Avaliação do Potencial Transformador das Tecnologias Emergentes” de Ângela Gil) bem como pesquisas adicionais realizadas em websites e em artigos científicos.

A análise destes simuladores teve como enquadramento resultados do projeto VOIL, nomeadamente as recomendações para a definição de uma estratégia de transformação digital - base para a seleção dos simuladores a integrar no ambiente virtual - bem como o curriculum para desenvolvimento de competências digitais avançadas – base para o desenvolvimento dos guias de aprendizagem.

Da análise destes simuladores resultaram 4 que foram considerados adequados a uma exploração detalhada em ambiente de aprendizagem virtual e foram desenvolvidos guias de aprendizagem por simulação.

1.4. Estrutura da dissertação

Esta dissertação está estruturada por capítulos e as suas subsecções. Começando pelo primeiro capítulo, este apresenta o enquadramento, os objetivos, o método de investigação assim como a estrutura da dissertação.

No capítulo 2, o conceito de Transformação Digital é explorado em detalhe e apresentadas as suas vantagens e os desafios, em particular no contexto das PME e suas limitações de recursos.

No capítulo 3 é feita a exploração do conceito de simulação e apresentadas as vantagens e os desafios a ela associados no contexto da formação de recursos humanos.

No capítulo 4 é abordado o conceito das tecnologias emergentes/disruptivas e os desafios a elas inerentes.

No capítulo 5, é apresentada a metodologia de trabalho usada de maneira detalhada bem como a questão de investigação a responder.

No capítulo 6, são apresentados os resultados do trabalho realizado, os quais são discutidos à luz dos contributos que se pretendia fazer. O capítulo 7 contém as conclusões, nas quais se integra informação sobre as limitações do trabalho e as linhas de trabalho para investigação futura.

Por fim, encontram-se listadas as referências bibliográficas usadas no decorrer da realização do presente trabalho. Esta dissertação inclui ainda anexos com informação adicional.

2. Transformação Digital

Nesta secção é apresentada a revisão da literatura elaborada, mostrando a definição de transformação digital, num contexto geral e, depois, no contexto mais específico das PMEs. Não só é apresentada a definição de transformação digital segundo a literatura como as vantagens e os desafios a ela associados. Este capítulo apresenta ainda casos e iniciativas de transformação digital, os quais permitem compreender os benefícios e desafios práticos destes processos de mudança organizacional. O nosso trabalho foca em particular a transformação digital nas PMEs e as particularidades destas iniciativas em contextos de maior limitação de recursos.

O termo digitalização refere-se à “ação ou processo de digitalização; a conversão de dados analógicos em formato digital” (Päivi Parviainen, Jukka Käriäinen, Maarit Tihinen, Susanna Teppola, 2017).

De acordo com a literatura, a transformação digital refere-se às “mudanças associadas à aplicação da tecnologia digital em todos os aspetos da sociedade humana” (E. Stolterman and A. C. Fors, 2004). Sendo que também é conhecida como “a capacidade de transformar produtos ou serviços existentes em variantes digitais e, assim, oferecer vantagens sobre produtos tangíveis” (O. Gassmann, K. Frankenberger and M. Csik. 2014, E. Henriette, F. Mondher and I. Boughzala, 2015). Já outros autores como S. Brennen e D. Kreiss. (2014) definem a digitalização como a adoção ou o aumento do uso do digital ou tecnologia por uma organização, uma indústria ou um país.

Dehning et al. (2003) e Orlikowski (1996) definem a transformação digital como aquela que realça o impacto das tecnologias de informação (TI) na estrutura, rotina e fluxo de informação numa organização. É também descrita como a capacidade da organização para se integrar e ajustar as TI (Lucas et al., 2013).

Inicialmente, a transformação digital resumia-se às transformações que eram introduzidas numa organização pela adoção das TI. Estas iniciativas estavam normalmente limitadas a melhorias nos processos de negócios das organizações e tinham como objetivos principais a melhoria da eficiência, a redução de custos e a otimização de processos de negócio (Ash & Burn, 2003; Kauffman & Walden, 2001). Já mais recentemente, as mudanças passaram de apenas transformação de processos de negócio para alterações

drásticas a nível dos modelos de negócio (Berman, 2012), da estratégia da organização e cultura (Cui & Pan, 2015; Zeng, Chen, & Huang, 2008), e da construção de alianças comerciais (Ash & Burn, 2003; Chatterjee et al., 2002).

As organizações começaram a implementar iniciativas de transformação digital nos anos 80, adotando tecnologia para acelerar os processos, sendo que nos anos 90 foram adotados os Data Warehouses e nos anos 2000 foram adicionadas redes digitais às organizações, ou seja, era possível ter telemóveis de forma acessível assim como processadores rápidos e computação distribuída. O pico da transformação digital deu-se no início de 2010, uma vez que se deu o crescimento exponencial dos recursos de armazenamento e computação de dados digitais (Ciara Heavin & Daniel J. Porter, 2018).

Encontra-se na seguinte tabela as definições de transformação digital encontradas na literatura:

Autores	Definição
Detuning et. al (2003) e Orlikowski (1996)	“aquela que realça o impacto das tecnologias de informação (TI) na estrutura, rotina e fluxo de informação numa organização”
E. Dtalerman e A. C. Fors (2004)	“mudanças associadas à aplicação da tecnologia digital em todos os aspectos da sociedade humana”
Lucas et. al (2013)	“a capacidade da organização para se integrar e ajustar às TI”
S. Brennen e D. Kreiss (2014)	“a adoção ou o aumento do uso do digital ou tecnologia por uma organização, uma indústria ou um país”
O. Gassman, K. Frankenberger e M. Csik (2014), E. Henriette, F. Mondher e I. Boughzala (2015)	“a capacidade de transformar produtos ou serviços existentes em variantes digitais e, assim, oferecer vantagens sobre produtos tangíveis”

Tabela 1 - Definições de Transformação Digital

A definição que adotamos aqui, e que resulta das definições apresentadas acima, é que **a transformação digital refere alterações nas formas de trabalho, funções e ofertas comerciais potenciadas pela adoção de tecnologias digitais numa organização ou num ambiente operacional da organização.**

A definição dotada neste documento implica que a transformação digital afeta vários níveis nas organizações, causando mudanças em todos eles; as iniciativas de transformação digital são, portanto, sempre complexas pelos impactos sistémicos que acarretam e pela dificuldade em antecipar os benefícios económicos e organizacionais finais. Os benefícios que são normalmente associados à transformação organizacional incluem (Päivi Parviainen, Jukka Kääriäinen, Maarit Tihinen, Susanna Teppola, 2017):

- **Nível dos processos:** adoção de novas ferramentas digitais e racionalização de processos, reduzindo etapas manuais;
- **Nível organizacional:** inovação de serviços, eliminação de práticas obsoletas e transformação da disponibilização de serviços existentes;
- **Nível de domínio comercial:** mudança de papéis e cadeias de valor nos ecossistemas, tornando-as mais ágeis e confiáveis;
- **Nível da sociedade:** mudança de estruturas da sociedade, tornando-as mais flexíveis e alinhadas com as necessidades dos cidadãos.

Reis et al. (2018) categoriza a transformação digital em três elementos:

- **Tecnológica** - transformação digital baseada no uso de novas tecnologias digitais, tais como as redes sociais ou as ferramentas de análise;
- **Organizacional** - transformação digital que requer uma mudança nos processos organizacionais ou a criação de novos modelos de negócio;
- **Social** - a transformação digital é um fenómeno que pode transformar as condições de vida dos cidadãos e melhorar o seu bem-estar.

O nível de transformação digital também varia e é influenciado pelas mudanças das necessidades dos clientes, pelo nível de aplicação das tecnologias, pela posição da liderança da organização e pelo tamanho da organização (Asta Tarutë, Jurga Duobienė, Lina Klovienė, Elena

Vitkauskaitė, Viktorija Varaniūtė, 2018).

A transformação digital vai mais além do que acrescentar ao negócio as tecnologias; ela requer pensar e reestruturar todos os processos de uma organização (Everlin Piccinini, Andre Hanelt, Robert. W. Gregory, Lutz M. Kolbe, 2015). Isto é, todos os processos existentes dentro da organização (todas as tarefas realizadas pelas pessoas) têm de ser pensados e estruturados de modo a tirar vantagem do mesmo, usando as tecnologias.

A literatura relacionada com a transformação digital está a aumentar, mas o foco mais comum continua a ser as tecnologias e áreas mais específicas como a saúde, educação, retalho (Päivi Parviainen, Jukka Kääriäinen, Maarit Tihinen, Susanna Teppola, 2017).

Devido à complexidade e aos desafios que o processo de transformação digital acarreta, os autores Päivi Parviainen, Jukka Kääriäinen, Maarit Tihinen, Susanna Teppola (2017) desenvolveram uma framework para apoiar as organizações:



Figura 1 - Framework de Transformação Digital (adotado de Päivi Parviainen, Jukka Kääriäinen, Maarit Tihinen, Susanna Teppola, 2017)

Segundo este modelo, o primeiro passo a dar quando as organizações decidem passar por um processo de transformação digital é analisar o impacto potencial que a transformação digital vai trazer à organização e decidir que objetivos a organização pretende atingir.

De seguida é necessário rever o estado atual da organização em relação à situação que se deseja atingir bem como antecipar os impactos sistémicos da digitalização. O terceiro passo passa por definir a abordagem que deve ser implementada para preencher a lacuna existente

entre o estado atual da organização e a posição desejada bem como definir as ações necessárias para alcançar a posição desejada.

O último passo é a implementação e validação das ações e, se necessário, refazer aos passos anteriores.

Os autores defendem que este modelo é usado iterativamente para construir gradualmente a solução e ajustar os objetivos e planos, caso seja necessário.

Uma transformação digital consistente e bem-sucedida requer investimento não apenas em artefactos tecnológicos e infraestruturas, mas também nas dimensões estratégicas, intelectuais, estruturais, formais e informais, sociais e culturais (Weick, K.E, 1995; Besson, P. e F. Rowe, 2012; Chan, Y. e B. Reich, 2007). As organizações não têm apenas de investir em nova tecnologia, também necessitam de investir em processos de inovação, em pessoas qualificadas, em cultivar uma cultura de mudança e melhoramento. É inevitável para as organizações que o façam para serem bem-sucedidas quando atravessam um processo de transformação digital.

2.1. Vantagens da Transformação Digital

A crescente utilização das tecnologias de informação e comunicação (TIC) tem um grande impacto na produtividade das empresas e na sua capacidade de inovação. A principal vantagem associada à transformação digital é que esta pode ser vista como uma transformação nos processos, atividades, competências e modelos de uma empresa (Asta Tarutė, Jurga Duobienė, Lina Klovienė, Elena Vitkauskaitė, Viktorija Varaniūtė, 2018). Esta transformação pode melhorar o desempenho da organização.

Para uma transformação digital bem-sucedida, as organizações devem compreender a evolução das tecnologias, identificar as tecnologias mais adequadas aos objetivos de inovação e avaliar o seu impacto na redução de custos e maior produtividade nas organizações. Desta forma, a organização poderá responder melhor a novas oportunidades de mercado, criar conhecimento sobre problemas/soluções e aumentar a eficácia dos seus processos de negócio (Rouach e Santi, 2001; Lichtenthaker, 2004, 2005, 2007; Arman e Foden, 2010).

A substituição do papel e manuais por software permite às empresas que recolham automaticamente dados que podem ser tratados de modo a melhorar a compreensão dos mesmos assim como o melhoramento do desempenho do processo. Permite perceber os custos, as suas causas, os fluxos de trabalho e as consequências sistémicas de decisões (S. Markovitch e P. Willmott, 2014).

Em suma, a digitalização das organizações melhora a sua produtividade, reduz os custos, cria conhecimento e aumenta a eficácia dos seus processos de negócio.

2.2. Desafios da Transformação Digital

A transformação digital é um processo complexo pelo que tem riscos associados. Compete aos gestores das organizações saber gerir e lidar com os riscos, os quais podem ser internos e externos.

Biahmou et al. (2016) afirma que um dos maiores desafios da transformação digital consiste em ajustar constantemente os produtos e serviços às condições em mudança e otimizar continuamente as cadeias de abastecimento por meio de tecnologias. O mercado está em constante mudança e por isso é necessário que as organizações estejam atentas a essa mudança e ajam, adaptando a sua estratégia de transformação digital à mesma.

Já Imran e Kantola (2019) acrescentam que para obter resultados ótimos de desempenho organizacional, as mudanças operacionais no ambiente externo, como por exemplo, mudanças nas tecnologias ou nos mercados, devem ser combinadas com mudanças interorganizacionais compatíveis. Uma vez que as organizações não atuam individualmente e mantém relações umas com as outras (ex^o fornecedores e clientes), as mudanças ocorridas numa organização vão afetar direta ou indiretamente outra organização com a qual mantenha uma relação e até o mercado onde está inserida e por isso é preciso efetuar mudanças internas compatíveis com as externas para que tudo continue a funcionar perfeitamente.

Para uma melhor compreensão destes desafios, Tiersky (2017) descreve as seguintes categorias, onde podem ser agrupados:

- **Prioridades** – os gestores vêm-se confrontados com o dilema de escolher entre o aumento da eficiência das operações atuais ou colocar o foco ser no cliente e no atendimento das suas necessidades. Os dois objetivos podem ser

incompatíveis e o foco na eficiência pode reduzir a satisfação do cliente, a sua lealdade e as suas compras.

- **Dados agregados ou personalizados** – a agregação de dados enfatiza a previsão do comportamento do cliente, o que pode levar a procurar padrões e ignorar as necessidades específicas de cada cliente.
- **Seleção do perfil de recursos humanos a contratar e compensar** (Panetta, 2016) – torna-se difícil avaliar o retorno financeiro de ter mais Data Scientists e pessoal de TI em comparação com mais formação e recursos para gestores e pessoal em áreas funcionais.
- **Definir que dados devem ser armazenados** – o armazenamento de dados tem um custo e é difícil perceber quais são os dados necessários capturar e analisar.
- **Definição de equilíbrios sociotécnicos potenciadores de realização profissional** (Newman, 2016) – as máquinas e os robôs continuam a provocar alguma desconfiança aos trabalhadores, pois têm medo de serem substituídos e isso provoca alguma resistência à mudança.
- **Segurança vs acessibilidade** (Filkins et al., 2016) – é um grande desafio em muitas áreas (exº medicina); os dados podem ser de fácil acesso, mas serem difíceis de usar, ou seja, os dados podem estar disponíveis de forma rápida e fácil, mas não serem compreendidos pelo utilizador.

Tal como já foi referido a transformação digital envolve implementar capacidades digitais para suportar o modelo de negócio, o que provoca impactos enormes nas organizações, desde das operações do dia-a-dia, a recursos, processos e utilizadores. Os gestores têm de lidar com paradoxos organizacionais que resultam da transformação das TI (Gregory et al. 2015; Robey 1997).

Existem outros desafios como por exemplo a construção de colaborações com diferentes parceiros na área do negócio e das TI para a criação de novos modelos de negócio e a garantia da segurança na adoção e implementação de novas tecnologias (Everlin Piccinini, Andre Hanelt, Robert. W. Gregory, Lutz M. Kolbe, 2015).

Estes desafios estão associados à implementação de componentes digitais e a

experiências dos clientes digitais no produto, assim como com a lógica do negócio expresso no modelo de negócio. É por isso necessário compreender e ser capaz de reagir rapidamente às mudanças do ecossistema (Everlin Piccinini, Andre Hanelt, Robert. W. Gregory, Lutz M. Kolbe, 2015).

Estão ainda relacionados com a necessidade de integrar conhecimento de áreas distintas e o papel das TI na organização bem como a mentalidade das pessoas que trabalham em organizações tradicionais com uma estrutura inflexível que resiste à inovação e à mudança (Everlin Piccinini, Andre Hanelt, Robert. W. Gregory, Lutz M. Kolbe, 2015).

Isto leva a que muitas organizações não são bem-sucedidas na transformação digital e porque as mesmas não mudam as suas mentalidades nem processos e não constroem uma cultura de mudança.

A falta de uma estratégia de digitalização e pessoas competentes também são dois dos principais obstáculos para a transformação digital bem como as preocupações com a segurança e insuficiência de recursos (Päivi Parviainen, Jukka Kääriäinen, Maarit Tihinen, Susanna Teppola, 2017), sendo que esta estratégia deve coordenar os vários tópicos da transformação digital e ajudar as organizações a percorrer a complexidade e ambiguidade identificando os seus próprios "pontos doces" digitais (Thomas Hess, Christian Matt, Alexander Benlian, Florian Wiesböck, 2016).

Um estudo realizado na Índia revelou que fatores como a idade, a qualificação educacional, o tamanho da organização e a sua localização são fatores que influenciam a transformação digital. As empresas mais novas, por norma, estão mais cientes da tecnologia e procuram evoluir mais rapidamente e estão mais recetivas a novas tecnologias do que as mais velhas. Isto leva a outra questão também importante que é o nível de educação das pessoas que fazem parte da organização: quanto mais sensibilizadas estão em transformação digital, melhor será o processo na organização.

O tamanho das organizações também é um fator importante no sentido que as PMEs diferem nas suas necessidades, expectativas, preferências e perceções com base no seu nível de investimento. A sua localização (no centro urbano ou suburbano) também acaba por influenciar a transformação digital, sendo que maior parte das empresas que está mais recetiva à transformação digital está localizada numa área urbana.

2.3. Transformação Digital nas PMEs

Para tirar proveito da economia digital, as empresas e, em particular, as pequenas e médias empresas (PMEs), precisam integrar o conhecimento sobre o potencial disruptivo das tecnologias de forma a serem capazes de transformar o seu modelo de negócios para aproveitar esse potencial.

Apesar dos riscos que derivam da partilha de competências e de conhecimento, as PMEs têm vindo a abrir o seu processo de inovação (Van de Vrande et al., 2009; Lee et al., 2010). Mas para isso precisam de um processo de Inteligência Tecnológica (Veugelers et al., 2009; Mortara et al., 2012, Lichtenthaler, 2005, 2007; Schoemaker et al., 2013), ou seja, “um processo de aquisição sistemática, avaliação e comunicação de informações sobre tecnologias que são tendência para procurar oportunidades e ameaças em tempo real” (Lichtenthaler, 2007).

A competição entre as PMEs é diferente das grandes empresas por causa da frequente escassez de recursos (Kim, Song, Lee, 1993), nomeadamente falta de recursos humanos qualificados e recursos financeiros.

As PMEs têm uma maior dificuldade em entrar no mercado devido ao seu tamanho significativamente limitado, capacidade tecnológica, conhecimento de marketing e poder de marca (Chu, 2009). Estas tendem a sofrer mais devido ao ambiente turbulento e competitivo onde operam (Moreno, Pinheiro, Joia, 2012), no entanto são essenciais para manter o crescimento da economia.

De acordo com Marchand, Kettinger, e Rollins (2002), as PMEs que implementam iniciativas de transformação digital acabam por fazer uma utilização mais eficaz da informação e apresentam um melhor desempenho comercial.

A transformação digital torna-se algo mais complexo para as PMEs devido aos seus custos. O processo de transformação digital por parte das PMEs depende das suas características, nomeadamente da dimensão da empresa, a indústria onde está inserida, a localização, a cultura praticada na mesma e a atitude perante a mudança (Stephen Burgess, 2002).

2.3.1. Desafios da Transformação Digital nas PMEs

A transformação digital traz às PMEs alguns desafios que as maiores empresas não enfrentam, nomeadamente escassez de recursos, falta de conhecimento especializado e impacto limitado no mercado (Carson & Cromie, 1990).

Segundo a Comissão Europeia (2015), a falta de conhecimento entre os colaboradores pode ser vista como um desafio nas PMEs uma vez que as mesmas não têm os recursos humanos com as capacidades necessárias o que leva à necessidade de formação e educação para melhorar as competências e conhecimentos profissionais (Cegarra-Navarro, J.G. e F. Dewhurst, 2007).

Muitas vezes um processo de transformação digital requer uma abordagem generalizada, não especificando apenas um problema ou uma área a ser melhorada, o que não tem em conta as necessidades específicas de uma PME, sendo que a abordagem escolhida deve estar alinhada com os objetivos de negócio (Levy, M., P. Powell, e P. Yetton, 2011).

Os fatores que afetam a transformação digital nas PMEs podem ser divididos entre fatores internos e externos.

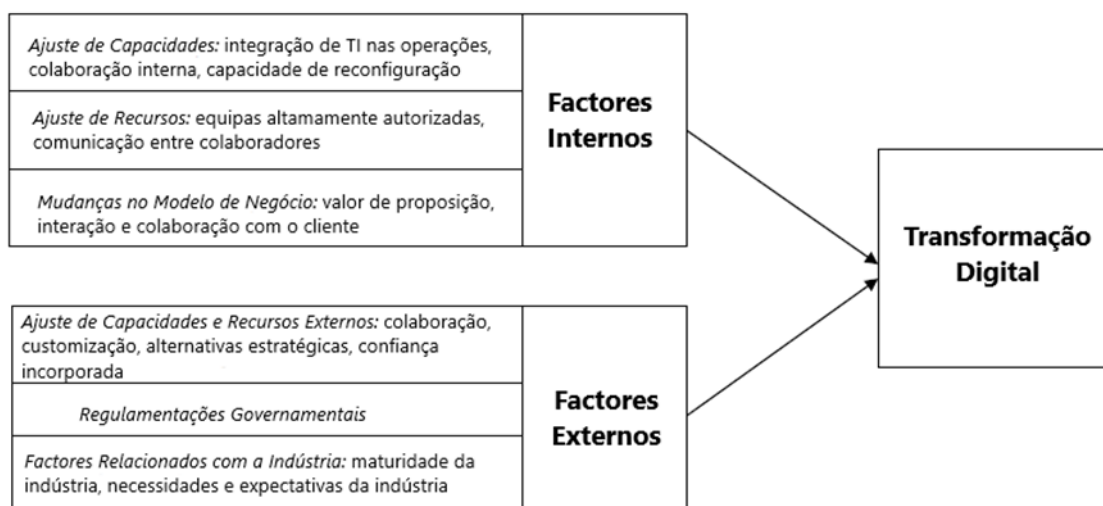


Figura 2 - Factores que afetam a transformação digital (adotado de Asta Tarutė, Jurga Duobienė, Lina Klovienė, Elena Vitkauskaitė, Viktorija Varaniūtė, 2018).

Ajuste de Capacidades - Uma transformação digital bem-sucedida requer que uma empresa desenvolva uma ampla gama de capacidades, que variam em importância dependendo do contexto empresarial e das necessidades específicas da empresa (Reis et al, 2018)

Ajuste de Recursos –para uma transformação digital bem-sucedida é preciso haver uma mudança interna nos recursos e uma coordenação e promoção dos mesmos e também das novas estruturas de empresas (Liu, Chen, & Chou, 2011).

Mudanças no modelo de negócio – são necessárias mudanças no modelo de negócio uma vez que a transformação digital requer mudanças e melhoramentos nas tarefas diárias da organização e o modelo de negócio é o essencial das organizações (Berman, 2012; Kaltum, Widodo, & Yanuardi, 2016; Krüger & Teuteberg, 2016; Remane et al., 2017).

Ajuste de capacidades e recursos externos - as capacidades externas como colaboração e customização são cruciais, pois através da customização é que uma organização se diferencia das outras. E é através da colaboração que as organizações podem inovar e ajudar-se mutuamente. De acordo com Liu, Chen e Chou (2011, p. 1736) "a colaboração é um processo recursivo em que duas ou mais organizações trabalham juntas para um objetivo comum".

Regulamentações governamentais – as regulamentações governamentais é um dos mais importantes fatores externos que impactam o processo de transformação digital numa empresa, uma vez que são os mesmo que ditam leis que as organizações obrigatoriamente têm de cumprir, o que pode limitar a liberdade de inovar e transformar (Krüger & Teuteberg 2016).

3. Simulação

A **aprendizagem em simulação** foi definida por Gaba (2004) como uma estratégia educacional que substitui ou amplifica as experiências que replicam aspectos do mundo real de forma interativa. Cioffi (2001) define a simulação como uma estratégia educacional utilizada para alcançar objetivos, pôr em prática conceitos explorados teoricamente.

O objetivo principal da **simulação** é "replicar alguns ou quase todos os aspectos essenciais de uma situação real, para que a situação possa ser mais facilmente compreendida e gerida quando ocorrer de facto" (Hovancsek, 2007).

Segundo Roddenberry, Johnson e Daniels (1966), a simulação permite ao aprendente ir mais além, prever as consequências das suas ações. Mas isso só é possível se o aprendente estiver ciente que o objetivo principal é aprender através da simulação.

Esta técnica de aprendizagem por simulação tem sido bastante usada e desenvolvida nas áreas de saúde, nomeadamente da medicina e da enfermagem (Yuan et al.,2012), sendo que a mesma ajuda a desenvolver capacidades como comunicação, trabalho em grupo, pensamento crítico e autoavaliação (Engel,1997). A simulação deve ser complementada com outras abordagens de aprendizagem.

Devido ao grande desenvolvimento da simulação nas áreas da medicina e enfermagem, maior parte da literatura retrata casos relativos a estas áreas, não havendo muita literatura referente a simulação em áreas de tecnologia. Este foi um desafio com o qual tivemos de lidar durante a elaboração deste trabalho.

Como referido, a área da medicina e da enfermagem são pioneiras na simulação, sendo que a aprendizagem baseada em simulação é reconhecida por estas áreas como uma peça essencial na aprendizagem dos seus alunos (Ricketts, 2011), pois através da simulação os alunos aprendem em como conciliar a teoria com a prática num ambiente laboratorial seguro. Além disso, as simulações fazem cada vez mais parte dos planos curriculares dos cursos destas áreas (Aggarwal R, Darzi A, 2009).

As sessões de formação em simulação, com objetivos específicos de aprendizagem, oferecem a oportunidade de passar pelas etapas do ciclo experimental de maneira estruturada. Além disso, a maior parte das vezes, combinam a componente experimental do exercício de simulação com uma análise e reflexão posterior sobre a experiência, visando assim facilitar a

implementação de mudanças na forma de pensar e agir. A simulação oferece assim uma oportunidade de experienciar algo de modo prático e de forma controlada, sendo que depois tal experiência pode ser analisada e refletida (Ruth M. Fanning, David M. Gaba, 2007).

A simulação pode também ser usada como uma ligação entre os alunos e os profissionais que se vão tornar, uma vez que esta lhes fornece situações reais com as quais se podem encontrar na vida profissional. Este tipo de ambiente encoraja e também requer aprendizagem ativa e crítica baseada em ações (José Miguel Padilha, Paulo Puga Machado, Ana Leonor Ribeiro, José Luís Ramos, 2018).

Define-se assim a simulação como um ambiente seguro, onde os conceitos teóricos podem ser postos em prática. Define-se como um ambiente que facilita uma situação de aprendizagem de uma situação real e permite o desenvolvimento de competências.

3.1. Vantagens e Desafios da Simulação na formação Recursos Humanos

A simulação permite que sejam testados vários cenários, mudando parâmetros ou adicionando recursos. Isto torna a simulação numa vantagem no sentido que é possível observar o que acontece ao adotar certo tipo de comportamento, ou seja, quais as consequências da mudança planeada (Amr Mahfouz, Jonh Shea, Amr Arisha, 2011).

Deste modo, podemos observar que a simulação ajuda a prever riscos e identificar como podem ser geridos de modo a causar o menor impacto possível (Kallio-Gerlander & Collan, 2006).

É possível afirmar que a simulação fornece a oportunidade às pessoas de aplicarem o seu conhecimento na prática e encontrarem soluções para os seus problemas (Peltonen, 2008). Uma vez que oferece um cenário fictício representativo da realidade, são possíveis testar várias opções e decisões de modo a encontrar soluções. Permite também o desenvolvimento das capacidades interpessoais tais como o pensamento criativo, autoconfiança, pois com a possibilidade de testar vários cenários, a pessoa desenvolve capacidades críticas em relação às decisões que toma (Kallio-Gerlander & Collan, 2006).

As pessoas aprendem melhor quando estão ativamente envolvidas e podem participar, desempenhando o seu papel. Podem experimentar não apenas eventos concretos de maneira cognitiva, mas também eventos transacionais de maneira emocional. Este tipo de aprendizagem

é exatamente o tipo de aprendizagem usada em simulação (Ruth M. Fanning, David M. Gaba, 2007), ou seja, a simulação permite que as pessoas possam experimentar e fazer parte da experiência, aprendendo com essa experiência.

Na revisão de literatura realizada, são muitos os artigos onde é demonstrado que os alunos submetidos à simulação como técnica de aprendizagem estão satisfeitos em usá-la, pois, segundo os mesmos, melhora níveis de confiança, o pensamento crítico, o julgamento crítico e as capacidades (Robyn P. Cant, Simon J. Cooper, 2017; Al-Ghareeb, Cooper, 2016; Adamson, 2015; Berndt, 2014; Cant, Cooper, 2010, 2014; Fisher, King, 2013; Norman, 2012; Skrable, Fitzsimons, 2014; Stroup, 2014; Weaver, 2011; Yuan et al., 2012b).

A mesma melhora a confiança dos que a ela recorrem, uma vez que permite que testem o mesmo cenário vezes sem conta e melhorarem as suas capacidades e os seus erros, ganhando assim confiança.

Tal como já foi referido anteriormente, a área da saúde é uma pioneira na simulação. Estes acreditam que a principal vantagem da simulação é a facilitação da aprendizagem prática, levando ao desenvolvimento de competências clínicas e ao aumento da autoconfiança no atendimento do paciente (Brydges, Nair, Ma, Shanks, & Hatala, 2012; Hayden, Smiley, Alexander, Kardong-Edgren, & Jeffries, 2014; Lateef, 2010; McCaughey & Traynor, 2010). Esta estratégia pode funcionar como um acelerador de conhecimento e contribuir para a futura prática de enfermagem com base em padrões de qualidade mais seguros e mais altos (José Padilha, Paulo Machado, Ana Ribeiro, José Ramos, 2018).

Existem evidências que comprovam a melhoria na transferência de conhecimento do aluno adquirida em aula (Tschannen, Aebersold, McLaughlin, Bowen & Fairchild, 2012) com recurso a simulação virtual, mesmo nas situações clínicas mais complexas, como por exemplo a gestão de sintomas (Tiffany & Hogle, 2016). O conhecimento adquirido através da simulação é melhor do que aquele adquirido através de outros métodos de aprendizagem (Stroup, 2014).

A simulação tem os seus desafios, sendo que um deles é “como medir o desempenho”. O desempenho envolve vários domínios da atividade humana: aspetos cognitivos (fatos e números), afetivos (valores pessoais, crenças) e psicomotores (obtenção de habilidades técnicas), o que torna o processo de avaliação de desempenho, um processo bastante complexo e difícil (Kardong- Edgren et al., 2010).

Para medir o domínio psicomotor são usados ensaios controlados e estudos quantitativos (estes também usados para aspetos cognitivos) (Cant, Cooper, 2016).

A orientação certa é considerada outro dos desafios da simulação. Os participantes precisam estar bem orientados para as capacidades e limitações do ambiente simulado, para que possam usufruir totalmente da experiência de aprendizagem. Isto é, os participantes devem ter a noção que apesar de a experiência ser “fictícia”, podia ser uma situação real, por isso devem comportar-se e tomar decisões como se fosse real (Virginia C. Muckler, 2017).

Em 2017, a autora Virginia C. Muckler, dedicou-se a estudar quais os fatores determinantes para a descrença na simulação, sendo que identificou 5: fidelidade, contrato fictício, segurança psicológica, aceitação emocional e significado atribuído.

Estes fatores devem ser vistos como desafios associados à simulação, uma vez que não permite que as pessoas acreditem na simulação. Sendo assim passo a explicar cada um deles, mais detalhadamente:

- **Fidelidade:** o grau de fidelidade influencia a maneira como a simulação é aceite; alguns participantes sentem e percebem a simulação de maneira diferente por isso o facilitador é responsável por determinar e implementar estratégias mais eficazes para garantir a fidelidade e imersão do participante na simulação, sendo que a simulação deve recriar as mesmas sensações e percepções da realidade (Wilson, Wittmann-Price, 2015; Kneebone, 2010). Por outro lado, a simulação de alta fidelidade pode sobrecarregar alunos iniciantes. Ou seja, a fidelidade pode ser confundida pela complexidade da tarefa, uma vez que, à medida que a fidelidade da simulação aumenta, geralmente há um aumento proporcional na complexidade da tarefa de aprendizagem (Norman G et. al, 2012; Artino AR J., Durning SJ, 2012; van Merriënboer JJG, Sweller J, 2010).
- **Contrato fictício:** o contrato fictício deve ser introduzido antes das atividades de simulação e tem como objetivo comprometer os participantes na simulação estabelecendo regras básicas e expectativas, incluindo as de comunicação e respeito mútuo, e garantir aos participantes que a intenção é que a aprendizagem ocorra sem repercussões por erros (Dieckmann, Friis, Lippert,

& Ostergaard, 2014; Simon et al., 2010; Arafeh, Hansen, & Nichols, 2010; Rudolph et al., 2014). Estando os participantes comprometidos com a simulação, alcançam a imersão total da simulação (Cheng et al, 2007).

- **Segurança psicológica:** a segurança psicológica no local de trabalho envolve as percepções que os indivíduos têm sobre as consequências de assumir riscos (Edmondson, 1999, 2002, 2003), esta permite que as pessoas superem os seus medos e arrisquem sair da zona de conforto. Ou seja, permite que os participantes estejam mais envolvidos na simulação. Para criar um ambiente de aprendizagem seguro na simulação Rudolph et al. (2014) sugerem que os facilitadores esclareçam as expectativas, criem um contrato de ficção, executem a simulação com atenção aos detalhes e respeitem os alunos e a sua segurança psicológica.
- **Aceitação emocional:** a aceitação emocional ocorre quando os participantes experimentam um apego emocional ao cenário de simulação, esta conexão emocional permite que os participantes se envolvam emocionalmente na simulação com o objetivo de aprender (Bauman, 2012). Para conseguir esta aceitação emocional, os facilitadores devem garantir o envolvimento emocional, detalhando o papel de cada participante na simulação (Kear, 2015).
- **Significado atribuído:** os alunos atribuem significado às suas experiências de simulação (Arafeh et al., 2010; Rudolph et al., 2014). A aprendizagem é mais significativa quando o conteúdo ou a habilidade aprendida é relevante para o aluno e aplicável ao que ele está a tentar realizar (Knowles, 1980), pois os alunos atribuem significado às experiências de simulação com base em suas percepções (Rutherford-Hemming, 2012). As percepções e significados atribuídos à simulação variam, assim como as reações individuais à simulação.

Um dos desafios é o facto que os utilizadores podem perder o senso de realidade. Têm noção que estão a “jogar” e são livres de tomar qualquer decisão, independentemente se na realidade essa decisão não seja possível (Karoliina Nisula, Samuli Pekkola, 2012).

3.2. Simulação nas PMEs

Anteriormente foi discutida a revisão de literatura realizada sobre a transformação digital nas PMEs. Foi referido o quão é complicado para as mesmas passar por um processo de transformação digital devido a escassez de recursos necessários e falta de conhecimento especializado (Carson & Cromie, 1990).

A literatura é escassa quanto à utilização de simulação nas PMEs. Poucos são os artigos publicados que falam sobre o uso da simulação nas PMEs. Isso deve-se em parte ao elevado custo das ferramentas de simulação e ao conhecimento necessário para desenvolver modelos/*frameworks* de confiança. No entanto, é necessário que as PMEs melhorem os seus processos e apliquem ferramentas de simulação, particularmente nos aspetos operacionais dos seus negócios (James O’Kane, Antonios Papadoukakis, 2007).

Através da revisão de literatura sobre simulação podemos dizer que esta pode ser um grande auxiliar não só para as PMEs, mas também para todas as organizações, pois através da simulação estas organizações conseguem testar vários cenários, novas tecnologias e treinar os seus recursos humanos. É em particular essencial para as PMEs, uma vez que as mesmas têm escassez de recursos humanos e financeiros para apostar em novas tecnologias e mudanças.

As PMEs mostram alguma relutância em contratar recém graduados, uma vez que estes não têm experiência e as PMEs também não têm a infraestrutura necessária para os ensinar pois exige tempo, dinheiro e constante assistência (Karolina Nisula, Samuli Pekkola, 2012). Neste sentido, a simulação seria uma grande ajuda pois permite a formação de recursos humanos, dota-os das capacidades necessárias.

Seguindo a anterior linha de pensamento, é possível afirmar que não é fácil transferir conhecimento entre pessoas, principalmente numa PMEs, onde os funcionários estão mais preocupados em dar lucro à empresa e satisfazer as exigências dos clientes do que a ensinar os novos que entram na organização. Além disso está comprovado que as pessoas aprendem mais facilmente numa situação prática, onde desempenham um papel, portanto, através da simulação as pessoas podem aprender através de contextos reais e ensaiar e também aprender com os resultados (Karolina Nisula, Samuli Pekkola, 2012).

As PMEs têm um receio maior em apostar em novas tecnologias do que as grandes empresas, pois caso sejam malsucedidas podem ter consequências catastróficas ou até mesmo falência (Mahmood Ali, Joanna Culliane, 2014). Daí ser essencial ter um ambiente controlado

onde se possam testar e ver as consequências da adoção de novas tecnologias na empresa, que resultados esperar.

A simulação permite que as PMEs testem cenários “what-if”, evitando assim o risco de falharem ao implementarem novos processos (James O’Kane, Antonios Papadoukakis, 2007). Estes cenários “what-if” podem mostrar como os processos operarão quando as variáveis que influenciam o processo forem introduzidas, modificadas ou retiradas, o que também permite a comparação com as existentes, levando assim a melhores decisões com base nos resultados obtidos (Francesco Aggogeri, Rodolfo Fagila, Marco Mazzola, Angelo Merlo, 2014).

4. Tecnologias Emergentes

Nesta secção é apresentada a revisão da literatura elaborada, para o desenvolvimento do conceito de tecnologias emergentes e os desafios a ela subjacentes, bem como a análise de vários simuladores (OpenSource e Comerciais) para várias tecnologias emergentes. ¹

4.1. Desafio das Tecnologias Emergentes

O conceito de tecnologias emergentes continua a não ser consensual entre a comunidade científica. No entanto, o assunto tem vindo a ser discutido de forma a clarificar as várias características que uma tecnologia deve apresentar para ser emergente.

Algumas definições, como a de Porter et al. (2002), enfatizam o potencial impacto das tecnologias emergentes na economia e na sociedade; já outros autores como Boon and Moors (2008) dão mais importância à incerteza associada ao processo emergente enquanto Small et al. (2014) salientam as características novas e o seu rápido crescimento.

Raji Srinivasan (2008) define as tecnologias emergentes como inovações científicas com o potencial de criar uma nova indústria ou transformar uma já existente e o *Business Dictionary* define como “novas tecnologias que estão em desenvolvimento ou serão desenvolvidas nos próximos cinco a dez anos e que alterarão substancialmente o ambiente social e de negócios”.

No entanto, uma tecnologia não precisa de ser nova para ser considerada uma tecnologia emergente, basta que não esteja amplamente disponível no mercado ou não seja muito usada (Vine, 2002), sendo também considerada emergente quando não esta inserida num determinado contexto. Por exemplo, a tecnologia é usada num determinado contexto para uma determinada tarefa e depois passa a ser usada noutra contexto nunca antes usado (Halaweh, 2013).

D. Rotolo et al. (2015) defende que uma tecnologia emergente é uma tecnologia de crescimento radicalmente novo e relativamente rápido. É principalmente caracterizada por uma certa coerência persistente ao longo do tempo e com o potencial de exercer um impacto considerável numa indústria, negócio ou sociedade. No entanto, o seu maior impacto é relativamente ao futuro e, portanto, a fase de emergência da tecnologia é incerta e ambígua. Acrescenta a esta definição que uma tecnologia emergente tem 5 atributos chave:

- **Novidade radical:** uma tecnologia emergente é considerada radicalmente nova (Small et al., 2014), isto é, as tecnologias emergentes cumprem uma determinada função usando um princípio básico diferente em comparação com o que foi usado

¹ São apresentados alguns termos em inglês para a qual não foi encontrada nenhuma tradução. Os termos encontram-se em itálico.

anteriormente para atingir um objetivo semelhante. Por esta razão é também usado o termo tecnologia disruptiva pois rompe com conceitos e práticas anteriores.

- **Crescimento relativamente rápido:** uma tecnologia emergente mostra taxas de crescimento maiores comparativamente a tecnologias não emergentes. Este rápido crescimento pode ser observado na quantidade de documentos publicados referentes à tecnologia (D.Rotolo et al., 2015);
- **Coerência:** uma tecnologia emergente pode demorar a aparecer e a marcar a sua posição no mercado, no entanto quando o faz, é coerente e persistente. Isto é, chega a uma fase que mantém-se estável e persiste em afetar vários mercados, criando a sua identidade (D.Rotolo et al., 2015);
- **Impacto proeminente:** as tecnologias emergentes exercem um impacto notável em domínios específicos, ou seja, criam grandes mudanças, alterando a composição de atores, instituições, padrões de interação entre os mesmos e os processos de produção de conhecimento associados (D.Rotolo et al., 2015);
- **Incerteza e Ambiguidade:** as tecnologias emergentes são caracterizadas pela incerteza dos seus possíveis resultados e utilização, que podem ser indesejados e indesejáveis, bem como pela ambiguidade nos significados que diferentes grupos sociais associam à tecnologia em questão (Stirling, 2007; Mitchel,2007).

As tecnologias emergentes afetam drasticamente organizações e os mercados como os conhecemos e isto deve-se ao facto de as tecnologias afetarem produtos e serviços como também os comportamentos dos consumidores e vantagens estratégicas das organizações e o que tem implicações económicas. Nenhuma indústria está imune aos efeitos que elas causam (Raji Srinivasan, 2008), pois sendo uma fonte de recursos tecnológicos, estas criam oportunidades e ameaças.

Uma tecnologia emergente diferencia-se de outras tecnologias pelo seu impacto social e ético, pois não é visto ou conhecido quando a tecnologia chega ao mercado ou a um contexto específico (Mohanad Halaweh, 2013).

A definição que adotamos aqui, e que resulta das definições apresentadas acima, é que uma tecnologia emergente é considerada radicalmente nova, e mostra taxas de crescimento maiores comparativamente a outras tecnologias. Pode demorar a aparecer e a marcar a sua posição no mercado, no entanto, quando tal acontece, exerce um impacto notável, ou seja, cria grandes mudanças. A sua utilização vai evoluindo ao longo do tempo e os resultados organizacionais e sociais da sua utilização vão-se revelando ao longo do tempo.

As características e os efeitos das tecnologias emergentes apresentem vários desafios para a inovação organizacional, resultando em oportunidades para os investigadores. Estes desafios passam pela alta complexidade, a volatilidade destas tecnologias e o facto de ser impossível determinar os retornos potenciais aos investimentos em tecnologias emergentes, não apenas por causa da incerteza do valor da tecnologia, mas também porque o valor pode ser substituído pelas tecnologias emergentes mais recentes (Raji Srinivasan, 2008).

O futuro de uma tecnologia emergente é desconhecido ou imprevisível, sendo que muitas organizações não querem correr o risco de apostar nelas para introduzir mudanças organizacionais (Stahl, 2011). O risco está ligado à incerteza associada a como por exemplo o custo, a taxa de penetração, os padrões e as especificações da tecnologia e a aceitação e reações do utilizador.

A taxa de adoção das tecnologias emergentes aumenta com o tempo: começa com baixa disponibilidade (em mercados, sociedades e países) e baixo efeito de rede. No entanto, a disponibilidade aumenta com o tempo à medida que o número de fabricantes e fornecedores aumenta e se espalha através das fronteiras e dos países industrializados (Mohanad Halaweh, 2013).

Um dos desafios associados às tecnologias emergentes é o seu custo. Uma nova tecnologia emergente é sempre cara no começo por causa dos volumes iniciais de produção baixos, mas com o tempo este custo diminui.

4.1.1. As tecnologias emergentes abordadas na dissertação

Durante o desenvolvimento desta dissertação foram abordadas cinco tecnologias emergentes: *blockchain*, *cloud and edge computing*, *digital twins*, *internet of things* e *robotic process automation*.

Nesta secção é feita a revisão de literatura de cada uma das tecnologias e as vantagens e desafios associados às mesmas.

4.1.1.1. Blockchain

Conceito

Blockchain é uma tecnologia de gestão descentralizada de dados e transações, desenvolvida primeiro para a criptomoeda Bitcoin (Jesse Yli-Huumo, Deokyoon Ko, Sujin Chol, Sooyong Park, Kari Smolander, 2016).

Uma blockchain integra uma sequência de blocos que corresponde a uma lista completa de registos de transações, como *conventional public ledger* (Lee Kuo Chuen, 2015), sendo que cada bloco aponta para o bloco imediatamente anterior através de uma referência que é essencialmente

um valor de hash do bloco anterior chamado bloco “pai”; a cadeia de blocos vai crescendo continuamente quando novos blocos são anexados (Zibin Zheng, Shaoan Xie, Hong-Ning Dai, 2018).

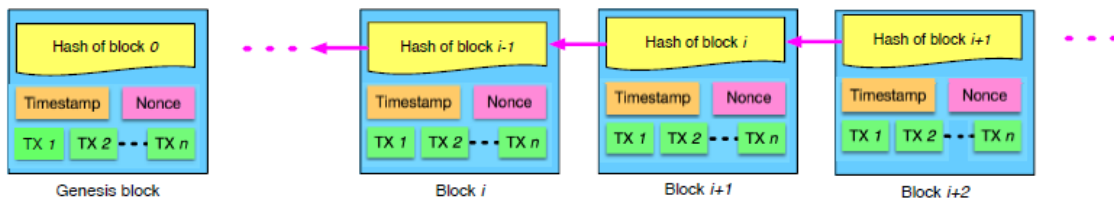


Figura 3 - Blockchain (adotado de Zibin Zheng, Shaoan Xie, Hong-Ning Dai, 2018)

Para realizar transações, o utilizador tem de possuir duas chaves: uma pública e uma privada. A chave privada é usada para assinar as transações e é confidencial; depois das transações assinadas digitalmente estas espalham-se por toda a rede e podem ser acedidas por chaves públicas, visíveis para todos na rede (Zibin Zheng, Shaoan Xie, Hong-Ning Dai, 2018).

As principais características de uma blockchain incluem descentralização, persistência, anonimato e auditoria (Zibin Zheng, Shaoan Xie, Hong-Ning Dai, Xiangping Chen, Huaimin Wang, 2017), o que torna esta tecnologia uma das que tem maior potencial para revolucionar a indústria e o comércio, promovendo mudanças económicas a nível global. Podemos ver esta mudança no sector financeiro, onde a blockchain pode ser aplicada em várias áreas tais como compensação e liquidação de ativos financeiros.

Esta tecnologia permite o desenvolvimento de soluções seguras, rápidas, de confiança e transparentes que podem tanto ser públicas como privadas.

Vantagens e Desafios

Ao contrário do típico sistema transaccional, que é centralizado, e todos os dados e informação são controlados e geridos por uma terceira parte além das duas entidades envolvidas na transação, a *blockchain* caracteriza-se pela sua descentralização. Ou seja, não há necessidade de haver uma terceira parte envolvida na transação, como um banco (Yli-Huumo et. al 2016).

Esta característica da *blockchain* traz vantagens como a transparência e anonimato uma vez que todos os nós da *blockchain* são anónimos (Yli-Huumo et. al 2016); também implica a redução de custos e de tempo, não tendo a transação de passar por uma entidade terceira.

Em suma, podemos caracterizar e dizer que as vantagens da *blockchain* são:

- **Descentralizada:** ao contrário dos sistemas convencionais de transações, a *blockchain* não requer uma entidade terceira que valide a transação, apenas requer as duas entidades que desejam efetuar a transação, daí ser descentralizada. Esta descentralização também permite reduzir significativamente os custos do servidor

(incluindo o custo de desenvolvimento e o custo da operação) e mitigar os gargalos de desempenho no servidor central, uma vez que apenas requer duas partes, não tendo de recorrer à terceira entidade, gastando assim tempo e recursos (Zheng et. al, 2018).

- **Persistência:** todas as transações precisam de ser confirmadas e registadas em blocos distribuídos por toda a rede, sendo difícil adulterar a informação. Qualquer falsificação pode ser detetada facilmente, uma vez que cada bloco transmitido tem de ser verificado/validado pelos restantes nós da rede (Zheng et. al, 2018).
- **Anonimato:** cada utilizador pode interagir com a rede com um endereço gerado pela rede *blockchain* e pode gerar vários para evitar a exposição de identidade. Assim o anonimato é preservado durante as transações efetuadas, através deste mecanismo (Zheng et. al, 2018).
- **Auditoria:** cada uma das transações na *blockchain* é validada e registada com a data/hora, sendo que cada utilizador pode facilmente verificar e rastrear os registos anteriores tendo acesso a qualquer nó na rede, o que melhora a rastreabilidade e a transparência dos dados armazenados na *blockchain* (Zheng et. al, 2018).

Mas como qualquer tecnologia, a *blockchain* também tem os seus desafios. Desafios esses que passam por escalabilidade, perda de privacidade, segurança, desperdício de recursos (Zheng et. al, 2018; Yli-Huumo et. al 2016).

Com a quantidade de transações a aumentar dia a dia, a *blockchain* torna-se “pesada”. Devido ao tamanho do bloco e o intervalo de tempo usado para gerar um novo bloco, a *blockchain* da Bitcoin, por exemplo, pode processar apenas cerca de 7 transações por segundo, ou seja, não consegue lidar com o processamento de milhões de transações em tempo real (Zheng et. al, 2018). Devido ao aumento da frequência das transações na *Blockchain*, a escalabilidade da rede Blockchain precisa de ser aprimorada para conseguir lidar com tanta transação (Yli-Huumo et. al 2016).

Acredita-se que a blockchain seja muito segura, pois os seus utilizadores só realizam transações com endereços gerados na rede, em vez da sua identidade real. No entanto, já foi demonstrado que a blockchain não pode garantir a privacidade transacional, pois os valores de todas as transações e saldos de cada chave pública são visíveis por todos os utilizadores (Meiklejohn et al., 2013; Kosba et al., 2016; Zheng et. al, 2018). Sendo a privacidade transacional algo que muitas organizações anseiam, pois não querem revelar as transações que realizam em publico, nem aos seus concorrentes, este torna-se um grande desafio a superar.

Segundo Yli-Huumo et. al (2016), a *blockchain* existe uma probabilidade de que 51% dos blocos da blockchain possa ser atacada. Uma entidade apenas poderia ter o controlo da maioria da rede, caso decidisse atacar uma *blockchain*, ou seja, teria o poder de controlar a blockchain, dando-lhe

liberdade para fazer o que quisesse e também roubar as outras entidades que faziam parte da rede. E uma vez que a *blockchain* é usada cada vez mais para efetuar transações entre entidades, esta falta de segurança pode ter um grande impacto na economia, pois caso a *blockchain* fosse atacada, uma grande quantidade de transações seria desperdiçada, roubada.

O desperdício de recursos é um grande desafio da *blockchain*, pois a mineração da rede gasta uma grande quantidade de energia, na ordem de alguns milhões de dólares/dia; para aumentar a sua eficiência é, portanto, necessário diminuir o desperdício de recursos (Yli-Huumo et. al 2016).

4.1.1.2. Cloud & Edge Computing

Conceito

O Instituto Nacional de Padrões e Tecnologia dos Estados Unidos da América (NIST) define *cloud computing* como um modelo que permite o acesso necessário e conforme a procura de rede, a um conjunto de recursos de computação que podem ser configurados e partilhados (como por exemplo, redes, servidores, armazenamento, aplicações e serviços). Estes recursos podem ser rapidamente guardados e disponíveis com o mínimo esforço de gestão ou interação com o provedor de serviços (Deepak Puthal, B. P. S. Sahoo, Sambit Mishra, Satyabrata Swain, 2015).

Isto é, podemos ter acesso a vários serviços e recursos através da *cloud* sem ter de estar a contactar diretamente com o provedor do serviço, e esses serviços e recursos estão disponíveis a qualquer altura em qualquer momento.

A computação em *cloud* fornece um ambiente online adaptável que tem a capacidade de lidar com um grande volume de trabalho sem afetar a execução da estrutura (Deepak Puthal, et. al, 2015).

Chou (2015) define computação em *cloud* como uma operação de terceirização de serviços de TI internos para fornecedores externos para atender às necessidades da empresa, ou seja, é uma TI que utiliza a abordagem de virtualização de recursos para fornecer serviços de TI através da Internet e segundo um modelo de procura.

O conceito de *cloud computing* foi evoluindo para *edge computing*. Esta evolução deve-se ao aumento da produção de dados, sendo a quantidade tão grande que a *cloud* sozinha não consegue lidar com eles e daí surgir o conceito de *edge computing*.

Edge computing refere as tecnologias que permitem que a computação seja realizada na borda da rede, em dados *downstream* por parte da *cloud* e dados *upstream* por parte de serviços de Internet of Things (IoT). Sendo "borda" definida como qualquer recurso de computação e de rede ao longo do caminho entre as fontes de dados e os *data centers* na *cloud* (Weisong Shi, Jie Cao, Quan Zhang,

Youhuizi Li, Lanyu Xu, 2016). O processamento de dados, em vez de acontecer todo na *cloud*, parte dele ocorre na borda da rede (Weisong Shi, Schahram Dustdar, 2016), o que diminui o tempo de resposta, o processamento torna-se mais eficiente e há menos pressão na rede.

Vantagens e Desafios

A transferência de aplicações e serviços para a *cloud* traz vantagens tais como: escalabilidade, redução de custos, aumento da agilidade, flexibilidade e sustentabilidade (Deepack Puthal et. al, 2015; Chou, 2014).

- **Escalabilidade:** por norma, aplicações que se encontram na *cloud*, são altamente escaláveis. É possível adicionar ou remover manualmente recursos ou a aplicação pode ser configurada para se dimensionar automaticamente e se ajustar, por exemplo, à procura de clientes. (Deepack Puthal et. al, 2015).
- **Redução de Custos:** existe uma redução de custos de TI, uma vez que só se paga pelo espaço de memória da *cloud* que se usa e não é preciso investir em hardware, software, rede e até a contratação de uma equipa de TI nas organizações (Chou, 2014).
- **Aumento da agilidade/ flexibilidade:** a *cloud* permite o acesso ao software em qualquer lugar a qualquer hora, desde que se tenha acesso à Internet (Chou, 2014).
- **Sustentabilidade:** transferir aplicações para *cloud*, significa a redução de custos com hardware e dispositivos de rede. Significa também que o uso de eletricidade nas organizações será diminuído, assim como de lixo eletrónico, contribuindo desta forma para um ambiente mais sustentável (Chou, 2014).

A *Cloud Computing* também acarreta desafios que é preciso contornar e ultrapassar. Estes desafios passam por: segurança, privacidade e integração com sistemas internos (Chou, 2014; Deepack Puthal et. al, 2015).

- **Segurança:** este torna-se o desafio mais preocupante. A partilha de dados preocupa as organizações, daí o fornecedor da *cloud* ter de garantir que os dados se encontram seguros e podem ser controlados pela organização (Chou, 2014; Deepack Puthal et. al, 2015).
- **Privacidade:** uma vez que os dados não se encontram armazenados dentro da organização e devem passar pela estrutura da internet, isto levanta preocupações em relação à privacidade, pois caso os dados sejam públicos, a privacidade das pessoas

fica em causa (Chou, 2014).

- **Integração com sistemas internos:** este desafio preocupa as organizações uma vez que a mesma possui sistemas internos que deseja que sejam integrados na *cloud*; no entanto, podem existir incompatibilidades entre os sistemas internos e a *cloud*, dificultando assim a fluência de dados entre aplicações (Chou, 2014).

4.1.1.3. Digital Twins

Conceito

A primeira definição de **Digital Twins** foi apresentada em 2002 por Michael Grieves no contexto de uma apresentação do setor referente à gestão do ciclo de vida do produto. Foi descrito como “um construto informacional digital sobre um sistema físico, criado como uma entidade por si só e vinculado ao sistema físico em questão. A representação digital deve incluir de maneira ideal todas as informações relativas ao ativo do sistema que possam ser potencialmente obtidas a partir de uma inspeção minuciosa no mundo real” (Werner Kritzinger, Matthias Karner, Georg Traar, Jan Henjes, Wilfried Sihm, 2018).

Digital Twins refere a criação de modelos virtuais para objetos e processos reais/físicos de modo a recriar o seu comportamento e simulá-lo em diferentes cenários (Qinglin Qi, Fei Tao, 2018). Isto é, os modelos virtuais simulam o comportamento de objetos e processos reais, a maneira como os mesmos se comportam em diferentes contextos reais. Daí o seu nome “*Digital Twin*” derivar do facto que o modelo virtual se comporta como um “gémeo” do objeto ou processo real/físico.

Através desta tecnologia é possível simular vários cenários, explorando os dados em tempo real de modo a desenvolver um sistema que trabalhe no seu melhor, na sua versão otimizada (Werner Kritzinger, et. al, 2018). Isto é possível devido ao facto de cada vez mais produtos são dotados de tecnologias de informação, comunicação e sensores. Os produtos são dotados de sensores que podem captar e produzir dados que depois são usados para criar o gémeo no sistema. Em conjunto com a capacidade de processar e comunicar esses mesmos dados, é possível criar “gémeos” virtuais (Sebastian Haag, Reiner Anderl, 2018).

Vantagens e Desafios

A principal vantagem que advém desta tecnologia é a sua capacidade de oferecer uma oportunidade de simular e otimizar o processos e produtos. Com esta tecnologia as empresas de manufatura podem virtualizar, estudar e manipular os seus processos logísticos e produtivos (Werner Kritzinger et. al, 2018).

Digital Twin pode ajudar a garantir a continuidade das informações durante todo o ciclo de vida do produto e assim otimizar o produto e o processo no qual está inserido. Esta continuidade de informações começa desde o comissionamento virtual de sistemas (de fabricação) e suporte a decisões e previsões de comportamento do sistema na fase de desenvolvimento do produto, bem como todas as fases subsequentes do ciclo de vida com base em simulações auxiliadas por computador (Sebastian Haag, Reiner Anderl, 2018).

Outra das vantagens desta tecnologia é a possibilidade de simular as condições de operação do produto em diferentes ambientes. Como resultado, pode confirmar quais os efeitos dos diferentes parâmetros ambientais e comportamentos de operação teriam sobre a vida útil e o desempenho, etc., de modo a controlar o estado e os comportamentos do produto físico (Qinglin Qi, Fei Tao, 2018).

Sendo uma tecnologia recente, ainda não foi explorada como outras tecnologias, sendo escassa a informação sobre a mesma. Através da revisão de literatura realizada foi identificado um desafio chave referente a esta tecnologia: a integração de dados. Embora existam muitos estudos sobre integração de dados, poucos deles foram realizados no contexto de *Digital Twins*, sendo por isso uma área a explorar (Fei Tao, Ang Liu, A.Y.C.Nee, 2019). No entanto, a integração de dados é um elemento chave no contexto de *Digital Twins*, uma vez que esta tecnologia processa uma grande quantidade de dados, de vários canais tais como máquinas, ambientes físicos, espaços virtuais e bases de dados históricas.

Relativamente a este desafio, os autores consideram que é necessário reduzir a dimensionalidade dos dados massivos, para que seja possível integrar os vários canais com a tecnologia (Fei Tao, Ang Liu, A.Y.C.Nee, 2019).

4.1.1.4. Internet of Things

Conceito

Existem diversas definições na literatura relativamente ao conceito de **Internet of Things (IoT)**. Por exemplo Kevin Ashton define IoT como “uma grade na qual coisas do mundo físico podem ser vinculadas à Internet através de sensores” (Emrah Irmak, Mehmet Bozdal, 2018).

Já Mamdakam et al. (2015) descreve IoT como uma rede aberta e abrangente de objetos inteligentes que são capazes de se auto-organizar, de partilhar dados. Também define estes objetos inteligentes como algo com capacidade de reagir a qualquer mudança que ocorra no nosso ambiente (Emrah Irmak, Mehmet Bozdal, 2018). Estas redes permitem desenvolver serviços inteligentes em domínios tão diversos como na administração da cidade, saúde, agricultura, produção, cadeias de abastecimento, educação entre outros (Emrah Irmak, Mehmet Bozdal, 2018).

De uma forma geral, podemos descrever IoT como a interconexão em rede de objetos do cotidiano, que geralmente são equipados com inteligência capaz de comunicarem de forma natural e intuitiva, em que cada dispositivo é capaz de compreender as características específicas de todos os elementos, respondendo às suas necessidades. Através da IoT a onnipresença da Internet aumentará, integrando todos os objetos que interagirão através de sistemas embutidos, o que leva a uma rede altamente distribuída de dispositivos que comunicam com seres humanos e também com outros dispositivos (Feng Xia, Laurence T. Yang, Lizhe Wang, Alexey Vinel, 2012).

Vantagens e Desafios

Estas redes de dispositivos comunicando entre si através da internet e com humanos traz vantagens, mas também desafios.

Uma das vantagens que podemos obter de IoT é o facto de o sistema processar dados e tomar decisões com base no resultado do processo. Ou seja, todas as ações podem ser realizadas automaticamente sem intervenção humana. Podemos ilustrar isto com um exemplo. No campus da universidade existem diversos edifícios (cantina, biblioteca, gabinetes...), e quase todos os edifícios têm seu próprio sistema de aquecimento, ventilação, sistemas de ar condicionado e sistema de elevador. Todos estes dispositivos devem ser geridos e mantidos, o que não é fácil. No entanto, é possível usar IoT para ajudar a gestão das instalações do campus, implementando um número suficiente de etiquetas RFID no edifício, que podem monitorar o comportamento dos sistemas de ar condicionado e dos elevadores, recolher informações, perceber a mudança do seu ambiente e localização. Como existe sinal Wi-Fi em todo o campus, os dados recolhidos de cada sistema podem ser transmitidos via Wi-Fi. Depois, o sistema processa os dados recolhidos e toma decisões sobre os mesmos; desta forma, por exemplo, pode ser feita uma gestão mais eficiente da energia. Além disso, analisando os dados, podemos conhecer a situação de manutenção das instalações e, se houver problemas, poderíamos realizar algumas ações específicas com antecedência para evitar danos (Lu Tan, Neng Wang, 2010).

Outras aplicações da IoT permitem uma visibilidade mais precisa e em tempo real do fluxo de materiais e produtos, melhorar o rastreamento de materiais, otimizar os custos de distribuição, prever receitas futuras levando a uma diminuição dos custos e um aumento da produtividade (In Lee, Kyoochun Lee, 2015).

Os dados recolhidos pelos dispositivos podem ser usados para descobrir e resolver problemas de negócios - como mudanças no comportamento do cliente e nas condições de mercado - para aumentar a satisfação do cliente e fornecer serviços de valor agregado aos clientes (In Lee, Kyoochun Lee, 2015).

Mas como qualquer tecnologia, a IoT tem os seus desafios. Estes passam por gestão de dados,

data mining, privacidade, segurança, interoperabilidade e legislação (In Lee, Kyoochun Lee, 2015; Emrah Irmak, Mehmet Bozdal, 2018).

- **Gestão de dados:** um dos desafios é que os sensores e os dispositivos de IoT estão a gerar grandes quantidades de dados que precisam ser processados e armazenados. Mas a arquitetura atual do data center não está preparada para lidar com a natureza heterogénea bem como com o grande volume de dados pessoais e empresariais recolhidos pelos dispositivos e sensores (In Lee, Kyoochun Lee, 2015).
- **Data Mining:** à medida que mais dados estão disponíveis para processamento e análise, a utilização de ferramentas de *data mining* torna-se uma necessidade; isto é, para a quantidade de dados que está cada vez mais a ser produzida por redes de IoT as ferramentas existentes de *data mining* não são suficientes, são precisas mais (In Lee, Kyoochun Lee, 2015).
- **Privacidade:** os dispositivos de IoT fornecem uma grande quantidade de dados sobre a localização e movimentos dos utilizadores de IoT, como por exemplo condições de saúde e preferências de compra, os quais podem desencadear importantes preocupações com a privacidade, uma vez que se tais dados sejam do conhecimento público as pessoas iriam ter a sua privacidade violada (In Lee, Kyoochun Lee, 2015; Emrah Irmak, Mehmet Bozdal, 2018).
- **Segurança:** a ameaça à segurança vai aumentando conforme os dispositivos adicionados vão aumentando. Qualquer tecnologia com uma segurança ineficiente é alvo de ciberataques e roubos, o que é bastante perigoso quando falamos de IoT uma vez que há sensores e dispositivos que recolhem informação privada das pessoas (In Lee, Kyoochun Lee, 2015; Emrah Irmak, Mehmet Bozdal, 2018).
- **Interoperabilidade:** esta é definida como a capacidade de um produto, serviço, sistema para interagir, comunicar, trocar informações e funcionar conjuntamente. Ainda não foi definido um padrão de comunicação entre dispositivos de IoT, ou seja, os dispositivos não têm um protocolo de comunicação entre eles, o que dificulta o seu funcionamento, tornando-se assim num desafio (Emrah Irmak, Mehmet Bozdal, 2018).
- **Leis:** do ponto de vista dos governos este poderá ser o maior desafio. À medida que cada país, cada governo adota tecnologia IoT, os países vão desenvolvendo os seus próprios regulamentos, as suas próprias leis, o que criará uma

preocupação mundial pela diversidade de infraestruturas de IoT que precisam ser auditadas e regulamentadas. É necessário criar legislações que sejam partilhadas para todos os governos (Emrah Irmak, Mehmet Bozdal, 2018).

É possível concluir que além dos benefícios que todos podemos tirar desta tecnologia, ainda existem muitos desafios que é preciso ultrapassar.

4.1.1.5. Robotic Process Automation

Conceito

Em 2014, o Instituto de *Robotic Process Automation* (IRPA) definiu *robotic process automation* (RPA) como “o uso da tecnologia que permite aos funcionários de uma empresa configurar software ou robôs, para capturar e interpretar aplicações existentes para processar transações, manipular dados e comunicar com outros sistemas digitais”.

Já o artigo da *Financial Express* (2016) definiu RPA como um conjunto de ferramentas de software automatizadas usadas pelas empresas para processamento repetido de tarefas complexas (*low-end*) sem nenhum envolvimento humano.

É também definida por Sutherland (2013) como a aplicação de tecnologias e metodologias específicas para usar um computador ou "robô virtualizado", em vez de uma pessoa para manipular o software de aplicações existentes. Ou seja, o software apenas substitui a pessoa responsável pela execução da tarefa.

A RPA tem como objetivo substituir as pessoas pela automação através da abordagem '*outside-in*', isto é, os sistemas de informação existentes permanecem inalterados. Em vez de se redesenhar o sistema, apenas os humanos são substituídos por agentes, o resto do processo permanece igual (Will M. P. van der Aalst, Martin Bichler, Armin Heinzl, 2018).

Esta tecnologia tem-se tornado numa necessidade para as atividades de negócios do dia-a-dia dada a importância de manter vantagem competitiva através da inovação dos processos e manutenção da eficiência dos processos produtivos (Somayya Madakam, Rajesh M. Holmukhe, Durgesh Kumar Jaiswal, 2019).

Vantagens e Desafios

Uma das vantagens da automação de processos passa pelo facto da integração de inteligência artificial e robots agregar valor imediato aos principais processos de negócio, isto inclui: folha de pagamento dos funcionários, recrutamento, processamento de faturas, gestão de inventários, criação de

relatórios, instalações de software, migração de dados, entre outros (Somayya Madakam et. al, 2019).

Uma vez que a automação de processos significa menor necessidade de funcionários para executar as tarefas, torna a execução de tarefas mais rápidas e fiáveis, libertando os seres humanos do trabalho e da monotonia das tarefas repetitivas. Esta automatização das tarefas e dos processos leva a uma maior produtividade e eficiência nas organizações, redução de custos, minimização de erros humanos e eliminação de riscos (Somayya Madakam et. al, 2019; Dahlia Fernandez, Aini Aman, 2018).

Existem dois principais desafios associados à automação de processos: o elevado custo e o medo por parte dos funcionários em serem substituídos (Will M. P. van der Aalst et. al, 2018; Dahlia Fernandez, Aini Aman, 2018).

Devido ao facto desta tecnologia substituir a mão de obra humana em muitas tarefas, os funcionários têm um certo receio de serem substituídos e ficarem sem trabalho; por esta razão as organizações têm de lidar com este desafio, esta resistência em relação à automação.

5. Metodologia de Trabalho

5.1. Objetivos

Este trabalho insere-se num projeto europeu, projeto VOIL que tem por objetivo desenvolver um laboratório virtual para apoiar as PME's no seu esforço de transformação digital. Sendo que a Universidade do Minho é uma das 8 organizações do consórcio, o objetivo deste trabalho de dissertação é apoiar as tarefas a cargo da Universidade do Minho, em particular:

- Desenvolvimento do “toolkit” de transformação digital das PME's;
- Curriculum para desenvolvimento de competências em transformação digital das PME's;
- Recursos de Simulação a disponibilizar às PME's.

5.2. Projeto VOIL

O projeto VOIL - Virtual Open Innovation Lab tem dois objetivos principais:

- Sensibilizar para o potencial das tecnologias emergentes para a transformação digital das pequenas e médias empresas;
- Criar um modelo pedagógico para simular competências digitais avançadas.

Para conseguir alcançar estes objetivos, este projeto propõe-se a desenvolver 5 outputs. Sendo que o primeiro output será um kit de ferramentas de transformação digital que fornece o conhecimento sobre as melhores práticas que surgiram na indústria em relação ao design de estratégias de transformação digital e à cocriação de soluções inovadoras.

O segundo output que este projeto se propõe a desenvolver é um currículo de transformação digital, que orientará educadores/professores que desejam desenvolver programas de formação com foco no desenvolvimento de competências digitais avançadas.

Quando ao terceiro output - Simulation Resource Pack – este incluirá notas de aprendizagem que orientarão a experimentação de tecnologias emergentes e/ou impactos da transformação digital através da utilização de simuladores.

A plataforma VOIL será o quarto output que corresponde ao ambiente de aprendizagem baseado em simulação que permitirá o desenvolvimento das competências digitais avançadas.

Já o último output será um relatório de implementação e validação. Este relatório será um output essencial, já que contém a experiência dos parceiros e a memória do projeto. Estas informações

apoiarão o uso da plataforma VOIL, bem como desenvolvimentos futuros.

Tal como referido anteriormente, este projeto é realizado em parceria com 8 organizações, sendo que cada um destes outputs serão desenvolvidos por uma ou mais organizações.

5.3. Método

O método de trabalho está condicionado pelas atividades a implementar no projeto VOIL, uma vez que este trabalho se insere no projeto.

Inicialmente foi feita uma revisão de literatura em relação aos vários temas abordados neste trabalho, com recurso a bases de dados como o Google Scholar (<https://scholar.google.com/>), Scopus (www.scopus.com). Foram também estabelecidos limites de datas de publicação. Como critério de seleção foram usados os termos “digital transformation”, “emerging technologies”, “simulation”, “simulation-based learning”, “digital twins”, “internet of things”, “blockchain”, “robotic process automation”, “cloud computing”. Foram encontrados 99 artigos e da análise do resumo, introdução e conclusões resultou que 49 desses artigos eram relevantes para o trabalho a realizar. Estes artigos serviram de referência teórica a este trabalho.

Lendo o resumo, procurou-se encontrar conceitos relacionados com o que se pretendia, caso isso não se verificasse, o artigo seria descartado. Caso contrário, prosseguia-se para a introdução onde seriam analisados os principais objetivos a discutir no artigo e se algum deles era do nosso interesse. Caso o mesmo não se confirmasse, o artigo deixava de ser relevante para este trabalho. Se fosse relevante, lia-se a conclusão para ter uma noção das conclusões a que chegaram os autores e se iam de encontro ao que era esperado ou não.

Para apoiar o desenvolvimento do ambiente virtual do projeto VOIL fez-se uma análise de 44 simuladores (*Open Source* e Comerciais) de tecnologias emergentes. Esta lista de simuladores resultou de uma dissertação anterior (“Avaliação do Potencial Transformador das Tecnologias Emergentes” de Ângela Gil) bem como pesquisas adicionais realizadas em websites e em artigos científicos.

A análise destes simuladores foi realizada com o auxílio de alguns alunos de mestrado, que ajudaram na instalação e na exploração dos mesmos.

Os 44 simuladores foram instalados e posteriormente explorados de modo a ser possível fazer uma lista de especificações técnicas e funcionalidades relativamente a cada um. Com base nas tabelas (ver Anexos Simuladores) elaboradas para cada simulador, foi feita uma avaliação tendo por base critérios relativamente a quatro categorias: estratégia de transformação digital, competências tecnológicas, uso e problemas técnicos.

Para cada uma destas categorias foram elaboradas questões às quais os simuladores teriam de responder para se qualificarem para as várias fases (ver Tabela 2). Foi através destas questões que se

foram eliminando os simuladores até a uma lista final de 4 simuladores.

Problemas Técnicos	Estratégia de Transformação Digital	Competências Tecnológicas	Uso do Simulador
O simulador é fácil de instalar?	O simulador é útil para avaliar as oportunidades de crescimento promovidas pelas tecnologias digitais?	O simulador permite avaliar o potencial de negócios da tecnologia?	O simulador é fácil de usar?
É fácil fornecer as informações para executar a simulação?	O simulador promove a autonomia dos funcionários para lançar e implementar iniciativas de digitalização?	O simulador permite identificar o potencial da tecnologia para criar valor comercial?	As informações fornecidas são esclarecedoras?
O sistema é confiável na maneira como funciona? (não crasha)	O simulador suporta a revisão de como as tecnologias digitais podem melhorar a proposta de valor da empresa, a criação e/ou entrega de valor?	O simulador permite testar novos projetos de processos?	O sistema é adequado para aprender sobre a tecnologia que simula?
	O simulador permite avaliar como a tecnologia garante a segurança nos processos de negócios, dados e clientes contra possíveis ataques externos e/ou uso inadequado dos dados?	O simulador permite a avaliação de riscos digitais, incluindo os associados à segurança e confiabilidade das informações?	A simulação fornece suporte online ao utilizador?
			Há informações suficientes na Web (tutoriais, vídeos, manuais do usuário) para apoiar o aprendizado dos recursos do simulador?

Tabela 2 - Critérios de Seleção

Inicialmente, o critério com maior peso na avaliação foi o de problemas técnicos, o qual

permitiu descartar 17 simuladores, reduzindo a lista de 44 simuladores para 27 simuladores. Estes 17 simuladores apresentaram dificuldades significativas de instalação: ou foi impossível instalar, ou a sua instalação era bastante complexa e/ou houve erros que não puderam ser resolvidos em tempo útil.

A avaliação dos 27 simuladores continuou em maior profundidade incidindo sobre as outras categorias. Foram definidos os objetivos gerais pretendidos para cada tecnologia e foram elaboradas 5 tabelas para cada uma das tecnologias. Nessas tabelas foram postos os vários simuladores e foram comparadas funções. Deste modo foi possível perceber quais os simuladores que eram mais completos e que iam de encontro aos objetivos pretendidos para cada uma das tecnologias, assim como tendo em conta a aprendizagem nas PMEs.

Concluída esta análise chegou-se a uma lista final de 4 simuladores, sendo para estes desenvolvidos guias de aprendizagem que incluem: objetivos de aprendizagem, cenários de aplicação, autoavaliação, diretrizes de discussão e avaliação dos resultados de aprendizagem.

Para a análise dos simuladores selecionados, utilizou-se uma abordagem de definição de uma estratégia de transformação digital bem como o curriculum para desenvolvimento de competências digitais avançadas, ambos desenvolvidos no projeto VOIL.

6. Resultados

Depois da pesquisa de simuladores (*open source* e comerciais) referentes a cada tecnologia foram identificados 44 simuladores no total. Cada um dos simuladores, foi instalado e explorado, sendo elaborada uma lista de funcionalidades relativamente aos simuladores em que a instalação foi possível.

Não foi possível a instalação dos seguintes 4 simuladores: GroudSim, WSNet, SenseSim, SimCad. Quanto ao GroudSim. A sua instalação foi impossível pois requeria alguns ficheiros que já não estavam disponíveis para download. Enquanto o WSNet não foi encontrado nenhuma versão para fazer download assim como o SenseSim. Relativamente ao SimCad, uma vez que é um simulador comercial e não tem uma versão teste, foi preciso contactar o fornecedor para uma versão free, no entanto, não houve resposta por parte do mesmo.

Retirando os 4 simuladores que não foi possível instalar, resultaram 40 simuladores que foram instalados. No entanto houve dificuldades técnicas na instalação de 7 simuladores: ENERPORT, IBM Hyperledger, EMUSIM, USARSim, UberSim, SimRobot, TOSSIM, não se procedendo à sua completa instalação por isso não foi possível a sua exploração.

Isto resultou numa lista de 33 simuladores instalados e prontos a serem explorados.

Para cada um dos 33 simuladores foi elaborada uma tabela com o nome do simulador, o tipo de simulador (*open source* ou comercial), uma breve descrição do simulador, incluindo o seu principal objetivo, especificações técnicas (guia de instalação incluído), funcionalidades e para alguns foram desenvolvidos objetivos de aprendizagem (ver Anexo).

Pode observar-se, no entanto que para alguns dos simuladores a informação sobre os mesmos era escassa e alguns bastante complicados de manusear, o que dificultou o processo de exploração dos mesmos.

6.1. Simuladores Blockchain

Simulador	Tipo	Descrição
Enerport	Open Source	É um projeto lançado na Conferência Anual do Centro Internacional de Investigação Energética (IERC) e tem como objetivo acelerar o comércio de energia peer-to-peer na Irlanda, usando a tecnologia Blockchain (IERC, 2018). Este projeto pretende permitir que os consumidores irlandeses possam oferecer ou vender a energia solar que produzam aos seus vizinhos, além de partilhar e gerir a sua própria eletricidade (IERC, 2018).

IBM Hyperledger	Open Source	<p>O IBM Hyperledger é um simulador Open Source que permite a realização de smart contracts (contratos inteligentes) em várias linguagens. É um simulador constituído por código aberto (IBM, n.d.).</p> <p>É um testnet que oferece um desempenho em grande escala, mantendo a privacidade (Krown, n.d.).</p>
Multichain	Open Source	<p>O Multichain é um simulador Open Source e o seu objetivo principal é apoiar as organizações a desenvolver e testar uma aplicação de blockchain com custos reduzidos. Destina-se principalmente para utilizadores que pretendam desenvolver plataformas de blockchain no sector financeiro (Cachin & Vukolic, 2017).</p>
Bitcoin Testnet	Open Source	<p>É um simulador open source em que o objetivo principal é testar uma aplicação da tecnologia blockchain para estudar o seu comportamento antes da implementação. As moedas de teste são separadas e distintas dos bitcoins reais e não têm qualquer valor, permitindo que sejam feitas experiências sem usar bitcoins reais (Krown, n.d.).</p>
Ethereum Testnet	Open Source	<p>O Ethereum Test Net é um simulador open source que fornece uma rede de teste. O objetivo principal deste simulador é oferecer um ambiente seguro para testar uma aplicação de blockchain de modo a garantir que quando implementada, a aplicação funciona conforme o esperado (Krown, n.d.).</p>
SIMBlock	Open Source	<p>É um simulador de rede blockchain de código aberto, desenvolvido pelo Distributed Systems Group, Instituto de Tecnologia de Tóquio. É orientado a eventos e é usado em projetos de investigação focada na tecnologia blockchain. O principal objetivo deste simulador é manipular uma rede blockchain constituída por um elevado número de nós de maneira a conseguir determinar o impacto das alterações introduzidas no desempenho e na segurança da blockchain.</p>
The Principle of Blockchain	Open Source	<p>Este simulador surgiu num contexto académico e tem por objetivo principal do simulador facilitar a aprendizagem da tecnologia blockchain ao permitir que pessoas com pouca experiência possam aprender.</p>
Elemental Battles	Open Source	<p>O Elemental Battles é um jogo tutorial de codificação online que ensina ao utilizador o básico para construir a base de código EOSIO, um software que introduz uma arquitetura blockchain projetada para permitir o dimensionamento vertical e horizontal de aplicações descentralizadas; esta arquitetura pode ser usada para lançar redes</p>

		blockchain privadas e públicas. Ensina ainda a definir contratos inteligentes (EOS, n.d.-a).
--	--	--

Tabela 3 - Simuladores Blockchain

Não foi possível a conclusão da instalação de 2 dos simuladores (Enerport, IBM Hyperledger) devido a complicações técnicas que não foi possível resolver. Outro simulador (The Principle of Blockchain), chegou-se à conclusão que seria difícil manusear, incluindo a sua instalação. Ou seja, restaram assim 5 simuladores referentes à tecnologia blockchain.

Foi elaborada uma tabela com os 5 simuladores de modo a ser possível observar as funções comuns entre os mesmos:

F1: Manipular uma *blockchain*: criação e operações numa cadeia de blocos

F2: Integrar com outras soluções: experimentar a integração com outras aplicações de TI

F3: Gerir carteiras de e-coins: experimentar transações digitais

F4: Gerir *smart contracts*: experimentar transações regulamentadas

	F1	F2	F3	F4
Multichain (OS)	X	X	X	
Bitcoin Testnet (OS)			X	X
Ethereum Testnet (OS)	X			X
SIMBlock (OS)	X			
Elemental Battles (OS)	X			X

Tabela 4 – Simuladores Finais Blockchain

Através desta tabela podemos observar que apenas o simulador Bitcoin Testnet não tem a função de manipular uma blockchain. A nível das funções, o Multichain é considerado o mais completo dos 5 simuladores, uma vez que é único que apresenta 3 das 4 funções e pode integração de outras soluções.

6.2. Simuladores Cloud & Edge Computing

Simulador	Tipo	Descrição
Cloud Sim	Open Source	É uma ferramenta de modelação e simulação de EdgeCloud desenvolvida na Universidade de Melbourne, na Austrália. Suporta a modelação e simulação de ambientes de EdgeCloud Computing em larga escala, incluindo datacenters, nós de computação, máquinas virtuais, provisionamento de recursos e provisionamento de máquinas virtuais e gestão de energia (Mostinckx et al., 2009). O objetivo principal é a realização de testes robustos com base em configurações personalizadas e implementação aplicações para Cloud.
Cloud Analyst	Open Source	É um simulador open source derivado do CloudSim. O seu objetivo principal é determinar o comportamento de aplicações de Internet de grande escala na Cloud. Permite realizar várias simulações com pequenas variações de parâmetros (Nayyar, 2016).
Edge Cloud Sim	Open Source	Este simulador é open source de EdgeCloud Computing. O objetivo principal é fornecer um ambiente de simulação onde é possível realizar experiências que tenham em conta os recursos computacionais e de rede. É uma extensão do simulador CloudSim, mas adiciona novas funcionalidades.
iCanCloud	Open Source	É uma plataforma de simulação destinada a modelar e simular sistemas de Cloud Computing (Arcos,n.d.). O principal objetivo é determinar o compromisso entre os custos incorridos e a taxa de desempenho durante as simulações. Este compromisso é referente a um determinado conjunto de aplicações executadas num hardware específico. De seguida, fornece aos utilizadores informações sobre esses custos.
EMUSIM	Open Source	Este simulador combina emulação e simulação, utilizando o simulador CloudSim. Esta combinação permite extrair informações automaticamente do comportamento da aplicação (via emulação) e usa essas informações para gerar o modelo de simulação correspondente, de modo a ajudar os desenvolvedores a obter modelos mais precisos e estimar o desempenho e custo da aplicação.
GROUDSIM	Open Source	O GroudSim é um simulador baseado em eventos, especialmente projetado para aplicações científicas. Esta ferramenta de simulação trabalha nas áreas de IaaS da cloud e é extensível de maneira conveniente para oferecer suporte a modelos adicionais, como armazenamento em cloud ou plataforma como serviço.

		Permite adicionar recursos durante a simulação e descrever cenários em XML, em vez de obrigar à escrita de código
GreenCloud	Open Source	É um simulador open source para Cloud Computing. O objetivo principal do simulador é oferecer uma modelação detalhada da energia consumida pelos equipamentos de TI de um datacenter (servidores de computação, computadores de rede e links de comunicação). (Greencloud, n.d.).

Tabela 5 - Simuladores Cloud&Edge Computing

Não foi possível a instalação de um dos simuladores (GroudSim), assim como não foi possível conclusão da instalação de 1 dos simuladores (EMUSIM) devido a complicações técnicas que não foi possível resolver. Ou seja, restaram assim 5 simuladores referentes à tecnologia *cloud & edge computing*.

Foi elaborada uma tabela com os 5 simuladores de modo a ser possível observar as funções comuns entre os mesmos:

F1: Modelar/Configurar arquiteturas: experimentar diferentes tipologias de redes de diversos recursos computacionais.

F2: Modelar recursos computacionais e o seu comportamento: determinar o desempenho de dispositivos, como dispositivos de borda, *datacenters* em cloud, sensores, links de rede, fluxos de dados e aplicação de processamento de fluxo.

F4: Analisar o desempenho da rede: testar o desempenho da rede sob uma carga realista e ajustável.

F5: Analisar a energia consumida: determinar o consumo de energia do ambiente configurado.

	F1	F2	F3	F4
EdgeCloudSim (OS)	X	X	X	X
CloudSim (OS)		X	X	X
iCanCloud (OS)	X			X
GreenCloud (OS)	X	X	X	X
CloudAnalyst (OS)			X	X

Tabela 6 – Simuladores Finais Cloud&Edge Computing

Através da tabela é possível constatar que os simuladores são bastante semelhantes. Isto deve-se ao facto de serem extensões de um outro simulador (CloudSim), ou seja, maior parte das funções são iguais e apenas algumas são adicionadas. No entanto podemos observar que o simulador EdgeCloudSim e o GreenCloud são os simuladores que incluem todas as funções, ou seja, são os mais completos da lista.

6.3. Simuladores Digital Twins

Simulador	Tipo	Descrição
Anylogic	Open Source	O Anylogic é uma aplicação da tecnologia Digital Twin que permite a utilização de três métodos (evento discreto, baseado em agente, dinâmica do Sistema), para simular sistemas complexos. O objetivo principal deste simulador é apoiar a decisão pela exploração de vários cenários de simulação de operações e serviços.
SIMIO	Comercial	O SIMIO é um simulador comercial que recebe dados operacionais, históricos e ambientais, sendo possível construir vários cenários para prever e gerar resultados, suportando grandes volumes de dados. O objetivo principal deste simulador é permitir a manipulação de parâmetros e fazer previsões a partir da digitalização da capacidade de produção da empresa.
SIM CAD	Comercial	É um simulador comercial dinâmico com visualização animada 2D/3D e realidade virtual. Permite que os utilizadores planeiem, otimizem e reorganizem processos e procedimentos enquanto otimizam layouts. Ou seja, é possível simular os processos produtivos, de forma a que seja possível analisá-los e melhorá-los. É também possível modificar o modelo durante a simulação e analisar o impacto das mudanças, sem a necessidade de parar ou reiniciar a simulação. Este simulador fornece uma extensa lista de ferramentas de relatório e análise, que podem ser usados no processo de tomada de decisão.

Tabela 7 - Simuladores Digital Twins

Não foi possível a instalação de um dos simuladores (SimCAD) devido a não haver uma versão grátis que pudesse ser instalada. Ou seja, restaram assim 2 simuladores referentes à tecnologia digital twins.

Foi elaborada uma tabela com os 2 simuladores de modo a ser possível observar as funções

comuns entre os mesmos:

F1: Manipular processos e objetos: representação de objetos ou processos reais com os seus dados, funções e recursos de comunicação

F2: Modelar cenários realísticos: criar conjuntos de processos/objetos interconectados em ambientes realistas

F3: Definir conexões com sistemas externos: usar dados de sistemas reais, como ERPs, para executar réplicas digitais de processos/objetos organizacionais

F4: Fazer análise de design: experimentar diferentes designs de processos e objetos interconectados

F5: Fazer análise de risco: determinar os riscos das decisões de design

	F1	F2	F3	F4	F5
Anylogic (OS)	X	X		X	
SIMIO	X	X	X	X	X

Tabela 8 - Simuladores Finais Digital Twins

A tabela anterior mostra uma grande semelhança entres os dois simuladores. O simulador SIMIO acaba por ser o simulador mais completo pois inclui todas as funcionalidades básicas deste tipo de simuladores. No entanto, existe um contra em relação ao SIMIO que é o facto de ser um simulador comercial e a sua versão grátis ter limitações significativas.

6.4. Simuladores Internet of Things

Simulador	Tipo	Descrição
TOSSIM	Open Source	O simulador TOSSIM é um projeto que inclui uma rede de sensores com grande número de pequenos dispositivos de hardware. Este simulador forma “manchas” durante a simulação através da conceitualização de funções de hardware e modelagem de software.
DPWSim	Open Source	O DPWSim é um kit de ferramentas de simulação open source que suporta o desenvolvimento de aplicações de IoT, através da utilização de Devices Profile for Web Services (DPWS), ajudando os desenvolvedores a criar protótipos. É também possível gerir, criar, armazenar e simular carga, o que oferece

		<p>uma alta flexibilidade aos utilizadores, nas simulações (Han et al., 2014). Aplicações de IOT que funcionem com dispositivos virtuais fornecidos pelo DPWSim, podem funcionar corretamente em dispositivos físicos sem qualquer alteração no código.</p>
iFogSim	Open Source	<p>É um kit de ferramentas Open Source para Cloud Computing, Edge Computing e IoT. O objetivo principal do simulador é modelar e testar redes de Edge Computing, IoT e de Fog Computing, medindo o impacto das técnicas de gestão de recursos tecnológicos em termos de latência, congestionamento de rede, consumo de energia e custo.</p>
Cooja	Open Source	<p>O Cooja é um simulador de IoT baseado em java, através do qual é possível testar o Contiki, um sistema operativo em Open Source para IoT, usado em projetos académicos, para criação de redes sem fios entre sensores, pois permite ligar à Internet microcontroladores de baixo custo e também de baixo consumo energético (PPLWare, 2019).</p>
WSNet	Open Source	<p>O WSNet é um simulador open source orientado a eventos para redes sem fio, mas que também pode ser usado para a IoT. É capaz de simular nós com diferentes fontes de energia, modelos de mobilidade, interfaces de rádio, aplicações e protocolos de roteamento. O objetivo principal do simulador é compreender o comportamento e impacto da rede em cenários reais.</p>
CupCarbon	Open Source	<p>O CupCarbon é um simulador de IoT open source que foi originalmente projetado como um simulador com forte ênfase no suporte à mobilidade de nós geográficos tendo por base ambientes do mundo real. Assim, este simulador compreender a mobilidade de nós geográficos em ambientes urbanos inteligentes para que por exemplo seja possível controlar e monitorar variáveis como o impacto ambiental do tráfego automóvel entre outras utilizações como a gestão do tráfego através de semáforos inteligentes, seleção de rotas de pedestres com base em parâmetros ambientais. O simulador utiliza agentes móveis e topologia detalhada do nível da rua com base em mapas do mundo real.</p>
SenseSim	Open Source	<p>O SenseSim foi projetado para simular a rede de sensores como um sistema autónomo e auto-organizado, incorporado no</p>

		<p>ambiente onde podem ocorrer muitos fenómenos diferentes. O objetivo principal deste simulador é compreender o comportamento e impacto da rede IoT nos processos do mundo real/externo.</p> <p>No SenseSim os dispositivos podem construir uma rede heterogénea na qual alguns deles podem ser sensores simples com recursos básicos de perceção e comunicação, mas também podem ser suportados por alguma "inteligência" fornecida por macroprogramas.</p>
OMNeT++	Open Source	<p>O OMNeT ++ é um simulador open source discreto baseado em eventos, desenvolvido em C ++ e usado para simular redes de comunicação, sistemas distribuídos e outras aplicações de multiprocessadores.</p> <p>Este simulador foi desenvolvido para preencher a lacuna entre simuladores orientados para a pesquisa e de código aberto, suportados pela comunidade, como o NS3, e simuladores comerciais caros, como o OPNET.</p>
Bevywise	Comercial	<p>É um simulador comercial que tem como objetivo principal explorar redes dinâmicas de milhares de dispositivos que ajudam a entender como vários dispositivos dotados de tecnologia IoT interagem.</p>
IOTify	Comercial	<p>É um simulador comercial de IOT que permite desenvolver soluções de IOT na Cloud, em grande escala e permite gerar tráfego personalizado a partir de milhares de endpoints virtuais e testar a plataforma em termos de escala, segurança e confiabilidade (Dean, 2017; Iotify, n.d).</p>
IOTSim	Comercial	<p>O IOTSim é um simulador comercial de IoT e é uma extensão do simulador Cloudsim. Simula aplicações de IoT usando o modelo MapReduce (modelo de programação, da Google, capaz de processar grandes volumes de dados de forma paralela), com um alto grau de precisão e com baixo custo e tempo (Zeng et al., 2017).</p> <p>Este simulador permite que os investigadores analisem a maneira como uma aplicação de IoT, compatível com MapReduce, funciona num determinado ambiente (Zeng et al., 2017). Reproduz a mecânica do centro de dados (como configurações de máquinas virtuais, requisitos computacionais e custos), em vez de interações de rede de sensores.</p>

Netsim	Comercial	O NetSim é um simulador comercial que pode ser usado para simular sistemas de IoT, de forma a testar o desempenho de aplicações reais numa rede virtual (Dean, 2017). O NetSim é usado para desenvolver protótipos de rede de computadores. Este simulador suporta várias origens e destinos e pode ser dimensionado para centenas de nós, sendo possível simular diversas situações com a ajuda dos cenários "What-if" e métricas de teste, como perda, atraso, erro, qualidade de serviço, entre outros cenários que podem ser implementados na ferramenta (Dean, 2017).
Ansys IoT Simulator	Comercial	O Ansys IoT Simulator é um simulador comercial que tem como objetivo principal a criação protótipos virtuais de sistemas completos, para criar dispositivos IoT reais. Ajuda a otimizar os protótipos, o desempenho de componentes ou subsistemas individuais, e todo o sistema de uma só vez (ANSYS, n.d.-c). Esta ferramenta pode ser usada em vários contextos incluindo <i>wearables</i> e dispositivos médicos, drones, carros, equipamentos industriais e muito mais, ajudando o utilizador a criar dispositivos mais acessíveis e lucrativos (Dean, 2017).
QualNet	Comercial	QualNet é um simulador comercial de IoT. É uma ferramenta capaz de reproduzir o comportamento de uma rede de comunicações real. O objetivo principal é avaliar o comportamento básico de uma rede e testar combinações de recursos de rede. Esta ferramenta de simulação de rede oferece um ambiente abrangente para projetar protocolos, criar e animar cenários de rede e analisar o seu desempenho.
SimpleIoTSimulator	Comercial	O SimpleIoTSimulator é um simulador comercial que permite criar ambientes de IoT constituídos em muitos sensores e gateways. Este simulador suporta protocolos comuns de IoT, como CoAP e MQTT como um protocolo baseado em publicação/assinatura. O objetivo principal deste simulador é permitir que os fornecedores de plataformas e gateway de IoT melhorem a qualidade do produto tendo particular atenção aos protocolos de comunicação.

Tabela 9 - Simuladores Internet of Things

Não foi possível a instalação de 2 dos simuladores (WSNet, SenseSim) e enquanto que outro (TOSSIM) não foi possível a conclusão da sua instalação uma vez que apareceram erros que não foram

possíveis solucionar. No caso de quatro simuladores (NetSim, Ansys IOT Simulator, QualNet, SimpleIOTSimulator), chegou-se à conclusão de que seriam difíceis de manusear pois requeriam conhecimentos de código e interfaces que não eram “*user friendly*”. Sendo também a sua instalação difícil de realizar. Ou seja, restaram assim 8 simuladores referentes à tecnologia *internet of things*.

Foi elaborada uma tabela com os 8 simuladores de modo a ser possível observar as funções comuns entre os mesmos:

F1: Manipular redes de dispositivos e sensores: criar dispositivos e sensores e conectá-los numa determinada topologia de rede.

F2: Manipular cenários geográficos: criar redes IoT geograficamente distribuídas para cenários realistas

F3: Fazer análise da energia consumida: determinar a energia necessária para operar uma rede IoT

F4: Monitorizar indicadores de desempenho chaves: rastrear as alterações ao longo da rede IoT - nós, mensagens e eventos - determinando, portanto, a eficiência da rede.

F5: Fazer análises de dados geográficos: análise da informação produzida por sensores em diferentes localizações geográficas (ex: *smart cities*), apoiando assim a conceção de ambientes inteligentes.

	F1	F2	F3	F4	F5
DPWSim (OS)	X			X	
iFogSim (OS)	X		X	X	
Cooja (OS)	X			X	
CupCarbon (OS)	X	X	X	X	X
OMNeT++ (OS)	X			X	
Bevywise	X			X	
IOTify	X	X	X	X	X
IOTSim	X			X	

Tabela 10 - Simuladores Finais Internet of Things

Através da tabela anterior é possível observar que o CupCarbon é o simulador mais completo dos 8 simuladores pois apresenta todas as funcionalidades básicas. Também é possível ver que todos os simuladores permitem a manipulação de redes e a monitorização dos indicadores de desempenho.

A partir desta análise, foi decidido que o CupCarbon passaria à lista final dos 4 simuladores.

6.5. Simuladores Robotic Process Automation

Simulador	Tipo	Descrição
Gazebo	Open Source	O Gazebo é um simulador open source para robótica que permite testar rapidamente algoritmos, projetar robôs, executar testes de regressão e treinar o sistema de inteligência artificial usando cenários realistas. Em suma, o objetivo principal do simulador é projetar e testar robôs, usando cenários realistas.
USARSim	Open Source	O USARSim é um simulador open source de alta fidelidade de robôs e ambientes baseados no jogo do Unreal Tournament. O objetivo principal deste simulador é testar e compreender situações de interação humano-robot e coordenação multi-robot.
ARGoS	Open Source	ARGoS é um simulador open source de robô que opera em mais de um campo físico em simultâneo. É capaz de simular eficientemente conjuntos de robôs de qualquer tipo, em grande escala. Tem plug-ins que podem ser adicionados de modo a ser possível personalizar. Tem a possibilidade de usar vários mecanismos de física de diferentes tipos e atribuí-los a diferentes partes do ambiente. Os robôs podem migrar de um mecanismo para outro de forma transparente. O objetivo principal deste simulador é a análise da interação entre robots.
UberSim	Open Source	O UberSim é um simulador open source baseado no mecanismo de física ODE e usa o OpenGL para gráficos. Foi criado especificamente com foco em pequenos robôs numa simulação de futebol de robôs. O objetivo principal deste simulador é explorar a interação multi-robot.
EyeSim	Open Source	O simulador EyeBot usa realidade virtual EyeSim VR. É um simulador de múltiplos robôs móveis com funcionalidade VR baseada no mecanismo de jogo Unity 3D que permite fazer experiências com os mesmos programas que são executados nos robôs reais, sendo que os ambientes de teste podem ser criados facilmente carregando arquivos de texto. O objetivo principal é testar o software de robôs.
SimRobot	Open Source	O SimRobot é um simulador open source de robótica. Utiliza o ODE (Open Dynamics Engine) para simulações de física e o OpenGL para gráficos. O objetivo principal do simulador é ser usado para simulações de RoboCup.

WeBots	Open Source	Webots é um simulador open source que fornece um ambiente de desenvolvimento completo para modelar, programar e simular robôs. Portanto, o objetivo principal deste simulador é experimentar com robôs virtuais de forma a otimizar a construção de robôs e interações multi-robot.
HEEDS	Comercial	O HEEDS tem como objetivo principal fornecer um ambiente que permita obter o melhoramento de projetos de simulação, de forma mais rápida e eficaz. Lida com os desafios automatizando o processo de análise, impulsionando o investimento em hardware de computação e procurando, eficientemente, soluções com melhor desempenho, fornecendo assim maneiras intuitivas de corrigir o desempenho do projeto. Permite definir os resultados desejados para o projeto de simulação e este, tendo em conta os parâmetros determinados, executa várias tarefas automaticamente, de forma a encontrar soluções, num curto prazo de tempo (MathWorks, n.d.).
SimCAD	Comercial	É um simulador comercial dinâmico com visualização animada 2D/3D e realidade virtual. Permite que os utilizadores planeiem, otimizem e reorganizem processos e procedimentos enquanto otimizam layouts (Capterra, n.d.). Ou seja, é possível simular os processos produtivos, de forma a que seja melhor analisá-los e melhorá-los. É também possível modificar o modelo durante a simulação e analisar o impacto das mudanças, sem a necessidade de parar ou reiniciar a simulação. Este simulador fornece uma extensa lista de ferramentas de relatório e análise, que podem ser usados no processo de tomada de decisão (Createasoft, n.d.).
ANSYS ACT	Comercial	O ANSYS ACT é um simulador comercial que permite criar aplicações para automatizar os fluxos de trabalho rotineiros, personalizar processos especializados, desenvolver aplicações específicas para negócios e integrar ferramentas e dados de terceiros à linha de produtos ANSYS. (ANSYS, n.d.-a)
LEAPWORK Automation Platform	Comercial	LEAPWORK Automation Platform tem por objetivo principal testar a automatização de processos repetitivos. Os utilizadores podem criar e executar automação de tarefas numa interface com o design baseado em fluxogramas (Leapwork, n.d.-b), sendo que não é necessário ter conhecimento técnico para definir os fluxos de automação, assim como não é necessário recorrer a código (Leapwork, n.d.-b).

Tabela 11 - Simuladores Robotic Process Automation

Não foi possível instalar 1 dos simuladores (SimCAD) uma vez que não havia uma versão grátis. Também não foi possível a conclusão da instalação de 3 simuladores (USARSim, UberSim, SimRobot) uma vez que apareceram erros que não foi possível solucionar. No caso do simulador Gazebo, chegou-se à conclusão de que seria difícil de manusear, incluindo a sua instalação. Ou seja, restaram assim 6 simuladores de tecnologia *robotic process automation*.

Foi elaborada uma tabela com os 6 simuladores de modo a ser possível observar as funções comuns entre os mesmos:

F1: Manipular o ambiente: definir ambientes realistas (ex: linha de produção)

F2: Manipular tecnologia de automação: definir a tecnologia a ser usada para automatizar tarefas (ex: robôs/algoritmos)

F3: Gerir fluxos de automação: definir e interconectar fluxos de informação e tarefas automatizadas

F4: Navegar por áreas problemáticas: identificar problemas para apoiar decisões de melhoria

F5: Analisar decisões automáticas: visualizar os impactos das decisões de automação nos ambientes realistas definidos

	F1	F2	F3	F4	F5
Argos (OS)	X	X			X
EyeSim (OS)	X	X	X		X
WeBots (OS)	X	X			X
HEEDS	X	X	X	X	X
ANSYS ACT	X	X	X	X	X
LEAPWORK	X	X	X	X	X

Tabela 12 – Simuladores Finais Robotic Process Automation

É possível observar através da tabela que os simuladores comerciais são os mais completos, apresentando todas as funções da tabela. De entre os simuladores *open source*, o mais completo, aquele que mais funcionalidades apresenta é o EyeSim.

6.6. Simuladores Finais

Depois das várias análises realizadas aos vários simuladores, chegou-se à lista final de 4 simuladores, os quais respeitam os critérios que foram definidos anteriormente.

De seguida encontra-se uma tabela com uma descrição do simulador, um guia de instalação e as suas principais características/funções, para cada um dos simuladores finais:

- **CupCarbon**

Simulador	CupCarbon (http://cupcarbon.com/)
Tecnologia Emergente	Internet of Things
Tipo	Open Source
Descrição	O CupCarbon é um simulador de IoT open source que foi originalmente projetado como um simulador com forte ênfase no suporte à mobilidade de nós geográficos tendo por base ambientes do mundo real. Assim, este simulador compreende a mobilidade de nós geográficos em ambientes urbanos inteligentes para que por exemplo seja possível controlar e monitorar variáveis como o impacto ambiental do tráfego automóvel entre outras utilizações como a gestão do tráfego através de semáforos inteligentes, seleção de rotas de pedestres com base em parâmetros ambientais. O simulador utiliza agentes móveis e topologia detalhada do nível da rua com base em mapas do mundo real.
Especificações técnicas	<ul style="list-style-type: none"> • É necessário possuir uma versão Java a executar corretamente no computador e também assegurar que se possui um certo conhecimento acerca dos comandos e regras da linguagem SenScript; • Instalação: <ul style="list-style-type: none"> ○ Fazer o download de uma pasta zipada e fazer a extração da pasta; ○ Para ter acesso ao simulador basta executar o ficheiro capcarbon.jar, que se encontra dentro da pasta; • As redes podem ser projetadas e prototipadas por uma interface ergonómica, usando a estrutura OpenStreetMap (OSM) para implementar sensores diretamente no mapa; • Oferece a possibilidade de simular algoritmos e cenários em várias etapas; • Disponibiliza guias e tutoriais para auxiliar, sendo que o guia inclui 17 exemplos básicos e 4 avançados; • Integra a API do OSM (Open Street Maps) e do Google Maps, permitindo a simulação de locais reais e utilizando de um fundo uniforme. Para além disso, através da API Google Elevation é ainda possível obter uma visualização da elevação do solo.
Funcionalidades	<ol style="list-style-type: none"> 1. Preparar Teste <ol style="list-style-type: none"> a. Criar um novo projeto; b. Abrir um projeto recente; c. Criar cenários de mobilidade: permite o desenho de cenários de mobilidade e a geração de eventos naturais, como incêndios ou fugas de gás, assim como a simulação de veículos e objetos voadores.

	<ul style="list-style-type: none"> d. Criar etapas e algoritmos; e. Criar objetos de simulação; f. Configurar os nós da rede: permite a configuração dos nós dinamicamente, ou seja, permite dividir os nós em redes separadas, ou unir em redes que sejam diferentes. g. Adicionar nós do sensor, estações base (Sinks), gases(atmosféricos), tempo/meteorologia, nós de sensor aleatoriamente; h. Integrar Open Street Maps e do Google Maps e Google Elevation; i. Parametrizar dispositivos: <ul style="list-style-type: none"> i. Network Information: encontram-se dados como número de sensores, número de sensores marcados, número de sensores isolados. ii. Radio Parameters: possui parâmetros para controlo do radio. iii. Devices & Objects: contém a lista dos dispositivos móveis criados e conectados à rede, assim como outros objetos. iv. Device Parameters: contém os parâmetros e os diferentes valores dos vários dispositivos ligados como por exemplo latitude, longitude, raio. <p>2. Executar Teste</p> <ul style="list-style-type: none"> a. Visualizar objetos em 2D ou 3D b. Parar a Simulação; c. Perceber o consumo de energia despendido na simulação; d. Executar códigos específicos através da SenScript window. <p>3. Analisar Resultados do Teste</p> <ul style="list-style-type: none"> a. Análise de dados geográficos: analisar dados relacionado com os mapas, distâncias e tempo. b. Melhorar a simulação: contém diversas funções/algoritmos para melhorar a simulação, ou seja, torná-la mais eficiente e eficaz, produzindo assim melhores resultados. <p>4. Guardar o Teste.</p>
<p>Objetivos de Aprendizagem</p>	<ul style="list-style-type: none"> • Compreender a mobilidade dos nós geográficos em ambientes urbanos inteligentes; • Testar opções de planeamento/gestão urbana.

Tabela 13 - Simulador Final CupCarbon

- Anylogic

Simulador	ANYLOGIC (https://www.anylogic.com/)
Tecnologia Emergente	Digital Twin
Tipo	Open Source
Descrição	O Anylogic é uma aplicação da tecnologia Digital Twin que permite a utilização de três métodos (evento discreto, baseado em agente, dinâmica do Sistema), para simular sistemas complexos. O objetivo principal deste simulador é apoiar a decisão pela exploração de vários cenários de simulação de operações e serviços.
Especificações técnicas	<ul style="list-style-type: none"> • Suportado pelos seguintes sistemas: <ul style="list-style-type: none"> ○ Microsoft Windows 10, x86-32 ex64; ○ Microsoft Windows 8, x86-32 e x64; ○ Microsoft Windows 7 SP1, x86-32 e x64; ○ Apple Mac OS X 10.10 (Yosemite) ou mais recente; ○ Universal; ○ SuSE Linux, x64; ○ Ubuntu Linux 10.04 ou acima, x64; ○ Linux Mint 17 ou acima, x64; • Requisitos: <ul style="list-style-type: none"> ○ 1.5GB de espaço livre no disco; ○ 4GB a 8GB de memória; ○ processador moderno com pelo menos 2 cores para um desempenho otimizado; • Instalação: <ul style="list-style-type: none"> ○ É necessário o preenchimento de um formulário para adquirir o simulador, sendo depois feito o download de ficheiro que tem 715MB; ○ É necessário aceitar os Termos e Condições e depois é dada a opção de instalar ou customizar.
Funcionalidades	<ol style="list-style-type: none"> 1. Preparar a simulação: <ol style="list-style-type: none"> a. Criar objetos: modelação envolvendo eventos e agentes, utilizando uma biblioteca de objetos padrão, onde é possível criar objetos personalizados sem recorrer à programação; b. Criar processos: permite a modelação de processos; c. Manipular parâmetros: vários cenários podem ser explorados variando os parâmetros. Podem ser inspecionados e consultados enquanto estão em ação e comparados entre si; 2. Executar a simulação: <ol style="list-style-type: none"> a. Registrar dados de entrada: há um registo de dados de entrada nos diversos processos e uma posterior análise detalhada; b. Executar modelo/processo: após a elaboração do processo há uma execução; c. Utilizar métodos: possui três métodos (evento discreto, baseado em agente, dinâmica do Sistema), sendo que os três métodos podem ser usados em qualquer combinação, para simular sistemas de

	<p>negócios de qualquer complexidade.</p> <p>d. Integrar mapas: possibilidade de usar <i>GIS Maps</i> para os modelos de simulação. Ao estilo do <i>Google Maps</i>, permite localizar cidades, ruas, estradas, hospitais, lojas ao usar os dados dos mapas. Os elementos do modelo podem também navegar ao longo do mapa, mais concretamente das estradas existentes no mapa (reais).</p> <p>3. Analisar Resultados da simulação:</p> <p>a. Visualizar resultados: depois da execução será apresentado vários resultados, nos quais é possível analisar esses mesmos resultados e alterar os valores nos subprocessos de forma a que este processo seja o mais eficaz possível, ou seja tentativa de encontrar uma solução ótima e ideal;</p> <p>b. Visualizar em Cloud: oferece tecnologias em <i>cloud</i>, sendo possível aceder às simulações em qualquer equipamento e em qualquer lugar;</p> <p>c. Visualizar em várias linguagens: fluxogramas de processo; gráficos de estado; gráficos de ação; diagramas de stock e fluxo;</p> <p>d. Visualizar em 2D/3D: permite converter fluxogramas em filmes iterativos com gráficos 3D e 2D. E permite importar modelos 3D personalizados, imagens, desenhos CAD e oferece a capacidade de integrar mapas GIS;</p> <p>4. Guardar Simulação;</p> <p>5. Outras ferramentas: possui bibliotecas de vários sectores, como por exemplo, biblioteca de tráfego rodoviário para movimentação de carros, camiões e autocarros em estradas, estacionamento e fábricas.</p>
Objetivos de Aprendizagem	<ul style="list-style-type: none"> • Compreender o funcionamento de uma linha de produção; • Analisar os resultados de mudanças nos processos; • Explicar novos cenários e mudanças nos processos existentes;

Tabela 14 - Simulador Final Anylogic

- EyeSim

Simulador	EyeSim (http://robotics.ee.uwa.edu.au/eyesim/)
Tecnologia Emergente	Robotic Process Automation
Tipo	Open Source
Descrição	O simulador EyeBot usa realidade virtual EyeSim VR. É um simulador de múltiplos robôs móveis com funcionalidade VR baseada no mecanismo de jogo Unity 3D que permite fazer experiências com os mesmos programas que são executados nos robôs reais, sendo que os ambientes de teste podem ser criados

	facilmente carregando arquivos de texto. O objetivo principal é testar o software de robôs.
Especificações técnicas	<ul style="list-style-type: none"> • É necessário Cygwin (EyeSim version), Xming; • Requisitos: sistema operativo Windows 8.1, 10; • Pode também ser instalado em Mas OS e Linux; • Instalação: <ul style="list-style-type: none"> ○ Fazer download do ficheiro e depois fazer download de uma pasta zipada com os ficheiros Cygwin e Xming; ○ Descompactar ambas as pastas; ○ Por último, fazer download da pasta zipada eyesimX; • Disponibiliza informações de como instalar em diferentes máquinas (Windows, linux) e quais as especificações necessárias para tal.
Funcionalidades	<ol style="list-style-type: none"> 1. Preparar Simulação <ol style="list-style-type: none"> a. Open Terminal: é aberto uma linha de comandos que permite manusear o programa; b. Load Sim: permite abrir novamente a simulação e continuar o trabalho. c. Manipular o “Mundo” <ol style="list-style-type: none"> i. Load World: permite fazer Load do “Mundo” que estávamos a manusear para a simulação. ii. Save World: é possível guardar o “Mundo” da simulação, podendo com toda a destreza ser completo posteriormente. iii. Create World: permite Criar o “Mundo” para proceder à simulação. iv. Reset World: é possível Criar um “Mundo” novo sem que se tenha que utilizar a própria funcionalidade de “Criar”, ou seja após a exploração da simulação caso queiramos abrir uma nova simulação podemos simplesmente dar Reset o que nos permite reconfigurar o “Mundo”. d. Parametrizar objetos: <ol style="list-style-type: none"> i. Load Object: contém um conjunto de objetos que podem ser utilizados para a simulação, no entanto também nos permite “carregar” para o “Mundo” novos objetos. ii. Add Object: permite ao agente adicionar diferentes objetos na simulação. iii. View Objects: permite observar todos os objetos em causa da simulação. e. Parametrizar robôs: <ol style="list-style-type: none"> i. Load Robot: o simulador contém com um conjunto de “robôs” que podem ser utilizados para a simulação, no entanto também é possível “carregar” para o mundo outros

	<p>tipos de robôs.</p> <ul style="list-style-type: none"> ii. Add Robot: permite ao utilizador adicionar diferentes tipos de robôs na simulação. iii. View Robots: permite observar os robôs que estão a ser utilizados na simulação. <p>2. Executar Simulação</p> <ul style="list-style-type: none"> a. Pausar e Continuar Simulação b. Manipular estado: <ul style="list-style-type: none"> i. Add Market: permite ao utilizador adicionar marcas na simulação. ii. Save State: permite guardar o estado da simulação. iii. Load State: após o estado ter sido guardado, é possível fazer download desse estado para a ferramenta. c. Manipular paredes: <ul style="list-style-type: none"> i. Add Wall: permite adicionar uma parede ao ambiente de simulação. ii. Remove Wall: é possível remover a parede que foi previamente colocada na simulação. iii. Paint Wall: permite fazer alterações estéticas na simulação, como por exemplo pintar a parede. <p>3. Analisar resultados da simulação</p> <ul style="list-style-type: none"> a. Birdseye View: permite observar a simulação numa perspetiva de cima. b. Follow Object: permite observar de perto o objeto. c. Reset Camera: permite reconfigurar a câmara. d. View Log: permite ao utilizador observar algumas informações à cerca da simulação nomeadamente o tempo cronometrado de quando iniciou, bem como por exemplo quando observou um objeto de perto. <p>4. Guardar Simulação: permite guardar o ficheiro da simulação em qualquer momento.</p> <p>5. Outras ferramentas:</p> <ul style="list-style-type: none"> a. Help: permite ao agente procurar alguma ajuda ou alguma informação da simulação/ferramenta. b. RoBIOS API: abre ao utilizador uma página HTML, que contém algumas informações na livraria bem como alguns comandos de utilização. c. About: permite ao utilizador retirar informação sobre o nome da ferramenta e a versão. d. Changelog: abre uma espécie de livraria dentro da ferramenta
<p>Objetivos de Aprendizagem</p>	<ul style="list-style-type: none"> • Compreender o potencial do uso de robôs em espaços com barreiras; • Testar software de robôs;

Tabela 15 - Simulador Final EyeSim

- Elemental Battles

Simulador	Elemental Battles (https://battles.eos.io/)
Tecnologia Emergente	Blockchain
Tipo	Open Source
Descrição	<p>O Elemental Battles é um jogo tutorial de codificação online que ensina ao utilizador o básico para construir a base de código EOSIO, um software que introduz uma arquitetura <i>blockchain</i> projetada para permitir o dimensionamento vertical e horizontal de aplicações descentralizadas e pode ser usada para lançar redes <i>blockchain</i> privadas e públicas. Ensina ainda a construir contratos inteligentes e criar uma aplicação totalmente funcional na plataforma EOSIO.</p> <p>O objetivo principal do simulador é apoiar a aprendizagem da tecnologia <i>blockchain</i> ao permitir ao utilizador aprender a construir código e aplicações de <i>blockchain</i> bem como <i>smart contracts</i>.</p>
Especificações técnicas	<ul style="list-style-type: none"> • Utiliza as linguagens <i>JavaScript</i> para a construção do <i>frontend</i> da aplicação web do <i>blockchain</i> e C++ para a elaboração do <i>smart contract</i>; • É necessário fazer uma autenticação no site do Elemental Battles; • Após a criação de conta, todas as transações assinadas pela chave privada do titular da conta, podem ser autenticadas através das chaves públicas da conta associada, pelo <i>blockchain</i>; • Requisitos: <ul style="list-style-type: none"> ○ 7GB livres de memória RAM; ○ 20GB de espaço livre em disco; ○ pelo menos 2 cores CPU; • Suportado pelos seguintes ambientes: <ul style="list-style-type: none"> ○ Amazon 2017.09 e mais recentes; ○ Centos 7; ○ Ubuntu 16.04 (Ubuntu 16.10 recomendado); ○ Ubuntu 18.04; ○ MacOS Mojave 10.14 e mais recentes; • Requer a instalação das ferramentas necessárias para a execução do tutorial: <i>Node.js</i>, <i>Docker</i>, <i>EOSIO</i>, <i>EOSIO Contract Development Toolkit</i>, <i>eosjs</i>, <i>Reat</i>, <i>Redux</i>, <i>React-Redux</i>
Funcionalidades	<ol style="list-style-type: none"> 1. Preparar um jogo <ol style="list-style-type: none"> a. Selecionar tutorial/lição; b. Configurar arquivos do <i>smart contract</i>; c. Configurar <i>frontend</i>; d. Criar jogadores; e. Criar adversário IA: a Inteligência Artificial, neste caso, terá 4 estratégias diferentes. A cada ronda, a IA escolhe uma dessas estratégias de uma forma aleatória e decide qual a carta que deve ser jogada, o que acaba por tonar o jogo mais imprevisível. Cada estratégia irá calcular todas as combinações

	<p>possíveis do jogo, aplicando um mecanismo de ponderação estratégica, para determinar qual carta jogar.</p> <p>f. Definir regras do jogo; g. Definir ambiente de jogo;</p> <p>2. Executar jogo/transações; a. Implementar smart contract;</p> <p>3. Análise dos resultados do jogo;</p>
Objetivos de Aprendizagem	<ul style="list-style-type: none"> • Entender o básico para criar a base de código EOSIO; • Explicar como criar um contrato inteligente usando EOSIO;

Tabela 16 - Simulador Final Elemental Battles

6.7. Guias de Aprendizagem

Os guias de aprendizagem foram desenvolvidos de acordo com uma estrutura base, sendo depois feito os ajustes necessários para cada uma das tecnologias e adaptado a cada um dos simuladores. De seguida é apresentada a sua estrutura geral. No Anexo Exemplo de Guia de Aprendizagem encontra-se um exemplo de um guia. Uma vez que os guias foram desenvolvidos para o projeto VOIL, o guia de exemplo em anexo encontra-se em inglês.

Os guias encontram-se divididos em 5 secções principais, são elas: a visão geral do guia, o simulador, o cenário de aplicação, discussão e a abordagem de autoavaliação. Cada uma destas secções são subdivididas para oferecer orientação de aprendizagem detalhada em cada um destes tópicos.

Na primeira secção **visão geral do guia**, podemos encontrar o objetivo do guia, uma descrição do mesmo, por quantos capítulos é composto e o que de forma geral é tratado em cada capítulo. Também é nesta secção que encontramos uma descrição da tecnologia emergente a que o guia se refere bem como a forma como esta tecnologia se relaciona com a transformação digital e o estado da simulação relativamente à tecnologia, isto é, em que fase se encontra a mesma.

Na segunda secção, o **simulador** é descrito bem como o processo de instalação e as principais funções que podemos encontrar no simulador em questão.

Quanto à seguinte secção **cenário de aplicação**, pode-se considerar a maior e mais importante secção de todo o guia de aprendizagem pois é nesta secção que é descrito um cenário que pode ser aplicado ao simulador para desenvolver competências de utilização da tecnologia simulada. A construção deste cenário é descrita passo-a-passo para uma fácil compreensão do mesmo. São descritos os objetivos de aprendizagem, o tempo necessário à construção da simulação bem como à discussão da mesma. Esta secção também contempla alternativas ao cenário principal, ou seja, a mudança de parâmetros para observação das consequências das mudanças realizadas.

A quarta secção é a **discussão**. É nesta secção que são discutidos os resultados da simulação realizada. É feita uma reflexão da experiência realizada assim como dos desafios que foram encontrados

na realização do cenário de aplicação, e também a reflexão das mudanças nas capacidades. É também nesta secção que se propõe a realização de uma avaliação do sucesso da aprendizagem bem como a autodefinição de novos objetivos de aprendizagem e correspondentes cenários de simulação.

A última secção é a **abordagem à autoavaliação**, ou seja, nesta secção são apresentadas questões para permitir ao aprendente fazer uma avaliação dos conhecimentos adquiridos.

6.8. Utilização dos Simuladores e os Desafios da Transformação Digital nas PMEs: resumo do capítulo

Anteriormente foram explicados os principais desafios da transformação digital que estão associados às PMEs. Entre eles podemos encontrar escassez de recursos, falta de conhecimento especializado, falta de conhecimento entre os colaboradores e a abordagem escolhida deve estar alinhada com os objetivos de negócio da PME.

Estes desafios podem ser superados com auxílio à simulação, a simuladores. Os simuladores podem ser usados como ferramentas de aprendizagem uma vez que fornecem um ambiente seguro com os recursos tecnológicos para a aprendizagem de uma tecnologia.

Através dos simuladores é possível combater a escassez de recursos, uma vez, que muitos dos simuladores são de graça, o que facilita a aprendizagem das tecnologias antes das PMEs incorrerem em investimentos avultados. Permite também que a organização não seja posta em perigo, uma vez que tudo é feito num ambiente seguro, sem consequências.

Grande parte dos simuladores possui um manual de aprendizagem que ensina a qualquer pessoa a manusear com o simulador, ou seja, não são precisas pessoas especializadas para conseguir trabalhar com os mesmos. Através da exploração do simulador, qualquer pessoa aprende mais sobre a tecnologia em questão, permitindo assim às PMEs, formar os seus funcionários não especializados. Como consequência o processo de transformação digital é facilitado para as PMEs.

Através das tabelas desenvolvidas para cada simulador durante a realização deste trabalho, é possível ter uma noção de cada simulador, quais as suas especificações técnicas, o processo de instalação e as principais funcionalidades. Isto é vantajoso para as PMEs, no sentido que não têm de pegar num simulador pela primeira vez, sem saberem ao certo para que serve e como este os pode ajudar.

7. Discussão

Nesta secção pretende-se dar resposta à questão de investigação colocada inicialmente, assim como aos objetivos de investigação mencionados na secção 1.2: como aplicar a aprendizagem baseada em simuladores de tecnologia emergente nos esforços de transformação digital das PME's?

O primeiro objetivo definido requer uma definição do conceito de transformação digital. No capítulo 2, esse conceito é definido através de uma análise de literatura realizada, que resultou na definição a seguir:

“A transformação digital são alterações nas formas de trabalho, funções e ofertas comerciais potenciadas pela adoção de tecnologias digitais numa organização ou num ambiente operacional da organização”.

Quanto ao segundo objetivo, foi definido que era necessário definir as vantagens e desafios da transformação digital. Estas encontram-se descritas na secção 2.1 e 2.2., respetivamente.

Na seguinte tabela encontra-se um resumo das vantagens e dos desafios encontrados durante a realização da análise da literatura:

Vantagens da Transformação Digital	Desafios da Transformação Digital
Melhor o desempenho da organização	Ajustamento constantemente os produtos e serviços às condições em mudança
Reduzir custos	Otimização continuamente as cadeias de abastecimento por meio de tecnologias
Criar novas oportunidades de mercado	O dilema de escolher entre o aumento da eficiência das operações atuais ou colocar o foco no cliente
Criar conhecimento	Procura de padrões e ignorar as necessidades específicas de cada cliente
Aumentar a eficácia dos processos	Torna-se difícil avaliar o retorno financeiro
	Difícil perceber quais são os dados necessários capturar e analisar
	Desconfiança dos trabalhadores, pois têm medo de serem substituídos e isso provoca alguma

resistência à mudança

Falta de pessoas competentes

Tabela 17 - Resumo das Vantagens e Desafios da Transformação Digital

O terceiro objetivo definido foi a explicação dos desafios da Transformação Digital nas PMEs. De acordo com a revisão da literatura realizada e explicada na secção 2.3., resultou o seguinte resumo desses mesmos desafios:

- escassez de recursos;
- falta de conhecimento especializado;
- falta de conhecimento entre os colaboradores;
- a abordagem escolhida deve estar alinhada com os objetivos de negócio da PME.

O quarto objetivo que foi inicialmente definido requeria a explicação da importância da simulação, a explicação do seu conceito, bem como vantagens e desafios associados à mesma e ainda a simulação nas PMEs. O resultado da análise da literatura encontra-se ao longo do capítulo 3.

Quando ao conceito pode resumir-se em:

“A simulação como um ambiente seguro, onde os conceitos teóricos podem ser postos em prática. Define-se como um ambiente que facilita uma situação de aprendizagem de uma situação real e permite o desenvolvimento de competências”

Na seguinte tabela podemos encontrar um resumo das vantagens e desafios da simulação:

Vantagens da Simulação	Desafios da Simulação
Testar vários cenários, mudando parâmetros ou adicionando recursos	Dificuldade em medir o desempenho
Prever riscos e a sua gestão	O grau de fidelidade influencia a maneira como a simulação é aceite
Aplicar conhecimento na prática	A segurança psicológica no local de trabalho envolve as perceções que os indivíduos têm sobre as consequências de assumir riscos, tendo consequências na simulação
Permitir o desenvolvimento das capacidades interpessoais tais como o pensamento criativo,	Perda do senso de realidade

autoconfiança

Aprender com a experiência

Melhorar níveis de confiança, o pensamento crítico, o julgamento crítico e as capacidades

Tabela 18 - Resumo das Vantagens e Desvantagens da Simulação

O quinto objetivo definido requeria a definição do conceito de tecnologias emergentes. No capítulo 4, esse conceito é definido através de uma análise de literatura realizada, que resultou na definição a seguir:

“Uma tecnologia emergente é considerada radicalmente nova, e mostra taxas de crescimento maiores comparativamente a outras tecnologias. Pode demorar a aparecer e a marcar a sua posição no mercado, no entanto exerce um impacto notável, ou seja, cria grandes mudanças. São caracterizadas pela incerteza dos seus possíveis resultados e utilização.”

Por fim, com base nos conceitos, vantagens e desafios de cada tecnologia, e como forma de responder à questão inicial, foi elaborada uma lista de 4 simuladores, utilizando-os em ambiente digital de aprendizagem.

A seguinte tabela sumariza os 4 simuladores escolhidos:

Simulador	Descrição
CupCarbon	<p>O CupCarbon é um simulador de IoT open source que foi originalmente projetado como um simulador com forte ênfase no suporte à mobilidade de nós geográficos tendo por base ambientes do mundo real.</p> <p>O objetivo principal do simulador é compreender a mobilidade de nós geográficos em ambientes urbanos inteligentes para que por exemplo seja possível controlar e monitorar variáveis como o impacto ambiental que tem as deslocações dos veículos a combustível, ou outros usos mais complicados como a gestão do tráfego através de</p>

	semáforos inteligentes.
Elemental Battles	<p>O Elemental Battles é um jogo tutorial de codificação online que ensina ao utilizador o básico para construir a base de código EOSIO, um software que introduz uma arquitetura blockchain projetada para permitir o dimensionamento vertical e horizontal de aplicações descentralizadas e pode ser usada para lançar redes blockchain privadas e públicas. Ensina ainda a construir contratos inteligentes e criar uma aplicação totalmente funcional na plataforma EOSIO.</p> <p>O objetivo principal do simulador é apoiar a aprendizagem da tecnologia blockchain ao permitir ao utilizador aprender a construir código e aplicações de blockchain bem como smart contracts.</p>
Anylogic	<p>O Anylogic é simulador open source de Digital Twin que permite a utilização de três métodos (evento discreto, baseado em agente, dinâmica do Sistema), combinando estes três métodos consegue-se simular sistemas complexos.</p> <p>O objetivo principal deste simulador é apoiar a decisão pela exploração de vários cenários de simulação de operações e serviços.</p>
EyeSim	<p>O principal objetivo é testar o software do robô.</p> <p>É um simulador de vários robôs móveis com funcionalidade VR baseada no mecanismo de jogo Unity 3D que permite experimentar os mesmos programas executados em robôs reais.</p>

Tabela 19 - Lista Final de Simuladores

Esta dissertação contribui para três principais grupos: os académicos, os gestores de TI e os gestores das PMES.

É possível encontrar nesta dissertação revisões de literatura realizadas sobre os conceitos de transformação digital, simulação e tecnologias emergentes. Contém também resultados de como os simuladores podem ser bastante úteis para aprender.

Através da realização desta dissertação podemos concluir que áreas como a medicina e a enfermagem usam muito mais a simulação do que áreas de TI. A simulação ainda foi pouco explorada na área de TI e nas PMEs. Este conceito pode mudar uma organização e melhorá-la.

Com a ajuda dos simuladores é possível formar recursos humanos em organizações que não tem recursos financeiros para tal e assim continuar a ter funcionários formados e aptos a adaptarem-se às mudanças do mercado.

8. Conclusões

A elaboração deste trabalho teve como principal objetivo perceber em como a simulação e os simuladores podem ajudar as PMEs a passar por um processo de transformação digital.

A transformação digital é um conceito complexo que engloba muitos outros conceitos e requer recursos que muitas PMEs não possuem, como por exemplo, pessoas especializadas e recursos financeiros.

Através da revisão de literatura realizada foi possível identificar as vantagens e os desafios associados à transformação digital e à transformação digital nas PMEs, em particular.

Foi também realizada uma revisão de literatura sobre o estado da simulação como instrumento de aprendizagem. Também esta foi focada nas PMEs, uma vez que as mesmas eram o objetivo deste trabalho.

Com a identificação dos desafios que as PMEs enfrentam quando atravessam um processo de transformação digital, foi possível estudar vários simuladores de modo a selecionar os que melhor contornariam os desafios identificados na revisão de literatura efetuada.

8.1. Limitações

Aquando do início deste trabalho foi difícil encontrar literatura que abordasse os desafios associados à simulação nas PMEs. Foi necessário recorrer a outros domínios como medicina e enfermagem, por isso não se beneficiou de conhecimento da área de SI.

A maior dificuldade encontrada foi perceber que artigos abandonar, uma vez que chegou a um ponto que os artigos eram bastantes e a informação começava a ficar repetida.

Uma das principais dificuldades foi a instalação de simuladores comerciais, uma vez que para alguns não foi disponibilizado uma versão grátis. Isto limitou o número de simuladores a explorar em detalhe e uma vez que, no geral, simuladores comerciais são mais apelativos e sofisticados a nível visual. Alguns dos simuladores exigiam conhecimento de código, o que também dificultou a sua exploração.

Com o trabalho realizado, assume-se que os simuladores selecionados serão os mais adequados para auxiliar as PMEs, no entanto não foi possível fazer uma avaliação dos mesmos junto das PMEs, por isso não foi possível fazer um estudo e avaliar as escolhas finais.

8.2. Trabalhos Futuros

Apesar de não ter sido possível, durante este trabalho, avaliar os simuladores finais no contexto de

uma PME, é expectável que se desenvolva esse estudo de avaliação da eficácia dos simuladores no contexto real.

Esse estudo permitirá observar como as PMEs lidam com a simulação, contribuindo assim para o perceber a simulação nas PMEs, uma área também ela pouco explorada ainda. Assim como também permitirá observar a maneira como as PMEs contornarão os desafios que a transformação digital traz às organizações.

Assim, estas questões de como se comportam as PMEs com os simuladores, num contexto real e como ultrapassam os desafios da transformação digital necessitam ser mais estudadas. É necessário obter respostas mais concretas, de modo a que sejam desenvolvidas estratégias de transformação digital mais focadas nas PMEs.

Bibliografia

Aggogeri, F., Faglia, R., Mazzola, M., & Merlo, A. (2015). Automating the simulation of SME processes through a discrete event parametric model. *International Journal of Engineering Business Management*, 7(Godište 2015), 7-4.

Ali, M., & Cullinane, J. (2014). A study to evaluate the effectiveness of simulation based decision support system in ERP implementation in SMEs. *Procedia Technology*, 16, 542-552.

Anylogic (n.d.). Anylogic. Retrieved March 17, 2020, from <https://www.anylogic.com/>

ANSYS. (n.d.-b). ANSYS ACT Capabilities Extend Your Simulation Functionality. Retrieved May 13, 2020, from <https://www.ansys.com/products/structures/ansysact/act-capabilities>

ANSYS. (n.d.-c). Virtual System Prototyping. Retrieved May 13, 2020, from <https://www.ansys.com/campaigns/internet-of-things/virtual-system-prototyping>

ANSYS (n.d.). Engineering the Internet of Things. Retrieved March 26, 2020, from <https://www.ansys.com/campaigns/internet-of-things>

ANSYS (n.d.). Ansys ACT: Engineering Workflow Simulation Software The fastest, easiest way to tailor your simulation workflows. Retrieved March 28, 2020, from <https://www.ansys.com/products/structures/ansys-act>

Arcos. (n.d.). iCanCloud. Retrieved May 15, 2020, from <https://www.radiologyinfo.org/en/pdf/mammo.pdf>

Arcos. (n.d.). iCanCloud. Retrieved March 18, 2020, from <https://www.arcos.inf.uc3m.es/old/icancloud/Home.html>

ARGoS (n.d.). ARGoS Large-scale robot simulations. Retrieved March 27, 2020, from <https://www.argos-sim.info/>

Askalon (n.d.). Cloud Application Development and Computing Environment. Retrieved March

19, 2020, from <http://dps.uibk.ac.at/projects/groudsim/>

Bevywise (2020). IoT Simulator. Retrieved March 25, 2020, from <https://www.bevywise.com/iot-simulator/>

Cachin, C., & Vukolić, M. (2017). Blockchain consensus protocols in the wild. arXiv preprint arXiv:1707.01873.

Cant, Robyn P., and Simon J. Cooper. "Simulation-based learning in nurse education: systematic review." *Journal of advanced nursing* 66.1 (2010): 3-15.

Cant, Robyn P., and Simon J. Cooper. "Use of simulation-based learning in undergraduate nurse education: An umbrella systematic review." *Nurse Education Today* 49 (2017): 63-71.

Capterra. (n.d.). Simcad Pro. Retrieved May 15, 2020, from <https://www.capterra.com/p/115058/Simcad-Pro/>

Chou, D. C. (2015). Cloud computing: A value creation model. *Computer Standards & Interfaces*, 38, 72-77.

Cloudbus. (n.d.). EMUSIM: Integrated Emulation And Simulation For Evaluation Of Cloud Computing Applications. <https://doi.org/10.1142/S0129626409000304.EMUSIM>

Cloudbus (n.d.). CloudSim: A Framework For Modeling And Simulation Of Cloud Computing Infrastructures And Services. Retrieved March 18, 2020, from <http://www.cloudbus.org/cloudsim/>

CLOUDSIM & CLOUD SETUP (2013). Running and Using Cloud Analyst. Retrieved March 19, 2020, from <http://cloudsim-setup.blogspot.com/2013/01/running-and-using-cloud-analyst.html>

Create a Soft (n.d.). SimCad Simulation Software. Retrieved March 17, 2020, from <https://www.createasoft.com/>

Crosby, M., Pattanayak, P., Verma, S., & Kalyanaraman, V. (2016). Blockchain technology: Beyond bitcoin. *Applied Innovation*, 2(6-10), 71.

CupCarbon (2020). CupCarbon. Retrieved March 25, 2020, from <http://cupcarbon.com/>

Cyberbotics (n.d.). Webots OPEN SOURCE ROBOT SIMULATOR. Retrieved March 28, 2020, from <https://cyberbotics.com/>

Dean, M. (2017). 6 best IoT simulators for PC. Retrieved May 6, 2020, from <https://windowsreport.com/iot-simulators/>

Demo do Blockchain (n.d.). Blockchain distribuído. Retrieved March 10, 2020, from <https://andersbrownworth.com/blockchain/distributed>

Engel, C. E. (1997). Not just a method but a way of learning. The challenge of problem-based learning, 2, 17-27.

EOS. (n.d.-a). Build on EOSIO. Retrieved March 10, 2020, from <https://eos.io/build-oneosio/>

EthHub (n.d.). Test Networks. Retrieved March 10, 2020, from <https://docs.ethhub.io/using-ethereum/test-networks/>

EyeSim VR (n.d.). EyeSim VR - Unity Based EyeBot Simulator. Retrieved March 27, 2020, from <http://robotics.ee.uwa.edu.au/eyesim/>

Fanning, Ruth M., and David M. Gaba. "The role of debriefing in simulation-based learning." *Simulation in healthcare* 2.2 (2007): 115-125.

Fernandez, D., & Aman, A. (2018). Impacts of robotic process automation on global accounting services. *Asian Journal of Accounting and Governance*, 9, 123-132.

Gaurav, Kumar (2018). iFogSim: An Open Source Simulator for Edge Computing, Fog Computing and IoT. Retrieved March 24, 2020, from <https://opensourceforu.com/2018/12/ifogsim-an-open-source-simulator-for-edge-computing-fog-computing-and-iot/>

Gazebo (2014). Gazebo. Retrieved March 27, 2020, from <http://gazebo.org/>

Github (2020). EdgeCloudSim. Retrieved March 18, 2020 from <https://github.com/CagataySonmez/EdgeCloudSim>

Greencloud. (n.d.). Greencloud - The green cloud simulator. Retrieved March 19, 2020, from <https://greencloud.gforge.uni.lu>

Haag, S., & Anderl, R. (2018). Digital twin–Proof of concept. *Manufacturing Letters*, 15, 64-66.

Haji, Faizal A., et al. "Thrive or overload? The effect of task complexity on novices' simulation-based learning." *Medical education* 50.9 (2016): 955-968.

Halaweh, Mohanad. "Emerging technology: What is it." *Journal of technology management & innovation* 8.3 (2013): 108-115.

Han, S. N., Lee, G. M., Crespi, N., Heo, K., Van Luong, N., Brut, M., & Gatellier, P. (2014). DPWSim: A simulation toolkit for IoT applications using devices profile for web services. In 1st IEEE World Forum on Internet of Things (WF-IoT) (pp. 544–547). IEEE. <https://doi.org/10.1109/WF-IoT.2014.6803226>

Heavin, Ciara, and Daniel J. Power. "Challenges for digital transformation–towards a conceptual decision support guide for managers." *Journal of Decision Systems* 27.sup1 (2018): 38-45.

Hess, Thomas, et al. "Options for formulating a digital transformation strategy." *MIS Quarterly Executive* 15.2 (2016).

Hovancsek, M. T. (2007). Using simulation in nursing education. *Simulation in nursing education: From conceptualization to evaluation*, 1(9).

IBM. (n.d.). Hyperledger: blockchain collaboration changing the business world. Retrieved March 9, 2020, from <https://www.ibm.com/blockchain/hyperledger>

IERC. (2018). IERC LAUNCHES ‘ ENERPORT ’ BLOCKCHAIN ENERGY TRADING PROJECT RECENT. Retrieved March 9, 2020, from www.ierc.ie/news/ierc-launches-enerport-

blockchain-energy-trading-project/

Institute for Robotic Process Automation – IRPA. 2014. Introduction to Robotic Process Automation. Retrieved from <http://irpaai.com/what-is-robotic-process-automation/>

Iotify. (n.d.). Intelligent simulation. Retrieved March 25, 2020, from <https://iotify.io/iotnetwork-simulator/>

Irmak, E., & Bozdal, M. (2018, May). Internet of Things (IoT): The most up-to-date challenges, architectures, emerging trends and potential opportunities. *Foundation of Computer Science*.

Kritzinger, W., Karner, M., Traar, G., Henjes, J., & Sihn, W. (2018). Digital Twin in manufacturing: A categorical literature review and classification. *IFAC-PapersOnLine*, 51(11), 1016-1022.

Krown. (n.d.). 5 Types of Blockchain Simulators. Retrieved May 10, 2020, from <https://krown.io/214/5-types-of-blockchain-simulators>

Leapwork. (n.d.-b). LEAPWORK Automation Platform. Retrieved March 28, 2020, from <https://www.capterra.pt/software/162251/leapwork-automation-platform>

Leapwork. (n.d.-c). RPA Made for Business. Retrieved March 28, 2020, from <https://www.leapwork.com/product/business-automation>

Leapwork. (n.d.-d). Test Automation With LEAPWORK. Retrieved March 28, 2020, from <https://www.leapwork.com/product>

Lee, I., & Lee, K. (2015). The Internet of Things (IoT): Applications, investments, and challenges for enterprises. *Business Horizons*, 58(4), 431-440.

Li, Liang, et al. "Digital transformation by SME entrepreneurs: A capability perspective." *Information Systems Journal* 28.6 (2018): 1129-1157.

Madakam, S., Holmukhe, R. M., & Jaiswal, D. K. (2019). The future digital work force: robotic

process automation (RPA). *JISTEM-Journal of Information Systems and Technology Management*, 16.

Mahfouz, Amr, John Shea, and Amr Arisha. "Simulation based optimisation model for the lean assessment in SME: a case study." *Proceedings of the 2011 Winter Simulation Conference (WSC)*. IEEE, 2011.

Manzini, Raffaella, et al. "Quick and dirty Technology Intelligence for SMEs." *R&D management Conference*, July. 2016.

Morris, Marie Catherine, and Patrick Conroy. "Development of a simulation-based sub-module in undergraduate medical education." *Irish Journal of Medical Science (1971-)* (2019): 1-6.

Mostinckx, S., Van Cutsem, T., Timbermont, S., Boix, E. G., Tanter, É., & De Meuter, W. (2009). Mirror-based reflection in ambienttalk. *Software - Practice and Experience*, 39(7), 661–699. <https://doi.org/10.1002/spe>

Muckler, Virginia C. "Exploring suspension of disbelief during simulation-based learning." *Clinical Simulation in Nursing* 13.1 (2017): 3-9.

Multichain (2020). Multichain. Retrieved March 9, 2020, from <https://www.multichain.com/>

Nayyar, A. (2016). The Best Open Source Cloud Computing Simulators. Retrieved June 10, 2020, from <https://opensourceforu.com/2016/11/best-open-source-cloudcomputing-simulators/>

Network Simulation Tools (n.d.). TOSSIM. Retrieved March 24, 2020, from <http://networksimulationtools.com/tossim/>

Nisula, Karoliina, and Samuli Pekkola. "ERP-based simulation as a learning environment for SME business." *The International Journal of Management Education* 10.1 (2012): 39-49.

Oh, Pok-Ja, Kyeong Deok Jeon, and Myung Suk Koh. "The effects of simulation-based learning using standardized patients in nursing students: A meta-analysis." *Nurse education today* 35.5 (2015): e6-e15.

O'Kane, J., Papadoukakis, A., & Hunter, D. (2007). Simulation usage in SMEs. *Journal of Small Business and Enterprise Development*.

OMNet++ (2020). OMNeT++ 6.0 Preview 8 Available. Retrieved March 25, 2020, from <https://omnetpp.org/>

Padilha, J. M., Machado, P. P., Ribeiro, A. L., & Ramos, J. L. (2018). Clinical virtual simulation in nursing education. *Clinical Simulation in Nursing*, 15, 13-18.

Parviainen, Päivi, et al. "Tackling the digitalization challenge: How to benefit from digitalization in practice." *International journal of information systems and project management* 5.1 (2017): 63-77.

Pelletier, Claudia, and L. Martin Cloutier. "Challenges of Digital Transformation in SMEs: Exploration of IT-Related Perceptions in a Service Ecosystem." *Proceedings of the 52nd Hawaii International Conference on System Sciences*. 2019.

Piccinini, Everlin, et al. "Transforming industrial business: the impact of digital transformation on automotive organizations." (2015).

PPLWare. (2019). Cooja: Simulador do Contiki OS para a Internet das Coisas. Retrieved May 18, 2020, from <https://pplware.sapo.pt/tutoriais/vamos-experimentar-o-coojacontiki-os/>

Puthal, D., Sahoo, B. P., Mishra, S., & Swain, S. (2015, January). Cloud computing features, issues, and challenges: a big picture. In *2015 International Conference on Computational Intelligence and Networks* (pp. 116-123). IEEE.

Qi, Q., & Tao, F. (2018). Digital twin and big data towards smart manufacturing and industry 4.0: 360 degree comparison. *Ieee Access*, 6, 3585-3593.

Robot Soccer (n.d.). Carnegie Mellon UberSim Project. Retrieved March 27, 2020, from <http://www.cs.cmu.edu/~robosoccer/ubersim/>

Rofer, Thomas (2015). SimRobot - Robotics Simulator. Retrieved March 28, 2020, from

http://www.informatik.uni-bremen.de/simrobot/index_e.htm

Rotolo, Daniele, Diana Hicks, and Ben R. Martin. "What is an emerging technology?." *Research Policy* 44.10 (2015): 1827-1843.

Scalable Network Technologies (n.d.). QualNet - Network Simulation Software. Retrieved March 26, 2020, from <https://www.scalable-networks.com/qualnet-network-simulation>

SimBlock (2019). SimBlock Blockchain network simulator. Retrieved March 10, 2020 from <https://dsg-titech.github.io/simblock/>

SIEMENS (n.d.). HEEDS MDO. Retrieved March 28, 2020, from <https://www.redcedartech.com/index.php/solutions/heeds-software>

SIMIO (2020). Digital Twin. Retrieved March 17, 2020, from <https://www.simio.com/applications/industry-40/Digital-Twin.php>

SimpleSoft (n.d.). SimpleIoTSimulator The InternetOfThings Simulator. Retrieved March 26, 2020, from <https://www.smplsft.com/SimpleIoTSimulator.html>

S. Madakam, R. Ramaswamy, and S. Tripathi, "Internet of Things (IoT): A Literature Review," *J. Comput. Commun.*, vol. 3, no. 3, pp. 164–173, 2015.

Shi, W., Cao, J., Zhang, Q., Li, Y., & Xu, L. (2016). Edge computing: Vision and challenges. *IEEE internet of things journal*, 3(5), 637-646.

Shi, W., & Dustdar, S. (2016). The promise of edge computing. *Computer*, 49(5), 78-81.

SourceForge (2013). DPWSim. Retrieved March 24, 2020, from <https://sourceforge.net/projects/dpwsim/>

SourceForge (n.d.). The Contiki Operating System. Retrieved March 24, 2020, from <https://sourceforge.net/projects/contiki/files/Instant%20Contiki/>

SourceForge (2017). IoTSIM. Retrieved March 26, 2020, from <https://sourceforge.net/projects/iotsim/>

SourceForge (2015). USARSim. Retrieved March 27, 2020, from <https://sourceforge.net/projects/usarsim/>

Srinivasan, Raji. "Sources, characteristics and effects of emerging technologies: Research opportunities in innovation." *Industrial Marketing Management* 37.6 (2008): 633-640.

Sutherland, C. (2013). Framing a Constitution for Robotistan. Hfs Research, ottobre.

Tan, L., & Wang, N. (2010, August). Future internet: The internet of things. In 2010 3rd international conference on advanced computer theory and engineering (ICACTE) (Vol. 5, pp. V5-376). IEEE.

Tao, F., Zhang, H., Liu, A., & Nee, A. Y. (2018). Digital twin in industry: State-of-the-art. *IEEE Transactions on Industrial Informatics*, 15(4), 2405-2415.

Tarutè, Asta, et al. "Identifying Factors Affecting Digital Transformation of SMEs." (2018).

Tetcos (n.d.). NetSim. Retrieved March 26, 2020, from <https://www.tetcos.com/>

Testnet (2020). Testnet. Retrieved March 10, 2020 from <https://en.bitcoin.it/wiki/Testnet>

The Financial Express. 2016. Robotic process automation: The next big disruption. Retrieved from <http://www.financialexpress.com/opinion/robotic-process-automation-the-next-big-disruption/344032/>

Underwood, S. (2016). Blockchain beyond bitcoin.

Van der Aalst, W. M., Bichler, M., & Heinzl, A. (2018). Robotic process automation.

WSNet (n.d.). WSNet/Worldsens simulator. Retrieved March 24, 2020, from <http://wsnet.gforge.inria.fr/index.html>

Xia, F., Yang, L. T., Wang, L., & Vinel, A. (2012). Internet of things. *International journal of communication systems*, 25(9), 1101.

Yli-Huumo, J., Ko, D., Choi, S., Park, S., & Smolander, K. (2016). Where is current research on blockchain technology?—a systematic review. *PloS one*, 11(10), e0163477.

Zeng, X., Garg, S. K., Strazdins, P., Jayaraman, P. P., Georgakopoulos, D., & Ranjan, R. (2017). IOTSim: A simulator for analysing IoT applications. *Journal of Systems Architecture*, 72, 93–107. <https://doi.org/10.1016/j.sysarc.2016.06.008>

Zheng, Z., Xie, S., Dai, H., Chen, X., & Wang, H. (2017, June). An overview of blockchain technology: Architecture, consensus, and future trends. In *2017 IEEE international congress on big data (BigData congress)* (pp. 557-564). IEEE.

Zheng, Z., Xie, S., Dai, H. N., Chen, X., & Wang, H. (2018). Blockchain challenges and opportunities: A survey. *International Journal of Web and Grid Services*, 14(4), 352-375.

Adoption of Information Technology Among Small and Medium Enterprises in Indian Context

Anexo

Simuladores:

- ENERPORT

Simulador	ENERPORT (http://www.ierc.ie/news/ierc-launches-enerport-blockchain-energy-trading-project/)
Tecnologia Emergente	Blockchain
Tipo	Open Source
Descrição	<p>É um projeto lançado na Conferência Anual do Centro Internacional de Investigação Energética (IERC) e tem como objetivo acelerar o comércio de energia <i>peer-to-peer</i> na Irlanda, usando a tecnologia <i>Blockchain</i>.</p> <p>É um projeto colaborativo irlandês financiado pelo governo e pela indústria e liderado pelo IERC e em parceria com o Centro INSIGHT da SFI na NUI Galway, o <i>EnerPort</i>, também, envolve várias empresas nativas, incluindo <i>Systemlink Technologies</i>, <i>MSemicon</i> e <i>Verbatm</i>.</p> <p>Este projeto, tem como objetivo investigar a tecnologia <i>Blockchain</i> como uma solução para a comercialização de energia e também permitirá a possibilidade de recompensar os utilizadores pela economia de energia, além de gerir transações de pequena escala entre os vizinhos que geram eletricidade. Pretende que os consumidores irlandeses possam oferecer ou vender a energia solar que produzam aos seus vizinhos, além de partilhar e gerir a sua própria eletricidade.</p> <p>Um dos resultados deste projeto é a Integração de um simulador <i>Blockchain</i> com <i>OpenDSS (Open Source Distribution System Simulator)</i> para uma rede de distribuição de energia. O objetivo principal do simulador é fornecer um ambiente digital que permite que empresas e consumidores negociem a compra e venda de energia em escala regional e até nacional.</p>
Especificações técnicas	<ul style="list-style-type: none">• Suportado pelos seguintes sistemas operativos:<ul style="list-style-type: none">○ Windows;○ MacOS;○ Ubuntu Linux;• É necessário também o seguinte simulador: <i>OpenDSS (Open Source Distribution System Simulator)</i>;

- Para instalar este simulador necessário é necessário o *Cloning* do projeto do GitHub
A falta de informação sobre o simulador impossibilitou a continuação da instalação do mesmo.

Tabela 20 - Simulador ENERPORT

- **IBM Hyperledger**

Simulador	IBM Hyperledger (https://www.ibm.com/blockchain/hyperledger)
Tecnologia Emergente	Blockchain
Tipo	Open Source
Descrição	<p>O IBM Hyperledger é um simulador <i>Open Source</i> de <i>blockchain</i> que permite o desempenho em escala e preserva a privacidade dos dados das empresas. Permite a adaptação da <i>blockchain</i> às necessidades do sector onde se insere e também permite contratos inteligentes em diferentes linguagens. É um simulador constituído por código aberto, sendo que apresenta recursos inovadores mostrando assim uma transparência e responsabilidade.</p> <p>É um <i>testnet</i> que elimina a necessidade minerar, pois fornece uma interface onde o consenso é derivado pela maioria de 51% dos servidores, de execução independente. Este programa oferece um desempenho em grande escala, preservando a privacidade.</p>
Especificações técnicas	<ul style="list-style-type: none"> • Existem problemas na versão trial, e a versão paga apresenta custos um bocado elevados para pequenas empresas. • Para o <i>Peer</i> é necessário: <ul style="list-style-type: none"> ○ 1.1 CPU; ○ 2.2 GB de memória; ○ 200GB de armazenamento (inclui 100 GB para peer e 100 GB para base de dados de estado); • Para o CA é necessário: <ul style="list-style-type: none"> ○ 0,1 CPU; ○ 0,2GB de memória; ○ 20GB de armazenamento; • Para o Nó de pedido é necessário: <ul style="list-style-type: none"> ○ 0,35 CPU; ○ 0.7GB de memória;

	<ul style="list-style-type: none"> ○ 100GB de armazenamento; • Instalação: <ul style="list-style-type: none"> ○ É necessário a criação de uma conta e depois introduzir os seus dados, bem como dados do seu cartão de crédito e submeter (de referir, que na conta trial não deverá ter qualquer tipo de gasto, porém 1€ será creditado da sua conta para a ativação, este será devolvido num prazo de 7 dias.). • A ativação da conta, fornece 200 créditos para gastar na IBM Cloud, durante 30 dias. <p>Falta de suporte por parte do fornecedor impossibilitou a continuação da exploração deste simulador.</p>
--	--

Tabela 21 - Simulador IBM Hyperledger

- **Multichain**

Simulador	Multichain (https://www.multichain.com/)
Tecnologia Emergente	Blockchain
Tipo	Open Source
Descrição	O Multichain é um simulador Open Source; o seu objetivo principal é apoiar as organizações a construir e iniciar uma aplicação de blockchain com custos reduzidos. O utilizador consegue implementar dois computadores: um como teste e outro como uma versão real (Krown, n.d). Destina-se principalmente para utilizadores que pretendam desenvolver plataformas de blockchain no sector financeiro (Cachin & Vukolic, 2017).
Especificações técnicas	<ul style="list-style-type: none"> • Requisitos: <ul style="list-style-type: none"> ○ 512 MB de memória RAM; ○ 1 GB de espaço livre em disco; • Ambientes suportados: <ul style="list-style-type: none"> ○ Linux: 64-bit; ○ Ubuntu 12.04+; ○ CentOS 6.2+; ○ Debian 7+; ○ Fedora 15+;

	<ul style="list-style-type: none"> ○ RHEL 6.2+; ○ Windows: 64-bit, suporta Windows 7,8,10, Server 2008; ○ Mac: 64-bit, suporta OS X 10.12; ● Instalação: <ul style="list-style-type: none"> ○ É necessário fazer o download do Multichain e depois deverá ser feita a sua extração uma vez que se encontra num ficheiro zip e, de seguida, copiar a localização do ficheiro; ○ De seguida, é necessário abrir a linha de comandos e inserir “cd” e colocar a localização do ficheiro; ○ Depois destes passos, o utilizador encontra-se preparado para criar Blockchain e, para tal necessita de executar o seguinte comando: “<i>multichain-util create my-first-chain</i>”, sendo que “<i>my-first-chain</i>” é o nome dado. Após a criação da Blockchain, é gerado um ficheiro que contém todos os parâmetros que serão utilizados.
Funcionalidades	<ol style="list-style-type: none"> 1. Preparar simulação: <ol style="list-style-type: none"> a. Fazer transações: permite transações e sistemas de pagamento, ou seja, permite aos utilizadores realizarem transferências sem a intervenção de terceiros; b. Definir permissões: permite definir uma blockchain tão aberta ou tão fechada quanto necessário; c. Assinar transações: permite assinar as suas transações digitalmente através de uma chave privada, e ainda efetuar transações brutas. d. Enviar transações; e. Receber transações; f. Criar fluxos; g. Implementar blockchains: permite a implementação de ilimitadas blockchains por servidor para aplicações de <i>cross-chain</i> (conseguem comunicar uma com a outra) <ol style="list-style-type: none"> i. Criar ativos; ii. Emitir ativos: permite a emissão de milhões de ativos numa blockchain, sendo que são todos rastreados e verificados no nível da rede. iii. Criar blocos; 2. Executar Simulação 3. Analisar resultados da simulação <ol style="list-style-type: none"> a. Filtrar informação; 4. Guardar Simulação

	<p>a. Criar base de dados: permite a criação de várias bases de dados com valores-chave, séries temporais ou de identidade numa blockchain, o que torna ideal para partilha de dados, registo de data/hora e o arquivamento criptografado.</p> <p>5. Integrar soluções: permite a integração com soluções tecnológicas através da API, de forma a poder adaptá-lo à solução.</p> <p>6. Controlar: disponibiliza a opção de controlo, ou seja, pode decidir-se quem pode conectar-se;</p> <p>7. Outras ferramentas: no site também é possível encontrar um tutorial de como desenvolver uma blockchain no simulador, com os passos necessários.</p>
Objetivos de Aprendizagem	<ul style="list-style-type: none"> • Compreender os princípios básico para a construção de uma blockchain privada; • Formular uma blockchain privada;

Tabela 22 - Simulador Multichain

- **Bitcoin Testnet**

Simulador	Bitcoin Testnet (https://en.bitcoin.it/wiki/Testnet)
Tecnologia Emergente	Blockchain
Tipo	Open Source
Descrição	O objetivo principal do Bitcoin Testnet é testar a sua aplicação com recurso à tecnologia <i>blockchain</i> para se certificar de que funciona e se comporta como espetável, antes da implementação. As moedas de teste são separadas e distintas dos bitcoins reais e não têm qualquer valor, permitindo que sejam feitas experiências sem usar bitcoins reais.
Especificações técnicas	<ul style="list-style-type: none"> • Sistemas suportados: <ul style="list-style-type: none"> ○ Windows; ○ Mac OS X; ○ Linux; • É necessário ativar o recurso “Windows Subsystem for Linux”. Fazer o download e instalação do ubuntu (Distribuição Linux). • Instalação:

	<ul style="list-style-type: none"> ○ Fazer download de um ficheiro ZIP; ○ Criar conta no Linux; ○ Instalar os pacotes e atualizá-los; ○ Obter a fonte do GitHub; ○ Instalar a ferramenta de <i>cross-compilation</i> “mingw-w64” e alterar o “mingw32 g++” para <i>posix</i>; ○ Instalar o pacote “Build-essencial”, necessário para o obter o Bitcoin Core testnet no Windows; ○ Gerar o Bitcoin Core para Windows; ○ Abrir o Bitcoin Core (testnet);
Funcionalidades	<ol style="list-style-type: none"> 1. Preparar Teste: <ol style="list-style-type: none"> a. Criar a carteira; b. Encriptar a carteira: permite encriptar a carteira, aumentando assim a sua segurança; 2. Executar Teste <ol style="list-style-type: none"> a. Assinar/verificar mensagens; b. Registrar transações: permite fazer o envio de pagamentos para um determinado endereço. É necessário preencher um formulário que contem os seguintes campos: “Pagar a:” (campo onde temos de colocar o endereço de destino); “Etiqueta” (permite adicionar uma etiqueta ao endereço em questão, para que esse mesmo endereço seja adicionado a uma lista de endereços); “Quantia:” (digitar a quantidade de moedas que queremos enviar); “Taxa da transação:” (permite escolher a taxa de transação, bem como o número de blocos que queremos para o nosso tempo de confirmação); c. Solicitar pagamento: permite solicitar pagamentos, após preenchimento de um formulário; 3. Analisar Resultados do Teste <ol style="list-style-type: none"> a. Visualizar transações: permite verificar os endereços de envio e de receção de pagamentos, sendo também possível verificar informações acerca da cadeia de blocos, isto é, a data do último bloco, bem como cada um dos nós da cadeia; b. Visualizar histórico de transações: permite visualizar o histórico de transações; c. Consultar saldo da carteira de bitcoins: permite consultar o saldo da carteira, “saldo pendente” e o “total”, sendo que a unidade está em bitcoin (BTC). É possível também verificar as transações recentes; 4. Guardar Teste

Tabela 23 - Simulador Bitcoin Testnet

• Ethereum Testnet

Simulador	Ethereum Testnet (https://docs.ethhub.io/using-ethereum/test-networks/)
Tecnologia Emergente	Blockchain
Tipo	Open Source
Descrição	A Ethereum Test Net é um simulador <i>open source</i> que fornece uma rede teste. O objetivo principal deste simulador é oferecer um ambiente seguro para testar uma aplicação de blockchain de modo a garantir que quando implementada, a aplicação funcione conforme o esperado.
Especificações técnicas	<ul style="list-style-type: none"> • Encontra-se disponível nos sistemas operativos: <ul style="list-style-type: none"> ○ Windows; ○ MacOS, Ubuntu Linux; ○ qualquer outra plataforma Unix que suporte Java; • Instalação da linguagem de programação Ruby; • Instalação: <ul style="list-style-type: none"> ○ Instalar o <i>Homebrew</i> para, de seguida, instalar o Geth ○ Criar o arquivo <i>Genesis</i> e definir as suas características. Este vai ser o primeiro bloco da cadeia ○ Iniciar o nosso “<i>node</i>” ○ Configurar um caminho para outros “<i>peers</i>” se juntarem à rede ○ Criar uma conta pessoal ○ “<i>Mine Ether</i>”, ou seja, qualquer operação no Ethereum só é possível se existir Ether. ○ Para aceder ao simulador, é necessário executar o seguinte comando no Ubuntu: “geth --datadir ./myDataDir --networkid 1114 console 2>> myEth.log”.
Funcionalidades	<ol style="list-style-type: none"> 1. Preparar simulação: <ol style="list-style-type: none"> a. Criar redes de blocos: os blocos são desenvolvidos com nós de <i>Etherscan</i> e RPC, usados para contruir e explorar blocos; b. Verificar identidade: possui uma prova de autoridade, o que impede ataques à blockchain; c. Criar smart contract d. Assinar contrato: é obrigatório assinar o contrato inteligente; 2. Executar simulação:

- a. Mineração de redes de blocos;
- 3. Análise dos resultados da simulação;
- 4. Guardar simulação;

Tabela 24 - Ethereum Testnet

• SIMBlock

Simulador	SIMBlock (https://dsg-titech.github.io/simblock/)
Tecnologia Emergente	Blockchain
Tipo	Open Source
Descrição	<p>É um simulador de rede <i>blockchain</i> de código aberto, desenvolvido pelo <i>Distributed Systems Group</i>, Instituto de Tecnologia de Tóquio. É orientado a eventos e é adequado para uso em pesquisa de rede <i>blockchain</i>. Também possui uma ferramenta de visualização, pela qual se pode ver a transição da propagação de blocos.</p> <p>O principal objetivo deste simulador é manipular uma rede <i>blockchain</i> constituída por um elevado número de nós de maneira a conseguir para determinar o impacto das alterações introduzidas no desempenho e na segurança na <i>blockchain</i>.</p>
Especificações técnicas	<ul style="list-style-type: none"> • Suportado pelos sistemas operativos: <ul style="list-style-type: none"> ○ Windows; ○ MacOS; ○ Ubuntu Linux ou qualquer outra plataforma Unix que suporte Java; • São necessários o JDK (Versão 1.8.0 ou superior), o <i>Grable</i> (Versão 5.1.1 ou superior e está incluído no ficheiro zip que contém o SimBlock) e o Ubuntu (pode ser obtido através da Microsoft Store); • Instalação: <ul style="list-style-type: none"> ○ Após a realização do download do ficheiro zip, é necessário extrair-lo numa diretoria vazia; ○ Criar conta no Linux; ○ Instalar os pacotes e atualizá-los; ○ <i>Cloning</i> do repositório <i>GitHub</i>; ○ Instalar a <i>build</i> do <i>Gradle</i>; ○ Executar o comando <i>gradle</i>;

	<ul style="list-style-type: none"> ○ De forma a aceder ao Simulador SimBlock, é apenas necessário executar o comando “gradle :simulador:run” no Ubuntu.
Funcionalidades	<ol style="list-style-type: none"> 1. Preparar Simulação <ol style="list-style-type: none"> a. Criar rede de blocos (blockchain) b. Abrir rede de blocos: permite utilizar um ficheiro externo de maneira a ver a evolução dos blocos nessa execução; c. Implementar algoritmo de seleção de nós: permite implementar facilmente o algoritmo de seleção de nós vizinhos; d. Manipular condições de rede: atribui exclusivamente a dificuldade a todos os nós; 2. Executar simulação 3. Analisar os resultados da simulação <ol style="list-style-type: none"> a. Navegar a rede de blocos: permite navegar temporalmente na evolução da transição dos blocos, podendo ser escolhido um determinado momento ou intervalo de tempo; 4. Guardar Simulação

Tabela 25- Simulador SIMBlock

- **The Principle of Blockchain**

Simulador	The Principles of Blockchain (https://andersbrownworth.com/)
Tecnologia Emergente	Blockchain
Tipo	Open Source
Descrição	<p>Este simulador surgiu num contexto académico pelas mãos de Anders Brwnworth, uma vez que o mesmo, achou que havia uma debilidade nas ferramentas para o ensino de <i>blockchain</i> desenvolvendo esta solução, que disponibilizou de forma “open source”.</p> <p>O objetivo principal do simulador é facilitar a aprendizagem da tecnologia <i>blockchain</i> ao permitir que pessoas com pouca experiência possam aprender de forma simples.</p>
Especificações técnicas	<ul style="list-style-type: none"> • É aconselhado que os utilizadores tenham algum conhecimento de node.js;

	<ul style="list-style-type: none"> • Para poder executar o simulador a máquina precisará de ter instalado e configurado o node.js. (https://nodejs.org/en/download/), estando esta tecnologia disponível para as plataformas de Windows, Linux e macOS. • A instalação desta tecnologia divide-se em duas partes, que se encontram interligadas. <ul style="list-style-type: none"> ○ Instalação e Configuração da BlockChain Demo Master: é feito o download do código, depois deverá ser feita a sua extração uma vez que este código vem num ficheiro zip. De seguida, deverá copiar a localização do ficheiro, abrir o terminal node.js e inserir a localização do ficheiro. Inserir na prompt do Node.js o seguinte comando “npm install” e, quando esta estiver terminada deverá introduzir “node./bin/www”.
Funcionalidades	<ol style="list-style-type: none"> 1. Criar um hash (https://blockgeeks.com/guides/what-is-hashing/) 2. Criar um Bloco 3. Criar uma cadeia de blocos. 4. Definir aplicações em blockchain (explorar que tipo de aplicações são as permitidas)

Tabela 26 - Simulador The Principles of Blockchain

• Elemental Battles

Simulador	Elemental Battles (https://battles.eos.io/)
Tecnologia Emergente	Blockchain
Tipo	Open Source
Descrição	<p>O Elemental Battles é um jogo tutorial de codificação online que ensina ao utilizador o básico para construir a base de código EOSIO, um software que introduz uma arquitetura <i>blockchain</i> projetada para permitir o dimensionamento vertical e horizontal de aplicações descentralizadas e pode ser usada para lançar redes <i>blockchain</i> privadas e públicas. Ensina ainda a construir contratos inteligentes e criar uma aplicação totalmente funcional na plataforma EOSIO.</p> <p>O objetivo principal do simulador é apoiar a aprendizagem da tecnologia <i>blockchain</i> ao permitir ao utilizador aprender a construir código e aplicações de <i>blockchain</i> bem como <i>smart contracts</i>.</p>
Especificações técnicas	<ul style="list-style-type: none"> • Utiliza as linguagens <i>JavaScript</i> para a construção do <i>frontend</i> da aplicação web do <i>blockchain</i> e <i>C++</i> para a

	<p>elaboração do <i>smart contract</i>;</p> <ul style="list-style-type: none"> • É necessário fazer uma autenticação no site do Elemental Battles; • Após a criação de conta, todas as transações assinadas pela chave privada do titular da conta, podem ser autenticadas através das chaves públicas da conta associada, pelo blockchain; • Requisitos: <ul style="list-style-type: none"> ○ 7GB livres de memória RAM; ○ 20GB de espaço livre em disco; ○ pelo menos 2 cores CPU; • Suportado pelos seguintes ambientes: <ul style="list-style-type: none"> ○ Amazon 2017.09 e mais recentes; ○ Centos 7; ○ Ubuntu 16.04 (Ubuntu 16.10 recomendado); ○ Ubuntu 18.04; ○ MacOS Mojave 10.14 e mais recentes; • Requer a instalação das ferramentas necessárias para a execução do tutorial: <i>Node.js, Docker, EOSIO, EOSIO Contract Development Toolkit, eosjs, Reat, Redux, React-Redux</i>
Funcionalidades	<ol style="list-style-type: none"> 4. Preparar um jogo <ol style="list-style-type: none"> a. Selecionar tutorial/lição; b. Configurar arquivos do smart contract; c. Configurar frontend; d. Criar jogadores; e. Criar adversário IA: a Inteligência Artificial, neste caso, terá 4 estratégias diferentes. A cada ronda, a IA escolhe uma dessas estratégias de uma forma aleatória e decide qual a carta que deve ser jogada, o que acaba por tornar o jogo mais imprevisível. Cada estratégia irá calcular todas as combinações possíveis do jogo, aplicando um mecanismo de ponderação estratégica, para determinar qual carta jogar. f. Definir regras do jogo; g. Definir ambiente de jogo; 5. Executar jogo/transações; <ol style="list-style-type: none"> a. Implementar smart contract;

	6. Análise dos resultados do jogo;
Objetivos de Aprendizagem	<ul style="list-style-type: none"> • Entender o básico para criar a base de código EOSIO; • Explicar como criar um contrato inteligente usando EOSIO;

Tabela 27 - Simulador Elemental Battles

• EdgeCloudSim

Simulador	EdgeCloudSim (https://github.com/CagataySonmez/EdgeCloudSim)
Tecnologia Emergente	Cloud e Edge Computing
Tipo	Open Source
Descrição	Este simulador é open source de EdgeCloud Computing. O objetivo principal é fornecer um ambiente de simulação onde é possível realizar experiências que tenham em conta os recursos computacionais e de rede. É uma extensão do simulador CloudSim, mas adiciona umas funcionalidades.
Especificações técnicas	<ul style="list-style-type: none"> • Requer uma plataforma IDE (Ambiente de desenvolvimento integrado) como Eclipse ou NetBeans ou um terminal do computador através da linha de comandos; • Instalação: <ul style="list-style-type: none"> ○ Fazer download e desempacotar o arquivo ZIP ○ É necessário abrir o NetBeans e criar um novo projeto ○ Defina um nome para o projeto e adicione a pasta de origem existente; ○ Abra a página de configuração do projeto e as bibliotecas necessárias através do botão Adicionar JAR/Pasta; ○ Defina a opção Caminho de Trabalho para que os caminhos relativos funcionem corretamente; ○ Execute <code>sample_application / mainApp.java</code> • Para o devido funcionamento do sistema algumas precauções devem ser tomadas: <ul style="list-style-type: none"> ○ Recursos de computação em nuvem (executando tarefas no dispositivo móvel); ○ Incorporar o modelo de rede de acesso do telemóvel no EdgeCloudSim (3G / 4G / 5G); ○ Migração de tarefas entre a VMs Cloud ou Edge; ○ Modelo de consumo de energia para dispositivos móveis e de borda, bem como para os datacenters em nuvem;

	<ul style="list-style-type: none"> ○ Adição do modelo probabilístico de falha de rede considerando o congestionamento ou outros parâmetros, como a distância entre o dispositivo móvel e o ponto de acesso WiFi; ○ Ferramenta visual para exibir a topologia de rede.
Funcionalidades	<ol style="list-style-type: none"> 1. Preparar simulação <ol style="list-style-type: none"> a. Incorporar modelos de comportamento de rede; b. Modelar redes específicas para WLAN e WAN; c. Modelar a mobilidade de dispositivos; d. Decidir como e onde lidar com as solicitações de entrada do cliente; e. Gerir a localização de dispositivos de borda e clientes; 2. Executar Simulação <ol style="list-style-type: none"> a. Testar em três cenários diferentes: <ol style="list-style-type: none"> i. Camada única (Single-tier) - suporta os dispositivos móveis para utilizarem o servidor edge localizado na mesma célula da rede. ii. Duas camadas (Two-tier) - nesta arquitetura, os dispositivos móveis podem enviar as suas tarefas para a nuvem global usando a conexão WAN fornecida pelo ponto de acesso conectado (AP). iii. Duas camadas com EO (Two-tier with EO) - Apenas a arquitetura de duas camadas com EO pode descarregar as tarefas para qualquer servidor edge localizado em locais diferentes. Ela assume que o servidor edge e o orquestrador edge estão conectados na mesma rede. b. Gerar tarefas para a configuração fornecida; c. Gerador de carga realista e ajustável; 3. Analisar resultados da simulação <ol style="list-style-type: none"> a. Visualizar a percentagem de tarefas falhadas em função do número de dispositivos móveis; b. Visualizar o tempo médio de processamento em função do número de dispositivos móveis; c. Visualizar a utilização média do servidor em função do número de dispositivos móveis; d. Visualizar a média de tarefas com falha devido à mobilidade em função do número de dispositivos móveis; e. Visualizar a média de tarefas com falha devido à falha na rede em função do número de dispositivos móveis; f. Visualizar a média de tarefas com falha devido à capacidade da VM em função do número de

	<p>dispositivos móveis.</p> <p>g. Visualizar o atraso médio de LAN em função do número de dispositivos móveis;</p> <p>h. Visualizar atraso médio de WAN em função do número de dispositivos móveis;</p> <p>4. Guardar simulação</p>
Objetivos de Aprendizagem	<ul style="list-style-type: none"> • Compreender o funcionamento dos recursos computacionais e de rede; • Analisar o desempenho de um sistema de computação de borda; • Explicar como os recursos computacionais afetam o sistema;

Tabela 28 - Simulador EdgeCloudSim

• CloudSim

Simulador	CloudSim (http://www.cloudbus.org/cloudsim/)
Tecnologia Emergente	Cloud e Edge Computing
Tipo	Open Source
Descrição	<p>É uma ferramenta de modelagem e simulação de EdgeCloud desenvolvida na Universidade de Melbourne, na Austrália. Suporta a modelagem e simulação de ambientes de EdgeCloud Computing em larga escala, incluindo <i>datacenters</i>, nós de computação, máquinas virtuais, provisionamento de recursos e provisionamento de máquinas virtuais e gestão de energia.</p> <p>O objetivo principal é a realização de testes robustos com base em configurações personalizadas e implementação aplicações para Cloud.</p>
Especificações técnicas	<ul style="list-style-type: none"> • Requisitos: <ul style="list-style-type: none"> ○ Java version 8 ou mais recente; ○ Conhecimento de como usar CMD e java files; • Instalação: <ul style="list-style-type: none"> ○ Fazer download e extrair o conteúdo para uma pasta ○ Abra o Eclipse.exe e agora no Eclipse, abra o novo assistente de projeto; ○ Em seguida, selecione a opção “Java Project” uma vez feito, clique em “Next “; ○ Uma nova janela do projeto será aberta, onde será preciso pôr o nome do projeto e o caminho do

	<p>código-fonte do projeto CloudSim (o caminho onde o projeto CloudSim foi descompactado);</p> <ul style="list-style-type: none"> ○ Clique em Avançar para definir as configurações do projeto. ○ Certifique-se de navegar pelo caminho até poder ver a pasta bin, documentos, exemplos etc. no plano de navegação. ○ Depois de concluído, clique em "Avançar" para ir para a próxima etapa, ou seja, definir as configurações do projeto. ○ Abra a guia 'Bibliotecas' e, se não encontrar commons-math3-3.x.jar (aqui 'x' significa a versão menor da biblioteca que pode ser 2 ou superior) na lista, clique em 'Adicionar externo Jar '(commons-math3-3.x.jar será incluído no projeto nesta etapa). ○ Depois de clicar em "Adicionar JAR externo", abra o caminho em que os binários do commons-math foram descompactados e selecione "Commons-math3-3.x.jar" e clique em abrir. ○ Após a configuração do projeto, pode abrir o Explorador de Projetos e começar a explorar o projeto Cloudsim. <ul style="list-style-type: none"> • Tem à disposição vária documentação para auxílio, como exemplos e como instalar; • É escrito em java;
Funcionalidades	<ol style="list-style-type: none"> 1. Preparar a simulação: <ol style="list-style-type: none"> a. Criar servidores, Data Centers e entidades dinamicamente; b. Modelar servidores virtualizados: permite a modelação de hosts de servidor virtualizados, com políticas personalizáveis para provisionar recursos de host para máquinas virtuais; 2. Executar simulação <ol style="list-style-type: none"> a. Criar servidores, data centres e entidades dinamicamente durante a execução da simulação; 3. Analisar resultados da simulação <ol style="list-style-type: none"> a. Modelar recursos computacionais tendo em conta a energia gasta; <p>Guardar Simulação</p>

Tabela 29 - Simulador CloudSim

• **iCanCloud**

Simulador	iCanCloud (https://www.arcos.inf.uc3m.es/old/icancloud/Home.html)
Tecnologia Emergente	Cloud e Edge Computing

Tipo	Open Source
Descrição	<p>É uma plataforma de simulação destinada a modelar e simular sistemas de Cloud Computing. O principal objetivo é determinar o compromisso entre os custos incorridos e a taxa de desempenho durante as simulações, de um determinado conjunto de aplicações executadas num hardware específico e, em seguida, fornecer aos utilizadores informações sobre esses custos.</p> <p>Este simulador pode ser usado por vários utilizadores, desde utilizadores inexperientes a desenvolvedores de grandes aplicações de computação. É uma ferramenta escalável e flexível.</p>
Especificações técnicas	<ul style="list-style-type: none"> • Necessita de OMNeT++ e de INET <i>simulation frameworks</i>; • Instalação: <ul style="list-style-type: none"> ○ Fazer o download e descompactar a pasta ZIP; ○ Entrar na pasta IcanCloudGUI e executar o ficheiro; ○ Ao executar terá depois de escolher o caminho para o Icancloud e carregar em Open; • É obrigatório criar uma “Virtual Machine” com as peças de hardware que nos fornecem, escolhendo o CPU, a memória, o disco e o PSU; • Oferece uma interface assim como uma Interface de Programação de Aplicações baseada em <i>Portable Operating System Interface</i> (POSIX) e uma biblioteca <i>Message Passing Interface</i> (MPI) adaptada para modelação e simulação de aplicações;
Funcionalidades	<ol style="list-style-type: none"> 1. Preparar Simulação <ol style="list-style-type: none"> a. Modelar arquiteturas: arquiteturas de computação em cloud existentes e não existentes podem ser modeladas e simuladas. b. Configurar: oferece uma ampla gama de configurações para sistemas de armazenamento, que incluem modelos para sistemas de armazenamento local, remoto e paralelo. c. Integrar a cloud: tem um módulo hypervisor de cloud flexível que fornece um método fácil para integração e teste de intermediação em cloud nova e existente políticas. 2. Executar Simulação 3. Analisar Resultados da Simulação <ol style="list-style-type: none"> a. Obter consumo de energia: fornece métodos para obter o consumo de energia de cada componente de hardware no sistema de Cloud Computing. 4. Guardar Simulação

Tabela 30 - Simulador iCanCloud

• EMUSIM

Simulador	EMUSIM (http://www.cloudbus.org/cloudsim/emusim/)
Tecnologia Emergente	Cloud e Edge Computing
Tipo	Open Source
Descrição	<p>Este simulador combina emulação e simulação, utilizando o simulador CloudSim, de forma a permitir a utilização de modelos de artefactos de software, durante as simulações. Esta combinação permite extrair informações automaticamente do comportamento da aplicação (via emulação) e usa essas informações para gerar o modelo de simulação correspondente, de modo a ajudar os desenvolvedores a obter modelos mais precisos e estimar o desempenho e custo da aplicação.</p> <p>É principalmente útil quando o utilizador não tem conhecimentos do desempenho do software em diferentes níveis de paralelismo e simultaneidade, o que impede a utilização de simulação.</p> <p>O objetivo principal é testar o comportamento de uma aplicação via emulação, gerando modelos de simulação.</p>
Especificações técnicas	<ul style="list-style-type: none"> • Requer um sistema operativo Linux, assim como JAVA, podendo ser utilizado o Netbeans, XEN e CloudSIM, assim como acesso a um cluster cujos nós corram Xen's dom()). • Instalação <ul style="list-style-type: none"> ○ Fazer o download e desempacotar a pasta ZIP ○ Para correr o programa, é necessário de seguida integra-lo no <i>Netbeans</i> ○ No <i>Netbeans</i> é criado um projeto ○ Depois da criação do projeto foi adicionado à <i>librabries</i> as pastas que estão dentro da pasta lib ○ As pastas encontradas dentro da pasta scr são copiadas e emergidas com a pasta dentro do projeto <i>Netbeans</i> o que cria no projeto 4 novas classes que permitirão correr o programa <p>Houve problemas na conclusão da instalação. Ocorreram erros que não foi possível resolver.</p>

Tabela 31 - Simulador EMUSIM

• GROUDSIM

Simulador	GROUDSIM (http://dps.uibk.ac.at/projects/groudsim/)
Tecnologia Emergente	Cloud e Edge Computing
Tipo	Open Source
Descrição	O GroudSim é um simulador baseado em eventos, especialmente projetado para aplicações científicas em grade. Essa ferramenta de simulação trabalha nas áreas de IaaS da cloud e é extensível de maneira conveniente para oferecer suporte a modelos adicionais, como armazenamento em nuvem ou plataforma como serviço.
Especificações técnicas	<ul style="list-style-type: none"> • O simulador GroudSim faz parte do kit de ferramentas ASKALON; • Linguagem de programação java (J2SE 1.7); <p>Não foi possível a instalação pois ficheiros necessários já não se encontram disponíveis.</p>

Tabela 32 - Simulador GROUDSIM

• GreenCloud

Simulador	GreenCloud (https://gogreencloud.com/)
Tecnologia Emergente	Cloud e Edge Computing
Tipo	Open Source
Descrição	É um simulador <i>open source</i> para <i>Cloud Computing</i> . O objetivo principal do simulador é oferecer uma modelação detalhada da energia consumida pelos equipamentos de TI de um <i>data center</i> (servidores de computação, computadores de rede e links de comunicação). Isto deve-se ao facto do GreenCloud ser uma extensão do NS2 que implementa o modelo de referência de protocolo TCP/IP completo.
Especificações técnicas	<ul style="list-style-type: none"> • Necessita do Linux para funcionar; • Para o funcionamento do Ubuntu no software é necessário ter a virtualização ativada no computador; • Instalação: <ul style="list-style-type: none"> ○ Fazer o donwload e desempacotar o ficheiro ZIP ○ É necessário configurar e instalar o simulador, executando os códigos seguintes: ./configure, ./install ○ Executa-se o comando run no terminal para correr a simulação e para os resultados serem

	<p>apresentados através de uma Dashboard numa página Firefox.</p> <ul style="list-style-type: none"> • Pode ser usado para desenvolver novas soluções em monitorização, alocação de recursos, calendarização de carga de trabalho, bem como otimização de protocolos de comunicação e infraestrutura de rede. • Usa linguagem C ++ e OTcl, para implementar um único teste é necessário usar as duas linguagens diferentes.
<p>Funcionalidades</p>	<ol style="list-style-type: none"> 1. Preparar Simulação: <ol style="list-style-type: none"> a. Configurar simulação: é necessário configurar a caga do <i>datacenter</i> bem como o <i>Task Scheduler</i>. b. Gerir energia: o utilizador deverá preencher os campos: <i>Host, Network Switches, Virtualization</i> e escolher ou não a opção DVFS - <i>Dynamic Voltage and Frequency Scaling</i>. c. Configurar data center: o utilizador deverá preencher todos os campos apresentados no formulário referentes às configurações do <i>datacenter</i>. d. Selecionar o tipo de Tipologia: o utilizador deverá selecionar uma das quatro opções: <i>three-tier small; three-tier; tree-tier high speed; three-tier heterogenous small ou custom</i> e descrever os seus parâmetros de forma mais detalhada, isto é, o número de <i>switches</i> L3 na rede CORE, o número de <i>switches</i> agregados, <i>switches</i> numéricos na rede ACCESS e o número de <i>hosts</i> na rede. 2. Executar simulação: é executada uma simulação da distribuição de energia com base em todos os dados inseridos pelo utilizador, a qual é representada através de uma <i>dashboard</i>. 3. Analisar resultados da simulação: <ol style="list-style-type: none"> a. Resumir a Simulação: estão descritos os dados inseridos, um gráfico circular com a distribuição de energia, seis gráficos um resumo da simulação. b. Resumir a Gestão de Energia: é representado através de um gráfico circular com as percentagens relativas à distribuição de energia pela <i>cloud</i>. O gráfico circular é composto por quatro setores circulares diferentes cujo peso varia consoante os dados inseridos no formulário pelo utilizador: Energia do Server, Energia dos <i>switches</i> L3 na rede CORE, Energia dos <i>switches</i> numéricos na rede ACCESS e Energia dos <i>switchs</i> agregados. c. Simular o Datacenter: são apresentados dois gráficos referentes à simulação de energia do <i>datacenter</i>. O primeiro gráfico corresponde à média da distribuição de carga do <i>data center</i> durante o tempo de simulação e o segundo à média da distribuição de carga por server durante o tempo de simulação. 4. Guardar Simulação:

	<p>a. Consultar histórico de simulações: é possível ter acesso ao histórico de simulações, isto é, das <i>dashboards</i> apresentadas bem como aos dados que foram inseridos pelo utilizador. No histórico fica também registado a hora e o dia em que a mesma foi realizada.</p>
Objetivos de Aprendizagem	<ul style="list-style-type: none"> • Analisar a energia consumida por um sistema em cloud

Tabela 33 - Simulador GreenCloud

• CloudAnalyst

Simulador	CloudAnalyst (http://cloudsim-setup.blogspot.com/2013/01/running-and-using-cloud-analyst.html)
Tecnologia Emergente	Cloud e Edge Computing
Tipo	Open Source
Descrição	<p>É um simulador <i>open source</i> derivado do CloudSim; o seu objetivo principal é determinar o comportamento de aplicações de Internet de grande escala na <i>Cloud</i>. Permite modelar simulações em ciclo e realizar várias simulações com pequenas variações de parâmetros.</p> <p>Utiliza uma variedade de ferramentas e técnicas como:</p> <ol style="list-style-type: none"> 1. Produtos AWS Analytics; 2. Produtos do Google Cloud Analytics; 3. Serviços do Azure e produtos Microsoft relacionados
Especificações técnicas	<ul style="list-style-type: none"> • Requer um sistema operativo Linux como a linha de comandos; • Instalação: <ul style="list-style-type: none"> ○ Fazer o download e extrair os ficheiros da pasta ZIP ○ Clicar no arquivo “run.bat” ○ Na linha de comandos, executar a pasta CloudAnalyst ○ Clicar em “Show Region Boundaries” • Apresenta uma Interface gráfica que permite configurar e visualizar resultados, apresentando diversas configurações;
Funcionalidades	1. Preparar a simulação

	<p>a. Configurar Simulação: é possível manipular as configurações dos seguintes dados da simulação: Data Center, opções avançadas, opções principais;</p> <p>b. Configurar centro de dados: é possível adicionar, copiar e remover servidores;</p> <p>2. Executar a simulação</p> <p>3. Analisar os resultados da simulação</p> <p>a. Analisar através de gráficos e tabelas;</p> <p>4. Guardar simulações: o utilizador pode guardar os cenários de simulação e repeti-los com diferentes variações e os resultados da simulação são apresentados em forma de tabelas e gráficos;</p>
--	--

Tabela 34 - Simulador CloudAnalyst

• ANYLOGIC

Simulador	ANYLOGIC (https://www.anylogic.com/)
Tecnologia Emergente	Digital Twin
Tipo	Open Source
Descrição	O Anylogic é simulador <i>open source</i> de <i>Digital Twin</i> que permite a utilização de três métodos (evento discreto, baseado em agente, dinâmica do Sistema), combinando estes três métodos consegue-se simular sistemas complexos. O objetivo principal deste simulador é apoiar a decisão pela exploração de vários cenários de simulação de operações e serviços.
Especificações técnicas	<ul style="list-style-type: none"> • Suportado pelos seguintes sistemas: <ul style="list-style-type: none"> ○ Microsoft Windows 10, x86-32 ex64; ○ Microsoft Windows 8, x86-32 e x64; ○ Microsoft Windows 7 SP1, x86-32 e x64; ○ Apple Mac OS X 10.10 (Yosemite) ou mais recente; ○ Universal; ○ SuSE Linux, x64; ○ Ubuntu Linux 10.04 ou acima, x64; ○ Linux Mint 17 ou acima, x64; • Requisitos: <ul style="list-style-type: none"> ○ 1.5GB de espaço livre no disco;

	<ul style="list-style-type: none"> ○ 4GB a 8GB de memória; ○ processador moderno com pelo menos 2 cores para um desempenho otimizado; ● Instalação: <ul style="list-style-type: none"> ○ É necessário o preenchimento de um formulário para adquirir o simulador, sendo depois feito o download de ficheiro que tem 715MB; ○ É necessário aceitar os Termos e Condições e depois é dada a opção de instalar ou customizar. O processo de instalação dura cerca de 5 minutos e o programa está pronto a usar.
Funcionalidades	<p>6. Preparar a simulação:</p> <ol style="list-style-type: none"> a. Criar objetos: modelação envolvendo eventos e agentes, utilizando uma biblioteca de objetos padrão, onde é possível criar objetos personalizados sem recorrer à programação; b. Criar processos: permite a modelação de processos; c. Manipular parâmetros: vários cenários podem ser explorados variando os parâmetros. Podem ser inspecionados e consultados enquanto estão em ação e comparados entre si; <p>7. Executar a simulação:</p> <ol style="list-style-type: none"> a. Registar dados de entrada: há um registo de dados de entrada nos diversos processos e uma posterior análise detalhada; b. Executar modelo/processo: após a elaboração do processo há uma execução; c. Utilizar métodos: possui três métodos (evento discreto, baseado em agente, dinâmica do Sistema), sendo que os três métodos podem ser usados em qualquer combinação, para simular sistemas de negócios de qualquer complexidade. d. Integrar mapas: possibilidade de usar <i>GIS Maps</i> para os modelos de simulação. Ao estilo do <i>Google Maps</i>, permite localizar cidades, ruas, estradas, hospitais, lojas ao usar os dados dos mapas. Os elementos do modelo podem também navegar ao longo do mapa, mais concretamente das estradas existentes no mapa (reais). <p>8. Analisar Resultados da simulação:</p> <ol style="list-style-type: none"> a. Visualizar resultados: depois da execução será apresentado vários resultados, nos quais é possível analisar esses mesmos resultados e alterar os valores nos subprocessos de forma a que este processo seja o mais eficaz possível, ou seja tentativa de encontrar uma solução ótima e ideal; b. Visualizar em Cloud: oferece tecnologias em <i>cloud</i>, sendo possível aceder às simulações em qualquer equipamento e em qualquer lugar;

	<p>c. Visualizar em várias linguagens: fluxogramas de processo; gráficos de estado; gráficos de ação; diagramas de stock e fluxo;</p> <p>d. Visualizar em 2D/3D: permite converter fluxogramas em filmes iterativos com gráficos 3D e 2D. E permite importar modelos 3D personalizados, imagens, desenhos CAD e oferece a capacidade de integrar mapas GIS;</p> <p>9. Guardar Simulação;</p> <p>10. Outras ferramentas: possui bibliotecas de vários sectores, como por exemplo, biblioteca de tráfego rodoviário para movimentação de carros, camiões e autocarros em estradas, estacionamento e fábricas.</p>
Objetivos de Aprendizagem	<ul style="list-style-type: none"> • Compreender o funcionamento de uma linha de produção; • Analisar os resultados de mudanças nos processos; • Explicar novos cenários e mudanças nos processos existentes;

Tabela 35 - Simulador Anylogic

• SIMIO

Simulador	SIMIO (https://www.simio.com/applications/industry-40/Digital-Twin.php)
Tecnologia Emergente	Digital Twin
Tipo	Comercial
Descrição	<p>O SIMIO é um simulador comercial que recebe dados operacionais, históricos e ambientais, sendo possível construir vários cenários possíveis para prever e gerar resultados, sendo que suporta grandes volumes de dados.</p> <p>O objetivo principal deste simulador é permitir a manipulação de condições e fazer previsões a partir da digitalização da capacidade de produção da empresa.</p>
Especificações técnicas	<ul style="list-style-type: none"> • Instalação: <ul style="list-style-type: none"> ○ É necessário o preenchimento de um formulário e é enviado em seguida um email com todas as instruções para o seu download e instalação; ○ O ficheiro de download tem 533MB; ○ É feito um download de uma pasta zipada, com o simulador para instalar e dois PDF sendo que um deles é um pequeno guia de instalação e o outro é a licença; ○ Depois de fazer o download é executado o setup.exe;

	<ul style="list-style-type: none"> ○ De seguida foi aberto o programa já instalado; ● Devido ao facto de a instalação ser de versão gratuita, o programa apresenta certas restrições em relação à versão paga; ● É possível criar modelos bastante realistas em 3D recorrendo a animações realísticas do Google 3D Warehouse;
Funcionalidades	<ol style="list-style-type: none"> 1. Preparar simulação <ol style="list-style-type: none"> a. Criar modelo digital dos processos de produção b. Criar objetos: modelação envolvendo eventos e agentes, utilizando uma biblioteca de objetos padrão, onde é possível criar objetos personalizados sem recorrer à programação; c. Criar processos: permite a modelação de processos; d. Criar ligações com sistemas externos: permite interações com sistemas MES, MRP e ERP em tempo real e ainda permite otimização de agendas e reagendamentos quando necessário; e. Manipular parâmetros dos processos f. Registar dados de entrada: há um registo de dados de entrada nos diversos processos e uma posterior análise detalhada; 2. Executar Simulação <ol style="list-style-type: none"> a. Execução do modelo/processo: após a elaboração do processo há uma execução; 3. Analisar resultados da simulação <ol style="list-style-type: none"> a. Visualização de resultados: depois da execução será apresentado vários resultados, nos quais é possível analisar esses mesmos resultados e alterar os valores nos subprocessos de forma a que este processo seja o mais eficaz possível, ou seja tentativa de encontrar uma solução ótima e ideal; b. Análise de risco: permite reduzir riscos e melhorar as operações, pois é possível ocorrer uma análise do planeamento de risco; 4. Guardar simulação 5. Exemplos: contém vários exemplos de modelos que contêm uma descrição do que o modelo faz, como foi construído de modo a auxiliar o utilizador a compreender os resultados e o funcionamento do modelo;
Objetivos de Aprendizagem	<ul style="list-style-type: none"> ● Avaliar o desempenho de diferentes designs de processos; ● Testar as opções de digitalização; ● Explicar novos cenários e mudanças nos processos existentes;

Tabela 36 - Simulador SIMIO

- **SimCAD**

Simulador	SimCAD process Simulator (https://www.createasoft.com/)
Tecnologia Emergente	Digital Twin
Tipo	Comercial
Descrição	<p>É um simulador comercial dinâmico com visualização animada 2D/3D e realidade virtual. Permite que os utilizadores planeiem, otimizem e reorganizem processos e procedimentos enquanto otimizam <i>layouts</i>. Ou seja, é possível simular os processos existentes num processo produtivo, de forma a que seja melhor analisá-los e melhorá-los. É também possível modificar o modelo durante a simulação e analisar o impacto das mudanças, sem a necessidade de parar ou reiniciar a simulação.</p> <p>Este simulador fornece uma extensa lista de ferramentas de relatório e análise, que podem ser usados no processo de tomada de decisão.</p>
Especificações técnicas	<ul style="list-style-type: none"> • Suportado por: <ul style="list-style-type: none"> ○ Windows 7 SP1; ○ Windows 8; ○ Windows 8.1; ○ Windows 10; • Requisitos: <ul style="list-style-type: none"> ○ 8 GB de RAM; ○ 10GB de espaço de disco; ○ placa de vídeo padrão; • Requisitos recomendados: <ul style="list-style-type: none"> ○ Processador Multi-Core (i5, i7); ○ Windows 7 com SP1, 8.x, 10 (64 bits); ○ 16GB de RAM; ○ 30GB de espaço de disco; ○ placa de vídeo 3D necessária para 3D; • Requer o contacto com a equipa para ter acesso a uma demo com tempo limitado; • Instalação: <ul style="list-style-type: none"> ○ Para proceder ao download da solução existem duas opções iniciais: proceder ao preenchimento de

	<p>um formulário de forma a sermos contactados pelos vendedores da SimCad ou agendar uma demonstração do produto;</p> <ul style="list-style-type: none"> • Permite a ligação a diferentes bases de dados externas. <p>Falta de suporte por parte do fornecedor impossibilitou a continuação da exploração deste simulador.</p>
--	--

Tabela 37 - Simulador SimCad

• TOSSIM

Simulador	TOSSIM (http://networksimulationtools.com/tossim/)
Tecnologia Emergente	Internet of Things
Tipo	Open Source
Descrição	O simulador TOSSIM é um projeto que possui uma rede de sensores com grande número de pequenos dispositivos de hardware, sendo que o simulador projeta os tópicos de deteção sobre a rede. Este simulador forma manchas através da conceitualização de funções de hardware e modelagem de software.
Especificações técnicas	<ul style="list-style-type: none"> • É necessário fazer o download do sistema TinyOS para que o TOSSIM possa se executado. • Usa duas interfaces nas linguagens C ++ e Python. • Permite fazer debug usando gdb. • Possui plugins, sendo que a maior parte deles tem como principais características, mensagens de debug, definição de pontos de interrupção, leituras convertidas de analógico para digital, definição de localização. <p>Este simulador é bastante técnico o que dificultou a compreensão do mesmo.</p>

Tabela 38 - Simulador TOSSIM

• DPWSim

Simulador	DPWSim (https://sourceforge.net/projects/dpwsim/)
Tecnologia Emergente	Internet of Things
Tipo	Open Source
Descrição	<p>O DPWSim é um kit de ferramentas de simulação <i>open source</i> que suporta o desenvolvimento de aplicações de IoT, através da utilização de <i>Devices Profile for Web Services</i> (DPWS), ajudando os desenvolvedores a criar protótipos. É também possível gerir, criar, armazenar e simular carga, o que oferece uma alta flexibilidade aos utilizadores, nas simulações.</p> <p>O principal objetivo do simulador é criar dispositivos DPWS virtuais, que podem comunicar com outros dispositivos ou clientes via protocolos DPWS para ajudar os desenvolvedores destes dispositivos na criação de protótipos.</p> <p>Aplicações de IOT que funcionem com dispositivos virtuais fornecidos pelo DPWSim, podem imediatamente funcionar corretamente em dispositivos físicos sem qualquer alteração no código;</p>
Especificações técnicas	<ul style="list-style-type: none"> • Necessita de uma máquina com Java instalado; • Instalação: <ul style="list-style-type: none"> ○ Fazer o download e após o download, abrir um ficheiro com extensão jar. • Possui uma interface gráfica • É escrito em linguagem de programação Java, podendo ser executado em qualquer máquina que possua Java.
Funcionalidades	<ol style="list-style-type: none"> 1. Preparar Teste <ol style="list-style-type: none"> a. Criar um espaço de simulação; b. Criar novos dispositivos: existem várias maneiras de definir um novo dispositivo DPWS, desde a criação manual a gerar automaticamente a partir de um dispositivo físico; c. Adicionar dispositivos: Permite adicionar os dispositivos predefinidos no programa e ainda adicionar um dispositivo a partir de um ficheiro; d. Gerar dispositivos virtuais de dispositivos físicos DPWS; 2. Executar Teste <ol style="list-style-type: none"> a. Testar o estado dos dispositivos: para conseguirmos ver os vários estados do dispositivo temos de aceder a web e colocamos o nosso Ip e a porta, seguido de /dpwsim, onde vamos poder testar todos os estados que os nossos dispositivos podem ter;

- 3. Analisar resultados do teste
- 4. Guardar Teste

Tabela 39 - Simulador DPWSim

- iFogSim

Simulador	iFogSim (https://opensourceforu.com/2018/12/ifogsim-an-open-source-simulator-for-edge-computing-fog-computing-and-iot/)
Tecnologia Emergente	Internet of Things/ Cloud e Edge Computing
Tipo	Open Source
Descrição	<p>É um kit de ferramentas em <i>Open Source</i> para <i>Cloud Computing</i>, <i>Edge Computing</i> e IoT.</p> <p>O objetivo principal do simulador é modelar e testar redes de <i>Edge Computing</i>, IoT e de <i>Fog Computing</i> medir o impacto das técnicas de gestão de recursos tecnológicos em termos de latência, congestionamento de rede, consumo de energia e custo.” (Gupta et al., 2016).</p> <p>É uma extensão do simulador CloudSim, que é utilizado para a simulação de ambientes baseados em <i>Cloud Computing</i>.</p>
Especificações técnicas	<ul style="list-style-type: none"> • É suportado pelos seguintes sistemas: <ul style="list-style-type: none"> ○ Windows 7,10 [64-bit]; ○ MacOS 10.12; ○ Linux; • É necessário instalar: NetBeans IDE 8.2, Eclipse IDE; • Instalação: <ul style="list-style-type: none"> ○ Fazer o download de uma pasta zip e descomprimir a pasta zip; ○ É necessário abrir um ambiente de programação, que pode ser ou o NetBeans ou Eclipse, e criar um novo Projeto Java. Depois de criar o novo projeto importar todas as pastas que do arquivo ZIP, para o projeto java. ○ Depois de importar, fazer <i>debug</i> do projeto, o que vai originar todos os ficheiros que necessários para correr o programa. ○ Se a instalação for bem-sucedida vai originar uma pasta denominada de org.fog.gui.example, abrir a

	<p>pasta e correr o documento FogGui.java.</p> <ul style="list-style-type: none"> • É constituído por 3 componentes básicos: <ul style="list-style-type: none"> ○ Físicos - incluem dispositivos de Fog Computing que agem como data centers, oferecendo recursos de memória, rede e computação; ○ Lógicos - são compostos por aplicações, como por exemplo AppModule e AppEdges; ○ Gestão - são constituídos por objetos de Controller e Module Mapping, que recolhem os dados sobre custo, uso de rede e consumo de energia, durante o período de simulação. • As suas bibliotecas estão escritas em java e podem ser executadas em qualquer ambiente baseado em java. • Tem à disposição um guia que ensina a instalar.
Funcionalidades	<ol style="list-style-type: none"> 1. Preparar simulação <ol style="list-style-type: none"> a. Criar um projeto; b. Criar a topologia de rede; c. Criar entidades e sensores: inclui gateways e máquinas virtuais em cloud; d. Configurar entidades e sensores; e. Interligar diferentes entidades; 2. Executar Simulação 3. Analisar Resultados da Simulação <ol style="list-style-type: none"> a. Calcular consumo de energia; b. Visualizar latência média; 4. Guardar Simulação

Tabela 40 - Simulador iFogSim

- Cooja

Simulador	Cooja (https://sourceforge.net/projects/contiki/files/Instant%20Contiki/)
Tecnologia Emergente	Internet of Things
Tipo	Open Source
Descrição	<p>O Cooja é um simulador de IoT baseado em java, através do qual é possível testar o <i>Contiki</i>, um sistema operativo em <i>Open Source</i> para IoT, usado em projetos académicos, para criação de redes sem fios entre sensores, pois permite ligar à Internet microcontroladores de baixo custo e também de baixo consumo energético (PPLWare, 2019).</p> <p>O principal objetivo do simulador é facilitar a aprendizagem sobre redes IoT à base de sensores e outro equipamento de baixo custo.</p>
Especificações técnicas	<ul style="list-style-type: none"> • Suporta todos os sistemas operativos; • Requisitos: <ul style="list-style-type: none"> ○ 2,5GB de armazenamento; • Instalação: <ul style="list-style-type: none"> ○ Fazer o download e descompactar a pasta zipada; ○ Abrir a pasta “InstantContiki2.7” e selecionar o ficheiro Instant_Contiki_Ubuntu_12.04_32-bit.vmx. ○ Será apenas necessário executar (“clicar”) no software com o nome de “Cooja” e abri-lo. • Oferece uma interface gráfica onde é possível observar as simulações feitas, ou seja, é possível visualizar a troca de dados entre sensores, e os dados provenientes destes, assim como a sua representação gráfica. • Permite a simulação de diferentes plataformas e níveis, ou seja, tanto o hardware como o software (IGI Global Disseminator of Knowledge, n.d.), ou seja, é possível fazer a simulação da rede de sensores no Cooja, e quando a simulação estiver configurada conforme desejado fazer o seu <i>uploading</i> para o hardware.
Funcionalidades	<ol style="list-style-type: none"> 1. Preparar teste <ol style="list-style-type: none"> a. Criar nodos de rede. 2. Executar teste <ol style="list-style-type: none"> a. Iniciar, suspender, recarregar e executar passos da simulação. b. Mostrar em que instante da simulação se encontra e a sua velocidade; c. Tirar notas acerca da simulação; d. Monitorizar o resultado dos nodos e uma linha de tempo de mensagens e eventos da simulação;

	<p>3. Analisar resultados do teste:</p> <p>a. Visualizar a localização de cada nodo na rede: verificar a localização de cada nodo e a sua posição por coordenadas, de forma a também ter uma percepção das distâncias entre estes.</p> <p>b. Verificar o estado de cada nodo, incluindo LEDs, mote IDs, etc;</p> <p>4. Guardar teste</p>
--	--

Tabela 41 - Simulador Cooja

• WSNNet

Simulador	WSNet (http://wsnet.gforge.inria.fr/index.html)
Tecnologia Emergente	Internet of Things
Tipo	Open Source
Descrição	O WSNNet é um simulador open source orientado a eventos para redes sem fio, mas que também pode ser usado para a IoT. É capaz de simular nós com diferentes fontes de energia, modelos de mobilidade, interfaces de rádio, aplicações e protocolos de roteamento. O objetivo principal do simulador é compreender o comportamento e impacto da rede em cenários reais.
Especificações técnicas	Não foi possível o download do simulador e por consequente não foi possível a sua instalação.

Tabela 42 - Simulador WSNNet

• CupCarbon

Simulador	CupCarbon (http://cupcarbon.com/)
Tecnologia Emergente	Internet of Things
Tipo	Open Source
Descrição	O CupCarbon é um simulador de IoT <i>open source</i> que foi originalmente projetado como um simulador com forte ênfase no suporte à mobilidade de nós geográficos tendo por base ambientes do mundo real. O objetivo principal do simulador é compreender a mobilidade de nós geográficos em ambientes urbanos inteligentes para que por exemplo seja possível controlar e monitorar variáveis como o impacto ambiental que tem

	as deslocações dos veículos a combustível, ou outros usos mais complicados como a gestão do tráfego através de semáforos inteligentes.
Especificações técnicas	<ul style="list-style-type: none"> • É necessário possuir uma versão Java a executar corretamente no computador e também assegurar que se possui um certo conhecimento acerca dos comandos e regras da linguagem SenScript; • Instalação: <ul style="list-style-type: none"> ○ Fazer o download de uma pasta zipada e fazer a extração da pasta; ○ Para ter acesso ao simulador basta executar o ficheiro capcarbon.jar, que se encontra dentro da pasta; • As redes podem ser projetadas e prototipadas por uma interface ergonómica, usando a estrutura OpenStreetMap (OSM) para implementar sensores diretamente no mapa; • Oferece a possibilidade de simular algoritmos e cenários em várias etapas; • Disponibiliza guias e tutoriais para auxiliar, sendo que o guia inclui 17 exemplos básicos e 4 avançados; • Integra a API do OSM (Open Street Maps) e do Google Maps, permitindo a simulação de locais reais e utilizando de um fundo uniforme. Para além disso, através da API Google Elevation é ainda possível obter uma visualização da elevação do solo.
Funcionalidades	<p>5. Preparar Teste</p> <ol style="list-style-type: none"> a. Criar um novo projeto; b. Abrir um projeto recente; c. Criar cenários de mobilidade: permite o desenho de cenários de mobilidade e a geração de eventos naturais, como incêndios ou fugas de gás, assim como a simulação de veículos e objetos voadores. d. Criar etapas e algoritmos; e. Criar objetos de simulação; f. Configurar os nós da rede: permite a configuração dos nós dinamicamente, ou seja, permite dividir os nós em redes separadas, ou unir em redes que sejam diferentes. g. Adicionar nós do sensor, estações base (Sinks), gases(atmosféricos), tempo/meteorologia, nós de sensor aleatoriamente; h. Integrar Open Street Maps e do Google Maps e Google Elevation; i. Parametrizar dispositivos: <ol style="list-style-type: none"> i. Network Information: encontram-se dados como número de sensores, número de sensores marcados, número de sensores isolados. ii. Radio Parameters: possui parâmetros para controlo do radio.

	<p>iii. Devices & Objects: contém a lista dos dispositivos móveis criados e conectados à rede, assim como outros objetos.</p> <p>iv. Device Parameters: contém os parâmetros e os diferentes valores dos vários dispositivos ligados como por exemplo latitude, longitude, raio.</p> <p>6. Executar Teste</p> <p>a. Visualizar objetos em 2D ou 3D</p> <p>b. Parar a Simulação;</p> <p>c. Perceber o consumo de energia despendido na simulação;</p> <p>d. Executar códigos específicos através da SenScript window.</p> <p>7. Analisar Resultados do Teste</p> <p>a. Análise de dados geográficos: analisar dados relacionado com os mapas, distâncias e tempo.</p> <p>b. Melhorar a simulação: contém diversas funções/algoritmos para melhorar a simulação, ou seja, torná-la mais eficiente e eficaz, produzindo assim melhores resultados.</p> <p>8. Guardar o Teste.</p>
Objetivos de Aprendizagem	<ul style="list-style-type: none"> • Compreender a mobilidade dos nós geográficos em ambientes urbanos inteligentes; • Testar opções de planeamento/gestão urbana.

Tabela 43 - Simulador CupCarbon

• **SenseSim**

Simulador	SenseSim ()
Tecnologia Emergente	Internet of Things
Tipo	Open Source
Descrição	<p>O SenseSim foi projetado para simular a rede de sensores como um sistema autónomo e auto-organizado, incorporado no ambiente onde podem ocorrer muitos fenómenos diferentes. O objetivo principal deste simulador é compreender o comportamento e impacto da rede IoT nos processos do mundo real/externo.</p> <p>No SenseSim os dispositivos podem construir uma rede heterogénea na qual alguns deles podem ser sensores simples com recursos básicos de perceção e comunicação, mas também podem ser suportados por alguma "inteligência" fornecida por macroprogramas</p>

Especificações técnicas	Não foi encontrada nenhuma opção para download deste simulador.
-------------------------	---

Tabela 44 - Simulador SenseSim

• OMNeT++

Simulador	OMNet++ (https://omnetpp.org/)
Tecnologia Emergente	Internet of Things
Tipo	Open Source
Descrição	<p>O OMNeT ++ é um simulador open source discreto baseado em eventos, desenvolvido em C ++ e usado para simular redes de comunicação, sistemas distribuídos e outras aplicações de multiprocessadores.</p> <p>Este simulador foi desenvolvido para preencher a lacuna entre simuladores orientados para a pesquisa e de código aberto, suportados pela comunidade, como o NS3, e simuladores comerciais caros, como o OPNET.</p> <p>O principal objetivo deste simulador é testar a capacidade de tráfego de redes complexas.</p>
Especificações técnicas	<ul style="list-style-type: none"> • Oferece uma plataforma genérica e flexível para desenvolver estruturas de simulação dedicadas a sistemas de TI complexos, como redes sem fio e sensores, as pilhas clássicas de IP e IPv6, redes de filas, redes óticas e várias arquiteturas de hardware; • Suportado pelos seguintes sistemas: <ul style="list-style-type: none"> ○ Windows 7, 10 [64-bit]; ○ MacOS 10.12; ○ Linux; • Instalação: <ul style="list-style-type: none"> ○ Fazer o download e selecionar o ficheiro certo de acordo com o sistema operativo; ○ Aquando do download, é preciso escolher o local onde para guardar a pasta e cujo caminho completo não contenha nenhum espaço. Seguidamente, extraia o arquivo zip; ○ Abrir a pasta “omnetpp-5.6.1” e selecionar o ficheiro mingwenv.cmd. Será exibida uma consola com a shell bash; ○ Os seguintes comando deverão ser inseridos na consola: \$./configuer, \$make ○ Será necessário verificar se a instalação foi efetuada corretamente, e nesse sentido deverá testar todas as amostras existentes e verificar se elas estão a funcionar corretamente. Como exemplo, inserindo os

	<p>seguintes comandos na shell bash: \$ cd samples / aloha , \$./aloha</p> <ul style="list-style-type: none"> ○ De seguida, clicar em “Ok” para definir o nome de configuração • Contém editores, visualizações e outras ferramentas para ajudar no desenvolvimento do código C++. Os arquivos C++ são abertos no IDE no editor de código-fonte C++. O IDE também permite iniciar o processo de construção, configurar essa mesma construção, iniciar simulações e realizar o <i>debug</i> ao modelo; • Inclui representação multidimensional eficiente e poderosa de sinais de rádio, que permite simular com precisão fenômenos como conversação cruzada e coexistência. Os componentes de visualização associados podem exibir mapa de calor, espectro, espectrograma, etc. • Possui bibliotecas que podem ser usadas em várias camadas, incluindo a criação de geradores de tráfego e outras aplicações.
Funcionalidades	<ol style="list-style-type: none"> 1. Preparar teste <ol style="list-style-type: none"> a. Criar projeto b. Criar modelo da Rede (TXC.NED) <ol style="list-style-type: none"> i. Definir componentes da rede: o editor NED do OMNeT ++ IDE possui dois modos, Design e Source. No primeiro a topologia pode ser editada graficamente, já no segundo pode ser editado diretamente como texto. Primeiramente deveser colocado o código no modo Source, e posteriormente visualizados os módulos criados no modo Design. ii. Montar unidades c. Implementar funcionalidades (TXC.CC): é necessário criar um ficheiro txc.cc através do caminho: New Source File e, de seguida, preencher com o código desejado. 2. Executar teste: será necessário criar um ficheiro omnetpp.ini, que informa o programa de simulação qual a rede a simular. Para isso, será necessário seguir o caminho File/New/Initialization file (INI). Quando criado, terá de ser associada, em apenas 2 linhas de código. 3. Analisar resultados do teste 4. Guardar Teste

Tabela 45 - Simulador OMNeT++

- Bevywise

Simulador	Bevywise (https://www.bevywise.com/iot-simulator/)
Tecnologia Emergente	Internet of Things
Tipo	Comercial
Descrição	O objetivo principal do simulador é explorar redes dinâmicas de milhares de dispositivos que ajudam a entender como vários dispositivos dotados de tecnologia IoT interagem.
Especificações técnicas	<ul style="list-style-type: none"> • Possui uma versão grátis com as mesmas funcionalidades que a versão paga e dura 30 dias; • Requisitos: <ul style="list-style-type: none"> ○ processador de 1 GHZ; ○ 512 MB de memória; ○ Database – MySQL – para armazenamento da rede simulada; • Sistemas operativos suportados: <ul style="list-style-type: none"> ○ Windows 7, 8 & 10; ○ Windows Server 2012 & 2016; ○ Ubuntu, Mac & Raspberry pi; • Instalação: <ul style="list-style-type: none"> ○ Para fazer o download é preciso preencher um formulário com as informações: nome, mail, número e o sistema operativo; ○ Depois é preciso fazer o download e após terminada a transferência, é iniciada a instalação através de um executável, em que é necessário concordar com os termos de serviço, escolher o caminho de instalação, e finalmente esperar que a instalação seja concluída; ○ Terminada a instalação, para correr o programa é necessário navegar até à página de instalação e correr um ficheiro .bat lá presente, que abrirá o programa no browser predefinido; • Para começar a desenvolver um ambiente é necessário criar um dispositivo uma espécie de sensor, uma componente da rede a simular; • A criação de eventos necessita de ser configurado com os seguintes parâmetros: <ul style="list-style-type: none"> ○ Tópico: é um formato de sequência UTF-8, usado pelo broker para filtrar mensagens para cada cliente conectado. Um tópico consiste em um ou mais níveis de tópico e cada nível de tópico é

	<p>separado por uma barra (separador de nível de tópico);</p> <ul style="list-style-type: none"> ○ QoS: garantia que os eventos chegam ao assinante; ○ Reter mensagem: reter significa manter ou reter algo. Da mesma forma, em eventos, se o sinalizador de retenção for um, o broker manterá ou manterá os detalhes e mensagens correspondentes desses eventos. Portanto, ajudará os clientes recém-inscritos a obter uma atualização de status; ○ Tipo de eventos: formato dos eventos ou mensagens a serem publicados. O simulador suporta dois tipos de eventos ou tipo de mensagem: eles são Texto [mensagem normal] e mensagem JSON; ○ Variante: define como os eventos serão enviados ao assinante; <ul style="list-style-type: none"> • Suporta todas as especificações MQTT, e como todos os níveis de QoS, retém e mantém mensagens, autenticação MQTT e sessão limpa; • É capaz de conectar-se ao hub IoT do Azure, ao núcleo da AWS IoT, a qualquer plataforma IoT como a plataforma Bevywise IoT e a qualquer outra aplicação ou agente do MQTT; • Permite criar <i>templates</i> para os dispositivos e também criar redes dinâmicas, sendo que suporta valores dinâmicos para cada parâmetro introduzido • Possui uma interface que permite criar modelos, redes, adição em massa de dispositivos, configurar mensagens e padrões de comportamento;
Funcionalidades	<ol style="list-style-type: none"> 1. Preparar Teste: <ol style="list-style-type: none"> a. Criar uma nova rede de dispositivos: uma Network é uma rede IoT a simular, ou seja, este programa permite que sejam criadas várias simulações concorrentes, embora apenas possa ser executada uma de cada vez; b. Visualizar rede: permite visualizar as Networks já existentes e os templates; c. Criar templates para dispositivos e mensagens: permite a criação de templates para os dispositivos simulados com tópicos e mensagens específicas que ajudam a construir simulações mais rapidamente; d. Configurar e customizar mensagens: permite a configuração e customização de mensagens enviadas pelos dispositivos, tanto em TEXT como em JSON; e. Criar cenários reais: permite criar um cenário real com uma resposta automática usando a simulação de comportamento, um mecanismo de resposta automática que responderá ao assinante com base nas mensagens e tópicos recebidos. f. Gerir compatibilidade de conexão: permite a conexão com o Microsoft Azure IoT Hub, Amazon

	<p>Web Services IoT Core, Bevywise IoT platform assim como todas outras aplicações MQTT.</p> <ol style="list-style-type: none"> 2. Executar Teste 3. Analisar Resultados do Teste <ol style="list-style-type: none"> a. Alterar definições dos dispositivos; 4. Integrar interface Python: permite a integração de uma interface Python o que possibilita a utilização de um algoritmo próprio, customizando o simulador ainda mais; 5. Guardar teste: é possível também guardar as múltiplas redes simuladas numa base de dados MySQL e reutilizá-las.
--	--

Tabela 46 - Simulador Bevywise

• IOTify

Simulador	IOTify (https://iotify.io/iot-network-simulator/)
Tecnologia Emergente	Internet of Things
Tipo	Comercial
Descrição	<p>É um simulador comercial de IOT que permite desenvolver soluções de IOT na <i>Cloud</i>, em grande escala e permite gerar tráfego personalizado a partir de milhares de <i>endpoints</i> virtuais e testar a plataforma em termos de escala, segurança e confiabilidade.</p> <p>O objetivo principal do simulador é identificar e corrigir problemas de todo o sistema.</p> <p>Quando executada a simulação são obtidos resultados que podem ser explorados segundo diferentes parâmetros:</p> <ul style="list-style-type: none"> • Breve descrição da simulação em que é possível ver a largura de banda, o resultado, o número de repetições entre outros dados; • A atividade da rede em que é possível visualizar, no espaço de tempo da simulação, o contador de eventos; • A latência no envio de pacotes quer seja o valor mínimo, máximo e médio; • Estado, descreve o estado do objeto em cada cliente e iteração. É possível guardar este estado num csv.
Especificações técnicas	<ul style="list-style-type: none"> • A aplicação vai sugerindo <i>hardware</i>, protocolos e APIs para as aplicações IoT. • O simulador é utilizado na <i>cloud</i> - acedendo ao URL que recebido através do email, é efetuado o registo no site e por consequente ter acesso ao simulador. • Após a criação de um sistema neste simulador, o utilizador tem a possibilidade de o transferir para hardware

	<p>real, simplificando assim o processo de implementação das soluções desenvolvidas;</p> <ul style="list-style-type: none"> • A simulação é feita se segundo um <i>template</i>. Poderá ser um <i>template</i> em branco e definido pelo utilizador ou podem ser usadas cenários/simulações já existentes nomeadamente: <i>Smart Home</i>, <i>Smart Watch</i>, máquina de lavar louça, carro. Existe a possibilidade de selecionar um <i>template</i>, de entre as várias possibilidades; • É possível também escolher o tipo de conexão como por exemplo, HTTP, MQTT, assim como conectores de IoT, como o AWS ou o Azure;
Funcionalidades	<ol style="list-style-type: none"> 1. Preparar Teste: <ol style="list-style-type: none"> a. Definir um template, com código fonte; b. Escolher o tipo de conexão: HTTP, MQTT, etc; c. Escolher o protocolo de conectividade: MQTT e http, assim como conectores de IoT, como o AWS ou o Azure; d. Especificar os parâmetros para a simulação: o número de clientes ou a marcação do início da simulação numa determinada hora; 2. Execução do Teste 3. Análise dos resultados do teste <ol style="list-style-type: none"> a. Observar dados: a largura de banda; o resultado, o número de repetições, a latência no envio de pacotes quer seja o valor mínimo, máximo e médio; b. Ver atividade da rede: é possível visualizar, no espaço de tempo da simulação, o contador de eventos; c. Ver estado: descreve o estado do objeto em cada cliente e iteração. É possível guardar este estado num csv; d. Visualizar métricas criadas com a sua simulação; 4. Guardar Teste: criar e editar bases de dados;

Tabela 47 - Simulador IoTify

- **IOTSim**

Simulador	IOTSim (https://sourceforge.net/projects/iotsim/)
Tecnologia Emergente	Internet of Things
Tipo	Comercial

Descrição	<p>O IOTSim é um simulador comercial de IoT e é uma extensão do simulador Cloudsim. Simula aplicações de IoT usando o modelo <i>MapReduce</i> (modelo de programação, da Google, capaz de processar grandes volumes de dados de forma paralela), com um alto grau de precisão e com baixo custo e tempo (Zeng et al., 2017).</p> <p>Este simulador permite que os investigadores analisem a maneira como uma aplicação de IoT, compatível com MapReduce, funciona num determinado ambiente (Zeng et al., 2017).</p> <p>Este simulador reproduz a mecânica do centro de dados (como configurações de máquinas virtuais, requisitos computacionais e custos), em vez de interações de rede de sensores.</p> <p>O objetivo principal é facilitar a experimentação várias configurações de dispositivos de IoT, recursos de hardware e estruturas de processamento de big data e coletar insights sobre o impacto de cada opção de projeto nas garantias de desempenho;</p>
Especificações técnicas	<ul style="list-style-type: none"> • É necessário ter instalado na máquina pessoal um IDE (ambiente de desenvolvimento integrado) que suporte a linguagem Java; • Instalação: <ul style="list-style-type: none"> ○ Importar um projeto zip, que foi descarregado anteriormente para o ambiente de desenvolvimento integrado; ○ É necessário alterar algumas configurações do projeto importado para orientar o projeto de forma a ter uma classe <i>main</i> através das propriedades do programa e definir uma classe <i>main</i> na partição de “run” do projeto. A classe main é criada pela instalação de dos pressupostos <i>maven</i> no momento da importação e instalação; • Suporta a simulação de processamento de <i>Big Data IoT</i> usando o modelo <i>MapReduce</i> ou modelo <i>steam</i> no ambiente de computação em <i>cloud</i>; • Permite a modelação e simulação de várias aplicações de IoT em larga escala para execução em simultâneo num <i>data center</i> em <i>cloud</i> partilhada; • Permite modelar atrasos de rede e armazenamento existentes no processamento de aplicações de IoT;
Funcionalidades	<ol style="list-style-type: none"> 1. Preparar Teste <ol style="list-style-type: none"> a. Criar servidores, Data Centers e entidades dinamicamente; b. Definir parâmetros; 2. Executar Teste 3. Analisar resultados do teste

	<p>a. Visualizar Métricas: tempo médio de execução, tempo mínimo e máximo de execução, custo da rede;</p> <p>4. Guardar Teste</p>
--	---

Tabela 48 - Simulador IOTSim

• **Netsim**

Simulador	Netsim (https://www.tetcos.com/)
Tecnologia Emergente	Internet of Things
Tipo	Comercial
Descrição	<p>O NetSim é um simulador comercial que pode ser usado para simular sistemas de IoT, de forma a testar o desempenho de aplicações reais numa rede virtual (Dean, 2017). O NetSim é usado para desenvolver protótipos de rede de computadores.</p> <p>O objetivo principal deste simulador é simular sistemas de IoT.</p> <p>Este simulador suporta várias origens e destinos e pode ser dimensionado para centenas de nós, sendo possível simular diversas situações com a ajuda dos cenários "What-if" e métricas de teste, como perda, atraso, erro, qualidade de serviço, entre outros cenários que podem ser implementados na ferramenta (Dean, 2017).</p>
Especificações técnicas	<ul style="list-style-type: none"> • As bibliotecas de protocolo estão disponíveis como código C aberto para modificação do utilizador; • Possui 3 versões, profissional, académica e standard. Cada versão tem as suas características.
Funcionalidades	<ol style="list-style-type: none"> 1. Criar uma Rede: permite a criação de uma rede com vários dispositivos para que seja simulada; <ol style="list-style-type: none"> a. Configurar Rede: a rede criada pode ser configurada conforme os requisitos adequados, definidos pelo desenvolvedor; b. Executar Rede: permite a simulação da rede criada e configurada conforme os requisitos impostos; 2. Criar um Workspace: o <i>workspace</i> consiste em binários NetSim, códigos-fonte de protocolo e também atua como um espaço para as experiências do utilizador; <ol style="list-style-type: none"> a. Importar e exportar um Workspace: permite importar e exportar o espaço de experiências do utilizador; 3. Simular Redes Diferentes

	<p>4. Gerar e Configurar Tráfego</p> <p>5. Analisar Resultados: depois da realização de uma simulação, permite a análise dos dados/resultados obtidos na simulação;</p> <p>6. Desenvolver o próprio código: permite o desenvolvimento por parte do utilizador, ou seja, o seu próprio código;</p>
--	--

Tabela 49 - Simulador NetSim

• **Ansys IoT Simulator**

Simulador	Ansys (https://www.ansys.com/campaigns/internet-of-things)
Tecnologia Emergente	Internet of Things
Tipo	Comercial
Descrição	<p>O Ansys IoT Simulator é um simulador comercial que tem como objetivo principal a criação protótipos virtuais de sistemas completos, para criar dispositivos IoT reais. Ajuda a otimizar os protótipos, o desempenho de componentes ou subsistemas individuais, e todo o sistema de uma só vez (ANSYS, n.d.-c).</p> <p>Esta ferramenta pode ser usada em vários campos, incluindo wearables e dispositivos médicos, drones, carros, equipamentos industriais e muito mais, ajudando o utilizador a criar dispositivos mais acessíveis e lucrativos (Dean, 2017).</p>
Especificações técnicas	<ul style="list-style-type: none"> • Apesar de ser um software comercial, oferece uma versão grátis para estudantes; • Para utilização académica é possível efetuar o download de 3 pacotes: <ul style="list-style-type: none"> ○ Ansys Discovery Live Student 2020 R1: integra a modelação de geometria baseada na tecnologia Ansys SpaceClaim. Sendo que permite aprender física, sem a complicação de aprender a utilizar uma ferramenta de simulação complexa. ○ Ansys AIM Student 2020 R1: integra modelação de geometria baseada na tecnologia Ansys SpaceClaim e em solucionadores estruturais de fluídos e eletromagnéticas. Trata-se de uma ferramenta de simulação intuitiva e de ponta; ○ Ansys Student 2020 R1: é o pacote baseado em Ansys Workbench bundle do Ansys Mechanical, Ansys CFD, Ansys Autodyn, Ansys SpaceClaim e Ansys DesignXplorer. Trata-se de uma boa opção quando os professores utilizam esta ferramenta ou caso os alunos já estejam familiarizados com a plataforma Ansys Workbench;

	<ul style="list-style-type: none"> • Cada versão está dividida em: uma pequena descrição, os passos de instalação, requerimentos do sistema, limitações, duração da licença, o que está incluído e as versões anteriores; • Suportado pelos seguintes sistemas operativos: Microsoft Windows 10, 64-bit; • Requisitos: <ul style="list-style-type: none"> ○ 4 GB RAM; ○ 25 GB espaço livre do disco rígido; ○ OpenGL-capable; • Instalação: <ul style="list-style-type: none"> ○ Fazer o download de uma pasta zipada; ○ De seguida é necessário descomprimir a pasta onde está o ficheiro setup para instalar; • Tem um pequeno guia de iniciação
Funcionalidades	<ol style="list-style-type: none"> 1. Preparar Teste <ol style="list-style-type: none"> a. Criar Teste; b. Abrir Teste; c. Parametrizar dispositivos; d. Parametrizar ambiente; e. Parametrizar visualização: <ol style="list-style-type: none"> i. Alterar a perspetiva da “Câmara”, ou seja, da visualização do projeto; 2. Executar Simulação 3. Analisar Resultados da Simulação <ol style="list-style-type: none"> a. Realizar análises a vários componentes do projeto b. Alteração de parâmetros: 4. Guardar Simulação <ol style="list-style-type: none"> a. Partilhar simulação; b. Imprimir Simulação; 5. Integrar outras ferramentas

Tabela 50 - Simulador Ansys

- QualNet

Simulador	QualNet (https://www.scalable-networks.com/qualnet-network-simulation)
Tecnologia Emergente	Internet of Things
Tipo	Comercial
Descrição	QualNet é um simulador comercial de IoT. É uma ferramenta capaz de reproduzir o comportamento de uma rede de comunicações real. O objetivo principal é avaliar o comportamento básico de uma rede e testar combinações de recursos de rede. Esta ferramenta de simulação de rede oferece um ambiente abrangente para projetar protocolos, criar e animar cenários de rede e analisar o seu desempenho.
Especificações técnicas	<ul style="list-style-type: none"> • Suportado por Windows e Linux; • Permite o uso de OTB, redes reais e visualizações de terceiros para aprimorar o modelo de rede.
Funcionalidades	<ol style="list-style-type: none"> 1. Modelar: permite a modelagem de equipamentos sem fio móveis em ambientes terrestres, marítimos e aéreos. 2. Executar: executa cenários "what-if" variando o modelo, a rede e os parâmetros de tráfego. 3. Simular redes: permite a simulação da rede, independentemente do tamanho, usando técnicas de computação paralela. 4. Comportamento: usa modelos avançados para garantir que as simulações representem um comportamento real.

Tabela 51 - Simulador QualNet

- SimpleIoTSimulator

Simulador	SimpleIoTSimulator(https://www.smplsft.com/SimpleIoTSimulator.html)
Tecnologia Emergente	Internet of Things
Tipo	Comercial
Descrição	O SimpleIoTSimulator é um simulador comercial que permite criar ambientes de IoT constituídos em muitos sensores e gateways. Este simulador suporta protocolos comuns de IoT, como CoAP e MQTT como um protocolo baseado em publicação/assinatura.

	<p>O objetivo principal deste simulador é permitir que os fornecedores de plataformas e <i>gateway</i> de IoT melhorem a qualidade do produto tendo particular atenção aos protocolos de comunicação:</p> <ul style="list-style-type: none"> • Os grupos de vendas podem criar demonstrações realísticas que destacam os recursos da aplicação de gestão para reduzir os ciclos de vendas; • Os grupos de desenvolvimento e controlo de qualidade podem fazer testes funcionais e de escalabilidade sem grande infraestrutura de teste; • Os serviços profissionais podem criar serviços direcionados usando ambientes de clientes duplicados; • Os departamentos de Suporte técnico podem reproduzir mais facilmente os problemas dos clientes nos seus laboratórios de teste;
Especificações técnicas	<ul style="list-style-type: none"> • Suporta vários protocolos de IoT como MQTT, MQTT-SN, MQTT-Broker, CoAP, HTTP/s cliente, HTTP/s server. • Os protocolos ipv4 e ipv6 são ambos suportados sensorialmente e o simulador inclui um suporte integrado para perdas de comportamento em ambientes limitados; • As alterações nos dados do sensor podem ser modeladas para criar diferentes ambientes físicos; • É possível simular redes LoRaWAN para demonstrar e testar o Software da Aplicação do Utilizador e os Servidores de Rede LoRa (notas de aplicação); • Pode ser usado pelos provedores de serviços de rede LoRa e pelos seus clientes.

Tabela 52 - Simulador SimpleIoT Simulator

• Gazebo

Simulador	Gazebo (http://gazebosim.org/)
Tecnologia Emergente	Robotic Process Automation
Tipo	Open Source
Descrição	O Gazebo é um simulador <i>open source</i> para robótica que permite testar rapidamente algoritmos, projetar robôs, executar testes de regressão e treinar o sistema de Inteligência Artificial usando cenários realistas. Em suma, o objetivo principal do simulador é projetar e testar robôs, usando cenários realistas.
Especificações técnicas	<ul style="list-style-type: none"> • O Ubuntu é o sistema operativo em que este simulador apresenta uma melhor utilização. No entanto, é possível instalá-lo no Windows e no MAC;

- Requer:
 - GPU dedicada (placas *Nvidia* tendem a funcionar bem no Ubuntu);
 - CPU que seja pelo menos uma Intel I5 ou equivalente;
 - pelo menos 500 MB de espaço livre em disco;
 - *Ubuntu Trusty* ou uma versão mais recente instalada;
- Instalação no windows:
 - É recomendável a utilização de uma Shell MinGW para edição do trabalho e utilizar a linha de comandos do Windows para configuração;
 - É necessário criar uma pasta onde guardar o trabalho;
 - Fazer o download das seguintes dependências para a respetiva pasta: ibcurl HEAD; dlfcn-win32; protobuf 3.4.1; zlib; zziplib 0.13.62; freeImage 3.x; boost 1.67.0; OGRE 1.10.12 rc1; bzip2 1.0.6; TBB 4.3; Qt 5.7.0; Freetype 2.4.0; QWT 6.1.22;
 - Descompactar cada uma delas na pasta “gz-ws”;
 - Instalar Cmake, garantir que se seleciona a opção “AddCMake to system path for all users”;
 - Instalar o Ruby 1.9 ou outra versão superior. Durante a instalação certifique-se que adiciona o Ruby no caminho da diretoria;
 - Clonar Ignition Math, Transport, Sdformat e Gazebo;
 - Abrir a linha de comandos do Windows e carregar o *setup* do compilador copiando e colando a respetiva linha: "C:\Program Files (x86)\MicrosoftVisualStudio\2017\Community\VC\Auxiliary\Build\vcvarsall.bat".
 - Na linha de comandos configure e faça um *build* do Ignition Math;
 - Na linha de comandos configure e faça um *build* do Ignition Transport;
 - Na mesma linha de comandos, configure e faça o *build* do Sdformat;
 - Na mesma linha de comandos, configure e faça o build do Gazebo
- Instalação Linux:
 - Para proceder à instalação, tem de se ir ao ambiente Linux, abrir a linha de comandos e colocar o seguinte comando: `curl -sSL http://get.gazebosim.org | sh`
 - De seguida, escreve-se gazebo na linha de comandos para executar o mesmo;
- Possui extensas ferramentas de linha de comando que facilitam a introspeção e controlo da simulação.
- Permite o desenvolvimento de plugins personalizados para robô, sensor e controlo ambiental. Os plug-ins

	<p>fornece acesso direto à API do Gazebo.</p> <ul style="list-style-type: none"> • Permite executar a simulação em servidores remotos e fazer a interface com o Gazebo através da transmissão de mensagens com base em <i>socket</i> usando o Google Protobufs.
<p>Funcionalidades</p>	<ol style="list-style-type: none"> 1. File <ol style="list-style-type: none"> a. Guardar simulações 2. Edit <ol style="list-style-type: none"> a. Editar e criar ambiente 3. Camera <ol style="list-style-type: none"> a. Alterar as várias perspectivas da câmara, bem como seus ângulos; 4. View <ol style="list-style-type: none"> a. Integração no modelo eixos, filtros e estruturas simples; b. Definir visualização da simulação: permite-nos alterar a visualização da grelha, a origem, colocar transparência nos elementos ou o wireframe, bem como visualizar as colisões que podem ocorrer, as juntas, centro de massa, inercias, contactos e links. 5. Window <ol style="list-style-type: none"> a. Abrir novas janelas para obter informações sobre comandos do gazebo; b. Colocar a janela em modo “Full Screen”. 6. Help <ol style="list-style-type: none"> a. Ajudar o utilizador 7. Janela de Elementos: <ol style="list-style-type: none"> a. Editar propriedades do modelo: aqui é possível encontrar todos os elementos da tab World que estão no ambiente do simulador, permitindo visualizar e modificar as suas propriedades. Também é possível alterar a visão da câmara. b. Inserir modelos: permite inserir os modelos conhecidos pelo gazebo na tab World. Para isso, basta arrastar o modelo (drag and drop) para o World; c. Visualizar camadas: aqui são exibidos os diferentes grupos de visualização disponíveis na simulação, caso existam. Uma camada pode conter um ou mais modelos. Ativar ou desativar uma camada exibirá ou ocultará os modelos nessa camada. 8. Executar simulação; <ol style="list-style-type: none"> a. Verificar estados da simulação: mostra o fator de tempo real da simulação, assim como o tempo

	<p>real e tempo de simulação, apresentando também o valor de FPS da renderização da simulação;</p> <p>b. Explorar a execução da simulação: permite selecionar, mover, rodar, escalar, fazer undo-redo, inserir cubos, esferas, paralelepípedos, assim como luzes, copiar, colar, alinhar dois elementos, e alterar a vista da câmara.</p> <p>9. Fazer debug do Código.</p>
--	--

Tabela 53 - Simulador Gazebo

• USARSim

Simulador	USARSim (https://sourceforge.net/projects/usarsim/)
Tecnologia Emergente	Robotic Process Automation
Tipo	Open Source
Descrição	<p>O USARSim é um simulador <i>open source</i> de alta fidelidade de robôs e ambientes baseados no jogo do Unreal Tournament.</p> <p>O objetivo principal deste simulador é testar e compreender situações de interação humano-robot e coordenação multi-robot</p>
Especificações técnicas	<ul style="list-style-type: none"> • Requisitos: <ul style="list-style-type: none"> ○ Sistema Operativo Windows 2000/XP; ○ Linux; ○ Software Unreal Tournament 2004 com a versão 3339 ou patches mais recentes; ○ espaço livre no disco de 5,5 GB; ○ memória RAM mínimo é de 128MB mas o ideal é de 256MB para cima; ○ gráficos: 64MB NVIDIA ou ATI Hardware T&L card; ○ processador Pentium III 1.0 GHz ou AMD Athlon 1.0 GHz ou mais rápido (Pentium® ou AMD 1.2GHz ou mais como recomendável); • Instalação: <ul style="list-style-type: none"> ○ Instalar Unreal Tournament, cuja pasta de instalação será mencionada como %UT2004%. ○ Instalar a versão original do <i>patch</i> V3369 do Unreal Tournament: http://data.unrealtournament.com/UT2004-WinPatch3369.exe ○ Fazer o download dos ficheiros usarsim-2004

	<p>http://sourceforge.net/project/showfiles.php?group_id=145394</p> <ul style="list-style-type: none"> ○ Extrair os ficheiros anteriores para a pasta: %UT2004%\ ○ Compilar USARSim executando a <i>scrip</i> “make.bat” presente na pasta %UT2004%\System.
Funcionalidades	<p>Módulo Pyro</p> <ol style="list-style-type: none"> 1. Selecionar ambiente de simulação (arena ou mundo): permite selecionar a arena ou os mundos que se pretende adicionar no USARSim. 2. Selecionar robot: clique em “Robot” para selecionar o robô que pretende adicionar ao ambiente de simulação. Através do botão “View” pode visualizar o mesmo <ol style="list-style-type: none"> a. Definir controlo do robot: através da opção “Brain” pode definir a forma como o robô é controlado. Pode escolher Joystick.py para o controlar ou então a opção BBWander.py como se pode verificar na imagem seguinte. Esta última opção faz com que o robô se movimente pelo mundo de forma aleatória. 3. Executar robot: servidor irá executar e abrir o Unreal Tournament com as respetivas configurações. <p>Módulo Palyer</p> <ol style="list-style-type: none"> 1. Configurar ambiente de simulação (servidor, robots da simulação, navegação): é através do Player que é possível controlar os robôs do USARSim. No entanto, é necessário configurar o ficheiro de configuração do Player. Aqui irá definir o robô do USARSim, a localização do servidor, que tipo de robô é adicionado e onde é gerado. 2. Definir dispositivo (driver, dados de sensores): serão definidos dispositivos do USARSim, ou seja, definir os parâmetros que ajudam o Player a perceber onde e como é que os drivers estão ligados. 3. Executar simulação <p>Consola Unreal Tournament</p> <ol style="list-style-type: none"> 1. Corrigir erros: é possível aceder à consola do Unreal Tournament quando se inicia o USARSim e executar alguns comandos de debug como showdebug (escreve alguma informação de debug diretamente na janela do cliente). 2. Alterar propriedades do robot: permite abrir as propriedades de um ator do sistema, sendo possível alterar essas mesmas propriedades em tempo real. <p>USAR UI:</p> <ol style="list-style-type: none"> 1. Ligar simulação: basta clicar no botão “Conect” para conectar ao servidor. 2. Adicionar robot à simulação: para adicionar um robô à simulação deve introduzir o comando respetivo e

	<p>depois clicar em “Send”.</p> <p>3. Manipular robot</p> <p>SimpleUI</p> <ol style="list-style-type: none"> 1. Adicionar vídeo à simulação: permite utilizar imagens de vídeo na interface de controlo. 2. Controlar movimento do robot: possível dar-lhe ordens de comando a partir da caixa de comandos. 3. Gravar imagens: é possível monitorizar as imagens de vídeo, verificar o frame rate por segundo (FPS), e a largura (With) e altura (Height) da image. <p>Unreal Level Editor</p> <ol style="list-style-type: none"> 1. Adicionar objetos à simulação: é possível criar os mais variadíssimos objetos e mundos de acordo com os tipos de testes e simulações que pretender fazer. 2. Adicionar mapa: é possível editar mapas e adicionar novos objetos e atores ao mesmo. 3. Definir o posicionamento do objeto no mapa.
--	---

Tabela 54 - Simulador USARSim

• ARGoS

Simulador	ARGoS (https://www.argos-sim.info/)
Tecnologia Emergente	Robotic Process Automation
Tipo	Open Source
Descrição	<p>ARGoS é um simulador <i>open source</i> de robôs que opera em mais de um campo físico em simultâneo. É capaz de simular eficientemente conjuntos de robôs de qualquer tipo, em grande escala. Tem plug-ins que podem ser adicionados de modo a ser possível personalizar.</p> <p>Tem a possibilidade de usar vários mecanismos de física de diferentes tipos e atribuí-los a diferentes partes do ambiente. Os robôs podem migrar de um mecanismo para outro de forma transparente.</p> <p>O objetivo principal deste simulador é a criação de vários robôs e poder analisar a interação entre os mesmos.</p>
Especificações técnicas	<ul style="list-style-type: none"> • Suportado apenas em sistemas operativos como o Linux e Mac OSX e apenas é utilizado sob os termos de licença do MIT (Massachusetts Institute of Technology); • Instalação: <ul style="list-style-type: none"> ○ Fazer o seu download de um ficheiro; ○ Depois de feito o download verificar em que pasta o ficheiro se encontra;

	<ul style="list-style-type: none"> ○ Para ir para a pasta usar o comando “cd <nome_pasta>”; ○ Depois deste passo usar o comando “sudo apt install ./<nome_ficheiro>”; ○ Para conseguir usar o simulador é necessário instalar antes uns exemplos. Para isso é preciso usar o comando “git clone https://github.com/ilpincy/argos3-examples.git argos3-examples”. Caso dê erro usar o comando “sudo apt install git” e depois voltar a usar o comando acima; ○ De seguida, usar o comando “cd” para ir para a diretoria principal, e logo em seguida usar o comando “cd Transferências/argos3-examples” (verifique a pasta onde se encontra quando instalar os exemplos) para ir para a diretoria dos exemplos do simulador; ○ A seguir, criar uma diretoria com o nome “build” com o comando “mkdir build”; ○ Usar o comando “cmake ..” para preparar para compilar tudo. E depois o comando “make” para compilar, espere até chegar a 100%. ○ Por fim, poderá experimentar os exemplos fornecidos pelo simulador com o comando “argos3 -c experiments/<nome do ficheiro>” para descobrir os exemplos disponíveis vá à pasta experiments dentro da argos3-examples e encontrará lá todos os fornecidos que poderá experimentar ao usar o comando. <ul style="list-style-type: none"> • A interface de controlo é escrita em C++, no entanto é possível programar robôs noutras linguagens.
<p>Funcionalidades</p>	<ol style="list-style-type: none"> 1. Preparar simulação: <ol style="list-style-type: none"> a. Alocar recursos: permite ao utilizador adicionar e seleccionar os recursos e alocá-los onde for necessário; b. Implementar módulos: os robôs, sensores, atuadores, visualizações e motores físicos podem ser implementados como módulos definidos pelo utilizador, sendo que é possível várias implementações de cada tipo de módulo; c. Dividir o espaço: o espaço simulado pode ser dividido em subespaços, cada um dos quais é gerido por um mecanismo de física diferente; 2. Executar Simulação: <ol style="list-style-type: none"> a. Ajustar tempo de simulação: a possibilidade de a começar, parar, avançar ou até mesmo gravar a simulação; b. Parametrizar a câmara: permite controlar a direção a câmara usando o rato, ou usar os botões WASD. A função de mudança de câmara poderá ser realizada através do menu “Camera” e câmara mostra uma perspetiva diferente da simulação;

	<p>3. Analisar Resultados da Simulação</p> <p>a. Gravar simulação: é feita através de <i>frames</i>, ou seja, é uma gravação através de imagens tiradas a cada <i>frame</i>;</p> <p>4. Guardar Simulação;</p>
--	--

Tabela 55 - Simulador ARGoS

• UberSim

Simulador	UberSim (http://www.cs.cmu.edu/~robosoccer/ubersim/)
Tecnologia Emergente	Robotic Process Automation
Tipo	Open Source
Descrição	<p>O UberSim é um simulador <i>open source</i> baseado no mecanismo de física <i>ODE</i> e usa o <i>OpenGL</i> para gráficos. Foi criado especificamente com foco em pequenos robôs numa simulação de futebol de robôs, no entanto, o objetivo final é desenvolver um simulador para muitos tipos e tamanhos de plataformas de robótica.</p> <p>O objetivo principal deste simulador é explorar a interação multi-robot.</p>
Especificações técnicas	<ul style="list-style-type: none"> • Compatível com Linux; • Para a correta utilização do simulador é também preciso a instalação do OpenGL, um software de trabalho em computação gráfica; • Instalação: <ul style="list-style-type: none"> ○ Fazer download de uma pasta zipada. De seguida é feita a escolha de alocação da pasta onde a mesma irá ser descompactada; ○ De forma a proceder ao <i>buid</i> do simulador é necessário através da linha de comandos aceder ao diretório do simulador (<i>./ubersim</i>) e executar o comando <i>make</i>. Através deste comando irá ser originada uma subdivisão em árvore; • É necessário criar dois ambientes: <ul style="list-style-type: none"> ○ <i>UBERSIMCONFIG</i> precisa de ser definido como caminho para os arquivos de configuração localizado no diretório de configuração, ou seja, <caminho>/ubersim/config;

	<ul style="list-style-type: none"> ○ ODEHOME que deverá ser o caminho para o diretório .ode e o mesmo pode ser encontrado no seguinte site: http://opende.sourceforge.net/; • Para executar o simulador é necessário aceder ao diretório bin e executar ./ubersim. É aconselhável a instalação do software de futebol para robôs CMDragons. <p>As informações deste simulador eram muitas escassas. Desta forma, após a instalação do simulador não foi possível a sua execução uma vez que a compilação dos ficheiros não foi possível.</p>
--	--

Tabela 56 - UberSim

• EyeSim

Simulador	EyeSim (http://robotics.ee.uwa.edu.au/eyesim/)
Tecnologia Emergente	Robotic Process Automation
Tipo	Open Source
Descrição	O principal objetivo é testar o software do robô. É um simulador de vários robôs móveis com funcionalidade VR baseada no mecanismo de jogo Unity 3D que permite experimentar os mesmos programas executados em robôs reais.
Especificações técnicas	<ul style="list-style-type: none"> • É necessário Cygwin (EyeSim version), Xming; • Requisitos: sistema operativo Windows 8.1, 10; • Pode também ser instalado em Mas OS e Linux; • Instalação: <ul style="list-style-type: none"> ○ Fazer download do ficheiro e depois fazer download de uma pasta zipada com os ficheiros Cygwin e Xming; ○ Descompactar ambas as pastas; ○ Por último, fazer download da pasta zipada eyesimX; • Disponibiliza informações de como instalar em diferentes máquinas (Windows, linux) e quais as especificações necessárias para tal.
Funcionalidades	6. Preparar Simulação <ol style="list-style-type: none"> a. Open Terminal: é aberto uma linha de comandos que permite manusear o programa;

- b. **Load Sim:** permite abrir novamente a simulação e continuar o trabalho.
 - c. **Manipular o “Mundo”**
 - i. **Load World:** permite fazer Load do “Mundo” que estávamos a manusear para a simulação.
 - ii. **Save World:** é possível guardar o “Mundo” da simulação, podendo com toda a destreza ser completo posteriormente.
 - iii. **Create World:** permite Criar o “Mundo” para proceder à simulação.
 - iv. **Reset World:** é possível Criar um “Mundo” novo sem que se tenha que utilizar a própria funcionalidade de “Criar”, ou seja após a exploração da simulação caso queiramos abrir uma nova simulação podemos simplesmente dar Reset o que nos permite reconfigurar o “Mundo”.
 - d. **Parametrizar objetos:**
 - i. **Load Object:** contém um conjunto de objetos que podem ser utilizados para a simulação, no entanto também nos permite “carregar” para o “Mundo” novos objetos.
 - ii. **Add Object:** permite ao agente adicionar diferentes objetos na simulação.
 - iii. **View Objects:** permite observar todos os objetos em causa da simulação.
 - e. **Parametrizar robôs:**
 - i. **Load Robot:** o simulador contém com um conjunto de “robôs” que podem ser utilizados para a simulação, no entanto também é possível “carregar” para o mundo outros tipos de robôs.
 - ii. **Add Robot:** permite ao utilizador adicionar diferentes tipos de robôs na simulação.
 - iii. **View Robots:** permite observar os robôs que estão a ser utilizados na simulação.
- 7. Executar Simulação**
- a. **Pausar e Continuar Simulação**
 - b. **Manipular estado:**
 - i. **Add Market:** permite ao utilizador adicionar marcas na simulação.
 - ii. **Save State:** permite guardar o estado da simulação.
 - iii. **Load State:** após o estado ter sido guardado, é possível fazer download desse estado para a ferramenta.
 - c. **Manipular paredes:**
 - i. **Add Wall:** permite adicionar uma parede ao ambiente de simulação.
 - ii. **Remove Wall:** é possível remover a parede que foi previamente colocada na simulação.
 - iii. **Paint Wall:** permite fazer alterações estéticas na simulação, como por exemplo pintar a

	<p>parede.</p> <p>8. Analisar resultados da simulação</p> <p>a. Birdseye View: permite observar a simulação numa perspetiva de cima.</p> <p>b. Follow Object: permite observar de perto o objeto.</p> <p>c. Reset Camera: permite reconfigurar a câmara.</p> <p>d. View Log: permite ao utilizador observar algumas informações à cerca da simulação nomeadamente o tempo cronometrado de quando iniciou, bem como por exemplo quando observou um objeto de perto.</p> <p>9. Guardar Simulação: permite guardar o ficheiro da simulação em qualquer momento.</p> <p>10. Outras ferramentas:</p> <p>a. Help: permite ao agente procurar alguma ajuda ou alguma informação da simulação/ferramenta.</p> <p>b. RoBIOS API: abre ao utilizador uma página HTML, que contém algumas informações na livraria bem como alguns comandos de utilização.</p> <p>c. About: permite ao utilizador retirar informação sobre o nome da ferramenta e a versão.</p> <p>d. Changelog: abre uma espécie de livraria dentro da ferramenta</p>
Objetivos de Aprendizagem	<ul style="list-style-type: none"> • Compreender o potencial do uso de robôs em espaços com barreiras; • Testar software de robôs;

Tabela 57 - Simulador EyeSim

• SimRobot

Simulador	SimRobot (http://www.informatik.uni-bremen.de/simrobot/index_e.htm http://robotics.ee.uwa.edu.au/eyesim/)
Tecnologia Emergente	Robotic Process Automation
Tipo	Open Source
Descrição	O SimRobot é um simulador <i>open source</i> de robótica. Utiliza o ODE (<i>Open Dynamics Engine</i>) para simulações de física e o OpenGL para gráficos. O objetivo principal do simulador é ser usado para simulações de RoboCup.
Especificações técnicas	<ul style="list-style-type: none"> • Suportado por: <ul style="list-style-type: none"> ○ Windows 7/8; ○ Linux;

	<ul style="list-style-type: none"> ○ OS X 10.9/10; • É necessário instalar Microsoft Visual Studio Community 2019 Version 16.2.5. Instalar workload com C++ bem como packages MSVC v142 - VS 2019 C++ x86/x64 build tools, Windows 10 SDK 10.0.18362.0, e C++-ATL for v142 build tools (x86 & x64). • Instalação: <ul style="list-style-type: none"> ○ É necessário definir duas variáveis de ambiente. SIMROBOTHOME deve ser definido como diretoria raiz do SimRobot e PATH deve ser estendido para conter uma entrada \$SIMROBOTHOME/bin. Ambos sendo colocados no arquivo “.login”; ○ De seguida é necessário descompactar e preparar para a tarefa de compilação; ○ É necessário definir um link simbólico para utilização da linguagem de código C++; • Contém uma interface sendo que robôs personalizados também podem ser criados e adicionados.
Funcionalidades	<ol style="list-style-type: none"> 1. File <ol style="list-style-type: none"> a. Gravar e abrir uma simulação 2. View: permite formatar o ambiente de simulação, inclui sensores genéricos que os utilizadores podem personalizar. Isso inclui uma câmara, sensor de distância, um "pára-choques" para simular um sensor de toque e "estado do atuador" que retorna ângulos de juntas e velocidades de motores; 3. Simulation: permite executar a simulação <ol style="list-style-type: none"> a. Start: esta funcionalidade permite iniciar a simulação, pode também ser iniciado premindo a tecla F5; b. Reset: esta funcionalidade permite reiniciar a simulação, apagando todo o historial que estaria em execução anteriormente. Funciona como se iniciasse a simulação desde a origem; c. Step: esta funcionalidade permite uma execução da simulação mais parada, passando a executar por passos, um de cada vez, de forma estática. 4. Help: ajuda o utilizador

Tabela 58 - Simulador SimRobot

- WeBots

Simulador	WeBots (https://cyberbotics.com/)
Tecnologia Emergente	Robotic Process Automation

Tipo	Open Source
Descrição	Webots é um simulador <i>open source</i> que fornece um ambiente de desenvolvimento completo para modelar, programar e simular robôs. Portanto, o objetivo principal deste simulador é experimentar com robôs virtuais de forma a otimizar a construção de robôs e interações multi-robot.
Especificações técnicas	<ul style="list-style-type: none"> • Requisitos mínimos: <ul style="list-style-type: none"> ○ Windows com sistema de 64 bits; ○ processador dual core com 2 GHz; ○ 2 GB de RAM; ○ placa gráfica da NVIDIA ou AMD com 512 MB de RAM; • Requisitos recomendados: <ul style="list-style-type: none"> ○ Windows com sistema de 64 bits; ○ processador quad-core; placa gráfica da NVIDIA ou AMD recente; • Instalação: <ul style="list-style-type: none"> ○ É necessário fazer o download e executar o ficheiro e continuamente carregar em “next” até que a instalação seja concluída; • Permite que vários robôs sejam executados ao mesmo tempo e suporta as linguagens C/C ++, Java, URBI, Python, ROS e MATLAB. • Oferece várias opções de programação para controlar os robôs simulados. Também é possível programar um plugin de física para personalizar a física de uma simulação para adicionar, por exemplo, forças extras (vento, perturbações) ou manipular contatos de maneira diferente com, por exemplo, um modelo de atrito não uniforme. • Disponibiliza informação sobre como usar o simulador.
Funcionalidades	<ol style="list-style-type: none"> 1. Preparar Simulação <ol style="list-style-type: none"> a. Criar um projeto: permite ao utilizador criar simulações novas de raiz, ou seja, todos os campos necessários para a simulação funcionar estarão vazios. b. Criar um projeto com ajuda: o simulador ajuda o utilizador a criar um projeto, seguindo os passos indicados. c. Abrir mundos/simulações: permite ao utilizador abrir projetos que já tenha criado anteriormente. d. Criar animações HTML5 ou gravações da simulação;

	<p>e. Calibrar: o processo de calibração consiste em ajustar os parâmetros físicos de objetos e dispositivos para que eles se comportem como seus equivalentes físicos. Os utilizadores podem calibrar os seus próprios modelos para garantir resultados de simulação realistas;</p> <p>f. Editar código: possui um editor de código embutido, que permite ao utilizador criar tudo dentro do mesmo e testar em tempo real, tornando o seu trabalho mais fácil;</p> <p>2. Executar simulação</p> <p>a. Ajustar tempo de simulação: começar, parar, avançar e aumentar a velocidade para rápido ou muito rápido da simulação;</p> <p>3. Analisar resultados da simulação</p> <p>4. Guardar Simulação: permite ao utilizador guardar todo o seu trabalho realizado dentro do projeto onde se encontra.</p> <p>5. Bibliotecas: contém bibliotecas de modelos calibrados que o utilizador pode reutilizar nas suas próprias simulações ou enriquecer com as suas próprias criações.</p>
--	--

Tabela 59 - Simulador WeBots

• HEEDS

Simulador	HEEDS (https://www.redcedartech.com/index.php/solutions/heeds-software)
Tecnologia Emergente	Robotic Process Automation
Tipo	Comercial
Descrição	<p>O HEEDS tem como objetivo principal fornecer um ambiente que permita obter o melhoramento de projetos, de forma mais rápida e eficaz. Lida com os desafios automatizando o processo de análise, impulsionando o seu investimento em hardware de computação e procurando, eficientemente, soluções com melhor desempenho, fornecendo assim maneiras intuitivas de corrigir o desempenho do projeto.</p> <p>Permite definir os resultados desejados e este, tendo em conta os parâmetros determinados, executa várias tarefas automaticamente, de forma a encontrar soluções para desenvolver os produtos com mais eficiência, num curto prazo de tempo (MathWorks, n.d.).</p>
Especificações técnicas	<ul style="list-style-type: none"> • Suporta as arquiteturas 32 bits e 64 bits; • Requisitos: <ul style="list-style-type: none"> ○ Sistema operativo MS Windows; ○ Memória (RAM) de 4 GB ou mais;

	<ul style="list-style-type: none"> ○ 5 GB de espaço livre no disco rígido; ○ Processador Intel Pentium IV ou superior; ○ Tamanho total da instalação de 1 GB; ● Instalação offline/Instalação autónoma completa; ● Instalação: <ul style="list-style-type: none"> ○ Após realizar o download do arquivo de instalação, é necessário descompactar os arquivos; ○ No arquivo “install_HEEDS-MDO_2019.1_Win64”, deve ser iniciado o processo de instalação; ○ É necessário configurar algumas opções, como por exemplo a pasta onde instalar a solução, a licença e o respetivo nome, a linguagem; ● Fornece vários algoritmos de otimização de design como: SHERPA; SHERPA multi-objetivo (MO-SHERPA); Algoritmo genético; Programação quadrática sequencial;
Funcionalidades	<ol style="list-style-type: none"> 1. Preparar Simulação <ol style="list-style-type: none"> a. Criar projeto; b. Abrir projeto; c. Definir parâmetros: definir as variáveis e as respostas do projeto e as suas características. As variáveis são elementos que podem ser adicionados ao processo para realizar o estudo e pode possuir os estados: contínuo, dependente, discreto, constante ou texto. Podem ser configurados os valores esperados do processo para atingir os objetivos, denominados como respostas. <ol style="list-style-type: none"> i. Inserir valores: cada célula numa linha da tabela contém um campo para inserir um valor. Esses campos serão caixas de texto ou listas; ii. Associar valores: são associados os valores das variáveis e das respostas nos ficheiros de entrada e saída; d. Definir o tipo de estudo e o método do projeto: inclui tarefas para definir como é que o HEEDS MDO usará os processos num estudo de exploração de design. O tipo de estudo padrão é “Otimização” de parâmetros e o método de estudo padrão é SHERPA - soma ponderada de todos os objetivos. 2. Executar Simulação <ol style="list-style-type: none"> a. Marcar (vincular) os valores das variáveis e das respostas nos arquivos de entrada e saída; b. Definir e realizar as análises do processo para o projeto; 3. Analisar resultados da simulação

- a. **Monitorizar o estudo:** após a conclusão da definição do estudo para o projeto. Durante a execução, o HEEDS MDO exibe gráficos de tempo real que permitem monitorizar o progresso da execução
4. **Guardar simulação**

Tabela 60 - Simulador HEEDS

• **SimCAD**

Simulador	SimCAD process Simulator (https://www.createasoft.com/)
Tecnologia Emergente	Robotic Process Automation
Tipo	Comercial
Descrição	<p>É um simulador comercial dinâmico com visualização animada 2D/3D e realidade virtual. Permite que os utilizadores planeiem, otimizem e reorganizem processos e procedimentos enquanto otimizam <i>layouts</i>. Ou seja, é possível simular os processos existentes num processo produtivo, de forma a que seja melhor analisá-los e melhorá-los. É também possível modificar o modelo durante a simulação e analisar o impacto das mudanças, sem a necessidade de parar ou reiniciar a simulação.</p> <p>Este simulador fornece uma extensa lista de ferramentas de relatório e análise, que podem ser usados no processo de tomada de decisão.</p>
Especificações técnicas	<ul style="list-style-type: none"> • Suportado por: <ul style="list-style-type: none"> ○ Windows 7 SP1; ○ Windows 8; ○ Windows 8.1; ○ Windows 10; • Requisitos: <ul style="list-style-type: none"> ○ 8 GB de RAM; ○ 10GB de espaço de disco; ○ placa de vídeo padrão; • Requisitos recomendados: <ul style="list-style-type: none"> ○ Processador Multi-Core (i5, i7); ○ Windows 7 com SP1, 8.x, 10 (64 bits); ○ 16GB de RAM;

	<ul style="list-style-type: none"> ○ 30GB de espaço de disco; ○ placa de vídeo 3D necessária para 3D; • Requer o contacto com a equipa para ter acesso a uma demo com tempo limitado; • Instalação: <ul style="list-style-type: none"> ○ Para proceder ao download da solução existem duas opções iniciais: proceder ao preenchimento de um formulário de forma a sermos contactados pelos vendedores da SimCad ou agendar uma demonstração do produto; • Permite a ligação a diferentes bases de dados externas. <p>Falta de suporte por parte do fornecedor impossibilitou a continuação da exploração deste simulador.</p>
--	---

Tabela 61 - Simulador SimCAD

• ANSYS ACT

Simulador	ANSYS ACT (https://www.ansys.com/products/structures/ansys-act)
Tecnologia Emergente	Robotic Process Automation
Tipo	Comercial
Descrição	O ANSYS ACT é um simulador comercial que permite criar aplicações para automatizar os fluxos de trabalho rotineiros, personalizar processos especializados, desenvolver aplicações específicas para negócios e integrar ferramentas e dados de terceiros à linha de produtos ANSYS. (ANSYS, n.d.-a)
Especificações técnicas	<ul style="list-style-type: none"> • Apesar de ser um software comercial, oferece uma versão grátis para estudantes; • Podem ser instalados em qualquer computador que suporte MS Windows 64-bit; • Para utilização académica é possível efetuar o download de 3 pacotes: <ul style="list-style-type: none"> ○ Ansys Discovery Live Student 2020 R1: integra a modelação de geometria baseada na tecnologia Ansys SpaceClaim. Sendo que permite aprender física, sem a complicação de aprender a utilizar uma ferramenta de simulação complexa. ○ Ansys AIM Student 2020 R1: integra modelação de geometria baseada na tecnologia Ansys SpaceClaim e em solucionadores estruturais de fluídos e eletromagnéticas. Trata-se de uma ferramenta de simulação intuitiva e de ponta; ○ Ansys Student 2020 R1: é o pacote baseado em Ansys Workbench bundle do Ansys Mechanical,

	<p>Ansys CFD, Ansys Autodyn, Ansys SpaceClaim e Ansys DesignXplorer. Trata-se de uma boa opção quando os professores utilizam esta ferramenta ou caso os alunos já estejam familiarizados com a plataforma Ansys Workbench;</p> <ul style="list-style-type: none"> • Requisitos: <ul style="list-style-type: none"> ○ CPU Pentium® 4 2.0 GHZ or Athlon® 2000+ ou mais rápido; ○ Processador de 64-bit (x64); ○ 8GB RAM; ○ 4 GB de espaço livre no disco rígido; ○ placa de vídeo; ○ pelo menos 4GB RAM de vídeo (preferível 8G) no GPU; • Instalação: <ul style="list-style-type: none"> ○ Fazer o download de uma pasta zipada (demora cerca de 30 minutos); ○ De seguida é necessário descomprimir a pasta para uma diretoria temporária; ○ Deve-se depois procurar o ficheiro setup.exe, clicar com o botão direito do rato e selecionar a opção correr como administrador; ○ Por fim, é necessário escolher a diretoria onde irá ser instalado, sendo que terá de ser numa diretoria onde se tenha permissões de escrita; ○ Toda a instalação demorou cerca de 20 minutos; • Utiliza como linguagens de programação XML e IronPython; • Oferece um ambiente de desenvolvimento com modelos e aplicações predefinidas, para guiar o utilizador.
Funcionalidades	<ol style="list-style-type: none"> 1. Preparar Simulação <ol style="list-style-type: none"> a. Criar Simulação; b. Abrir simulação; c. Parametrizar objetos: <ol style="list-style-type: none"> i. Copiar, Snipe, Colar e Style Painter, que permite copiar estilos visuais de um objeto para o outro; d. Parametrizar ambiente: <ol style="list-style-type: none"> i. Exibir o ambiente de diferentes formas ii. “Atirar” uma face do projeto para onde quisermos, dentro da Janela de Design; iii. Adicionar várias formas ao projeto;

- iv. Escolher em que dimensão trabalhar (2D ou 3D);
- v. Fundir ou intercalar o projeto;
- vi. Adicionar planos, eixos, pontos, origens, equações, cilindros e esferas em 3D.

e. Parametrizar visualização

- i. Alterar a perspectiva da “Câmara”, ou seja, da visualização do projeto;
- ii. Adicionar novas janelas, cortar uma janela até 4 janelas de exibição ou mesmo trocar de janelas;
- iii. Alterar as configurações da grelha;

2. Executar Simulação

a. Manipular Objetos

- i. Inserir geometrias, imagens ou vídeos no projeto;
- ii. Adicionar tangentes aos objetos;
- iii. Alinhar e orientar objetos;
- iv. Medir objetos inseridos no projeto;
- v. Consultar as características de um objeto selecionado no projeto;
- vi. Realizar interseções nos objetos;
- vii. Alterar as faces e superfícies dos objetos sendo possível visualizar a face normal, em grelha, em curvatura, diédrica, esboços relativos ao eixo do Z e padrões de listas refletidas;
- viii. Comparar os resultados de desvio de superfícies entre 2 objetos.

3. Analisar Resultados da Simulação

a. Navegar pelas áreas problemáticas do projeto;

b. Detetar e corrigir falhas nas superfícies e detetar faces desaparecidas;

c. Corrigir curvas;

d. Realizar análises a vários componentes do projeto:

- i. Eliminar interferências entre objetos, faces dos objetos, margens e bordas do modelo;
- ii. Detetar faces deformadas, margens afiadas e espaços entre os objetos;
- iii. Criar, extrair, orientar, conectar ou cortar feixes;

e. Alteração de parâmetros:

- i. Simular o arrefecimento dos objetos através do conjunto de ferramentas fornecidas pelo IcePak;

	<ul style="list-style-type: none"> ii. Identificar objetos Icepak e simplificar objetos para o Icepak; iii. Criar superfícies nos corpos para simular aberturas, ventoinhas ou grelhas; iv. Converter um corpo fechado a um corpo equivalente em Icepak; v. Mostrar ou ocultar corpos Icepak, corpos que não sejam Icepak ou todos os corpos presentes no projeto. <p>f. Adicionar extras como notas, códigos de barras, tabelas;</p> <p>4. Guardar Simulação</p> <ul style="list-style-type: none"> a. Partilhar simulação; b. Imprimir Simulação; <p>5. Integrar outras ferramentas:</p> <ul style="list-style-type: none"> a. Adicionar geometrias de design assistido por computador (CAD); b. Criar uma superfície que se aproxime da geometria definida previamente; c. Extrair as curvas de secções de corpos facetados; d. Criar e identificar orifícios com base em valores padrão; e. Selecionar e mover um ponto de um objeto para a origem do referencial; f. Adicionar ferramentas. g. Realizar o download do pacote Keyshot, caso este ainda não esteja instalado no software; h. Esclarecer dúvidas sobre esta ferramenta; i. Aceder a uma galeria exemplo na internet com imagens tiradas a partir desta ferramenta;
--	--

Tabela 62 - Simulador ANSYS ACT

• **LEAPWORK Automation Platform**

Simulador	LEAPWORK Automation Platform (https://www.leapwork.com/)
Tecnologia Emergente	Robotic Process Automation
Tipo	Comercial
Descrição	LEAPWORK Automation Platform tem por objetivo principal testar a automatização de processos repetitivos, onde os utilizadores podem criar e executar automação de tarefas numa interface com o design baseado em fluxogramas (Leapwork, n.d.-b), sendo que não é necessário ter conhecimento técnico para definir os fluxos de automação, assim como não é necessário recorrer a código (Leapwork, n.d.-b).

Especificações técnicas

- Suportado pelos seguintes ambientes:
 - Windows 7,8.0 + 8.1 e 10;
 - Windows Server 2008 R2;
 - Windows Server 2012 e Server 2012 R2;
 - Windows Server 2016;
 - OSX versão 10.8 ou nas recentes;
- O *Controller* desempenha o papel de um *hub* no Leapwork e deve ser configurado de acordo com alguns fatores, como: número de fluxos de automação e a respetiva complexidade e duração, frequência de execução dos mesmos fluxos;
- Configuração de servidor standard com:
 - 16 GB RAM, SSD, SAN ou similar;
- Requerimentos:
 - 16 GB RAM;
 - 60 GB SSD;
- Na simulação de aplicações para o utilizador a máquina deve ser o mais similar possível às máquinas para os utilizadores. A configuração do hardware recomendada é:
 - min 8 GB RAM (recomendado 16 GB);
 - Intel i5, SSD;
- Para correr fluxos de automação em paralelo: no caso de execução de fluxos de automação web. Para uma execução de quatro browsers em paralelo, é recomendado:
 - 16 GB RAM;
 - SSD, SAN ou similares com 30 GB;
- São necessários 1921MB no hard drive para que a instalação seja possível;
- Tem duas opções aquando da instalação: aprender autonomamente ou com um especialista. Neste último caso é possível delinear objetivos, desenhar os primeiros fluxos de aprendizagem e projetar resultados.
- Instalação:
 - É preciso preencher um formulário com dados como nome e mail para conseguir a demo;
 - É mandado para o mail, um mail com um link onde é pedido uma password e automaticamente começa o download do simulador;

	<ul style="list-style-type: none"> ○ O download demora cerca de 30min; ○ Selecionar as componentes a instalar e as respetivas características, como por exemplo as pastas a guardar a aplicação no computador; ○ É necessário configurar algumas propriedades derivadas das opções selecionadas, tais como as portas de acesso dos controladores, da respetiva API e do agente; ○ É fornecida uma chave ao utilizador que permite proteger e ter acesso aos ficheiros da base de dados que serão criados; ● Disponibiliza tutoriais no site para que seja possível uma melhor aprendizagem da ferramenta; ● A subscrição após o download do pacote dura apenas um período de quatorze dias, não permitindo mais o seu acesso a partir do final da mesma <ul style="list-style-type: none"> ○ Apesar de ser uma ferramenta comercial é possível efetuar um free trial e desta forma ter um primeiro contacto com este tipo de softwares. Em termos académicos, caso o período de free trial se tenha esgotado, é possível entrar em contacto com o centro de apoio e eles oferecem uma extensão do prazo, sem colocar entraves.
Funcionalidades	<ol style="list-style-type: none"> 1. Preparar Simulação <ol style="list-style-type: none"> a. Definir fluxos de automação <ol style="list-style-type: none"> i. Criar fluxo: permite a criação de fluxos de automação (sem necessidade de desenvolver código), com design baseado em fluxogramas; ii. Definir ordem de execução e direção do fluxo: definir a ordem pela qual deve ser executado ou direcionado o fluxo; iii. Formatar visualização do fluxo: permite ao utilizador proceder à movimentação de fluxos ou subfluxos, ajustando o zoom de modo a que o fluxo ou subfluxo fique na totalidade apresentado na página, assim como o zoom dos detalhes apresentados na mesma página. b. Definir ambiente de simulação: executa os fluxos de automação em máquinas remotas, o que permite que o mesmo possa ser executado em diferentes configurações. Assim, os ambientes são as máquinas disponíveis onde os fluxos poderão ser executados. c. Definir agendamento <ol style="list-style-type: none"> i. Criar agendamento/ Descrever agendamento: possui um formulário, que permite inserir a descrição do Schedule com uma breve visão do que o mesmo pretende executar e no separador “Schedule”, se este deverá correr em <i>Ad-hoc</i> ou num tempo específico;

- ii. **Indicar fluxos a executar:** selecionar os fluxos que se pretendem ser incluídos no agendamento;
- iii. **Selecionar ambientes:** selecionar o ambiente ou ambientes onde o agendamento deverá executar os fluxos.
- iv. **Indicar variáveis de simulação:** é possível definir uma variável ou mais para um agendamento, com o respetivo nome e valor. Um fluxo pode ser executado e obter diferentes valores, dependendo em qual agendamento é que o valor é aplicado.
- v. **Indicar ações:** é possível adicionar uma ação de modo a que seja executada de acordo com as configurações indicadas.

2. Executar Simulação

- a. **Executar fluxo:** permite configurações que poderão ser feitas relativamente aos tempos de execução, aos valores por defeitos que os fluxos deverão assumir, entre outros;
- b. **Programar fluxo:** permite a programação da execução dos fluxos de automação, permitindo que sejam executadas outras tarefas ao mesmo tempo;
- c. **Obter documentação de execução:** permite a obtenção de documentação em vídeo dos fluxos gerados, após executar o fluxo;

3. Analisar os resultados da simulação

- a. **Criar um *dashboard*:** permite a criação de *dashboards*;
- b. **Executar *dashboard*:** os fluxos poderão ser executados e visualizados numa *Dashboard*;
- c. **Visualizar resultados:** acesso aos seus resultados através de gráficos ou documentação onde é possível constatar o estado final do fluxo.
- d. **Criar relatórios:**
 - i. **Descrever relatório:** permite documentar as execuções de teste de três maneiras, que são usadas em simultâneo. Elas são: uma gravação de vídeo de toda a execução, uma versão do design do fluxo e um registo de atividades com informações claras dos blocos efetuados. Com a reprodução da documentação é possível detetar e resolver problemas no fluxo
 - ii. **Definir agregação de informação:** é possível selecionar um ou mais campos de modo a agrupar os resultados. É possível adicionar vários campos, tendo a possibilidade de criar um relatório com o máximo detalhe.

- iii. **Definir filtro de informação:** especificar as condições para filtrar os resultados da execução

dos fluxos.

4. Guardar Simulação

- a. **Visualizar histórico de fluxos:** permite visualizar um histórico de versões criadas anteriormente, sendo possível restaurar e apagar uma versão;
- b. **Visualizar histórico:** é possível visualizar a lista de eventos, com a respetiva data e estado, que já ocorreram para este agendamento;
- c. **Gerir Base de Dados;**

5. Outras ferramentas: acesso a um centro de aprendizagem gratuito e artigos de ajuda, bem como a um Chat, com especialistas (Leapwork, n.d.-c).

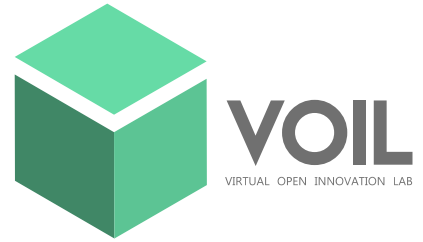
- a. **Configurar *Controller*:** conceder o acesso aos fluxos, gerindo os ambientes criados e a respetiva execução dos mesmos.
- b. **Gerir chaves de acesso à API:** para integrar componentes no simulador;
- c. **Configurar definições de Email:** é possível configurar o email, sincronizando-o com uma agenda, sendo então exequível encaminhar aos destinatários o pretendido
- d. **Configurar Proxy:** configurar o acesso a sites e a aplicações através de um proxy, mediante o contexto a que for aplicado.

Tabela 63 - Simulador Leapwork

Exemplo de Guia de Aprendizagem



Co-funded by the
Erasmus+ Programme
of the European Union



Intellectual Output 3: Learning Guide for IoT Technology

in the context of the Erasmus+ project *Virtual Open Innovation Lab – VOIL*

2020-06-15

Presented by:

- Institute of Entrepreneurship Development
- Wiesbaden Business School
- Universidad de Deusto
- University of Graz
- University of Agder
- Kaunas University of Technology
- University of Minho
- COTEC
- University of Münster

Content

Figures

Tables

1. Overview of the Guide
 - 1.1. Introduction to IoT technology
 - 1.2. IoT driven digital transformation: building smart environments
 - 1.3. Simulation-based learning of IoT technology
2. Start using CupCarbon simulator
 - 2.1. CupCarbon: a platform for designing and simulating IoT sensor networks
 - 2.2. Installing CupCarbon
 - 2.3. The key features of CupCarbon
3. Simulation-based learning: the case of pedestrian routes in the city
 - 3.1. Preparing for simulation
 - 3.1.1. Setting the learning objectives
 - 3.1.2. Resource to access before simulation
 - 3.1.3. The learning journey timeline
 - 3.2. Building the simulation experience
 - 3.2.1. Building the simulation environment
 - 3.2.2. Running the simulation and interpreting the results
 - 3.2.3. Changing simulation parameters to explore new possibilities
4. Debriefing
 - 4.1. Reflecting on the learning experience - Reaction/Description Phase
 - 4.2. Reflecting on changes in perspective and skills - Understanding/Analysis Phase
 - 4.3. Defining new learning objectives and corresponding simulation scenarios
- Application/Summary Phase
5. Self-assessment approach
 - 5.1. Quiz to self-assess the developed knowledge
 - 5.2. Scenarios to be implemented in CupCarbon
6. Additional resources
 - 6.1. Bibliography
 - 6.2. Web resources
 - 6.3. Other IoT Simulators
7. References

Figures

Figure 1. Example of a 2D part of a city as displayed in CupCarbon.

Figure 2. CupCarbon's menu bar.

Figure 3. The learning experience timeline.

Figure 4. The simulation environment.

Figure 5. Changing parameters to run a new simulation.

Figure 6. Routes display different temperatures.

Tables

Table 1. Overview of the Smart Environment Concepts.

Table 2. Steps to create the scenario in the CupCarbon.

1. Overview of the Guide

This guide was created to be used by learners interested in understanding the disruptive potential of Internet of Things (IoT) technology by simulating applications of this technology in simple and informative environments. The guide aims to be a useful resource for the development of the advanced digital skills necessary for sensing digitally enabled growth potentials.

This part of the guide provides an overview of the Internet of Things (1.1) as well as its potential to transform business processes and models (1.2). This section presents the principles of simulation-based learning that make it effective as an educational approach to the development of advanced digital skills (1.3) and is discussed the use of IoT in building smart environments (1.4).

In the remaining parts of this document, the IoT simulator CupCarbon is described along with information on how to install and use it (Chapter 2), the learning experience is explained and guided step-by-step (Chapter 3), and it is offered guidance for post-simulation reflection (Chapter 4) and self-assessment of learning (Chapter 5). In the last part of this document (Chapter 6) complementary resources are listed to deepen the learning started with this guide.

1.1. Introduction to IoT technology

The Internet of Things is a network of devices of all types (computers, smartphones, vehicles, home appliances, cameras, medical instruments, and many others) are connected to the Internet and are uniquely identifiable and globally accessible (Minerva et al., 2015; Patel & Patel, 2016).

This heterogeneity of the network allows a great level of combinations for all kinds of purposes and can be extensively applied in a large number of different domains, such as agriculture, social networks, buildings and cities, industry, transportation, healthcare, among others (Gubbi et al., 2013).

From a technical standpoint, IoT systems are usually divided into three separate layers, namely the *perception layer* that collects data from the environment, the *network layer* integrating network specifics such as communication protocols and heterogeneous

devices, and the *application layer* connecting the different functions the system provides and interfaces with the end-user (Silva et al., 2018).

The growth of the Internet of Things, coupled with the fact that multiple systems have different architectures, distinct devices, and form unique networks, increases the complexity of the system as a whole. The heterogeneity intrinsic to IoT systems in conjunction with the number of devices raises several issues, including the security and privacy concerns as well as interoperability problems due to the lack of standards (Li et al., 2015).

The challenges inherent in the adoption of IoT systems make their simulation a strategy of experimentation and learning with obvious benefits to avoid unnecessary costs, identify risks and develop the relevant skills for the operation of these complex systems.

1.2.IoT driven digital transformation: building smart environments

The numerous application possibilities of IoT make this technology particularly suitable for the digital transformation of organizations. This technology allows the interconnection of everyday objects through the internet, creating smart environments with the objective of facilitating the lives of individuals and making business more agile.

The smart environment concept integrates several application domains, including climate smart agriculture, smart commerce, smart energy, smart governance, smart health, smart city, smart manufacturing, smart mobility e smart workplace. The following table provides a brief overview of these concepts. In all cases the IoT is used to generate information to support decision and support a prompt, often automated, reaction to events.

Table 1. Overview of the Smart Environment Concepts.

climate smart agriculture	Set of technological strategies to increase, in a sustainable way, the resistance and adaptability of soil and agricultural products to climate change, also reducing or eliminating greenhouse effect emissions.
smart commerce	Technology is used to provide an enhanced and holistic customer experience by binding together cloud, social media, business analytics, and back-end enterprise systems.
smart	Technology is combined with open and agile governance practices to promote

governance	stakeholder participation and collaboration at all levels and in all branches of a country's or organization's governance process.
smart health	Application of advanced technology and systematic collection of patient data in order to allow personalized medical care, facilitate the diagnosis of silent diseases as well as anticipate and treat life-threatening health events such as strokes or heart attacks.
smart city	Urban centers in which a variety of IT applications are used to integrate and optimize municipal operations, reducing costs and improving the quality of life of their inhabitants.
smart manufacturing	Production based on fully integrated and collaborative systems that respond in real time to meet the changing demands of the plant, supply chain and customer needs.
smart mobility	Use of advanced technologies to improve the performance and attractiveness of the transport system in cities, aiming to reduce the need to travel and, therefore, reduce energy consumption and carbon emissions in the city.
smart workplace	The use of technology to connect and engage employees in their tasks, promoting employee collaboration and motivation while ensuring the flexibility of work environments and work relations.

The simulation scenario used to guide learning in Chapter 3 fits into the concept of intelligent mobility, with a focus on the application of IoT in a pedestrian recommendation system that allows interaction with the city in order to inform the user decision about the route to go.

1.3.Simulation-based learning of IoT technology

Simulators have been used in several fields to create digital environments for testing and experimenting with decisions and actions before being implemented in similar real environments. In the context of complex IoT systems, the need for simulation is even more evident since its development is time-consuming and expensive, and its implementation implies profound changes in business processes and models.

IoT simulators can help to prepare a project for installing smart device networks before their actual deployment, allowing to anticipate important aspects such as signal overload or the feasibility of the deployment in terms of location, interference, communication and cost.

There are some important terms to grasp before fully understanding IoT simulation. **Hardware-in-the-loop simulation** is a type of simulation used to test physical devices. This type of simulation integrates the physical device in a simulated network, allowing the sending of electrical signals between devices in the network in order to understand

the behavior of the device as well as its performance in the simulated network configuration.

Multi-level simulation allows for the simulation of different devices at different levels of detail. In this type of simulation, it is possible to create simulations at different levels of abstraction (ex: car, component, circuitry), focusing on the behavior of devices at each level or the interaction of devices at different levels of abstraction. A multi-level simulation tool provides the user with the option to adjust the trade-off between simulation speed and simulation correctness.

From the level of architecture, there are several architectures in common among IoT simulators, including

- **Multi-Agent simulation** that allows for the creation of environments where single, autonomous entities, called agents, act and interact with each other. This architecture is adequate when testing the behavior of autonomous agents in a decentralized system is relevant for the experimentation with decisions.
- **Discrete Event simulation** is particularly adequate when understanding the response of the system to specific (or a sequence of) events is the aim of the simulation. This type of simulation provides a dynamic platform within which the randomness inherent in complex systems can be emulated.

CupCarbon, the simulator used to implement the simulation scenario described in this guide, is a multi-agent and discrete event simulator.

The concept of wireless sensor network (WSN) is also central to the learning proposed in this guide. A WSN is an ad-hoc network composed of several sensor nodes capable of collecting, sending and receiving environmental information. Sensors are electronic and autonomous components that operate on wireless networks. A sensor consists of four components: (1) microcontroller, which is a programmable integrated circuit; (2) a radio antenna for wireless communication; (3) battery and (4) capture unit to intercept environmental information such as movement, temperature, humidity, gas, among other information

2. Start using CupCarbon simulator

2.1. CupCarbon: a platform for designing and simulating IoT sensor networks

CupCarbon is an open source IoT simulator for smart urban environments, with detailed topology support at street level based on real world maps. Thus, CupCarbon allows for the simulation of intelligent systems in the most diverse aspects of city management, from the mobility of people and vehicles to the management of disruptive events such as storms or fires (Bounceur, et al, 2018a; Mehdi et al., 2014).

The CupCarbon simulator is composed of 4 main parts: a 2D/3D environment block, an interference block, a radio channel block and an implementation block.

Radio Channel Block

CupCarbon integrates two models of radio propagation. The first is based on a point-by-zone acceleration structure called the visibility tree and which is useful to estimate the attenuation of the channel in an area and the impulse response of the channel (IR) according to a large number of receivers. The second integrates a complete 3D ray tracing associated with a Monte-Carlo algorithm, capable of exploring the diffuse behavior of the signal reflection surfaces.

Interference Block

Interference is a factor that limits the efficiency of networks with many devices attached. CupCarbon allows the generation of all received signals, as well as an accurate assessment of the resulting interference. This is possible because CupCarbon allows the implementation of a physical communication layer with the integration of three baseband models: the ZigBee (Noreen et al., 2016a), WiFi (Noreen et al., 2016b) and LoRa (Noreen et al., 2017) protocols. In this way, the evaluation of the quality of a connection takes into account the radio channel and data coding. In CupCarbon, it is possible to randomly generate interference at specific points in the network, as well as randomly generate global interference, either from factors that have little local impact or from factors distributed in a similar way throughout the network.

2D/3D Environment Block

CupCarbon allows viewing in 2D and 3D. The 2D environment is useful for creating realistic simulations and making improvements in the functioning of the IoT network; the 3D environment helps to make more precise adjustments when considering terrain elevation. (Bounceur, et al, 2018b). Considering the elevation of the terrain as well as detecting the presence of buildings is important because they are factors with an impact on the propagation of the signal and on the generation of interference that reduce the efficiency of the network. Fig. 1 shows an example of a city displayed in the 2D environment of the CupCarbon simulator and integrating atmospheric temperature sensors

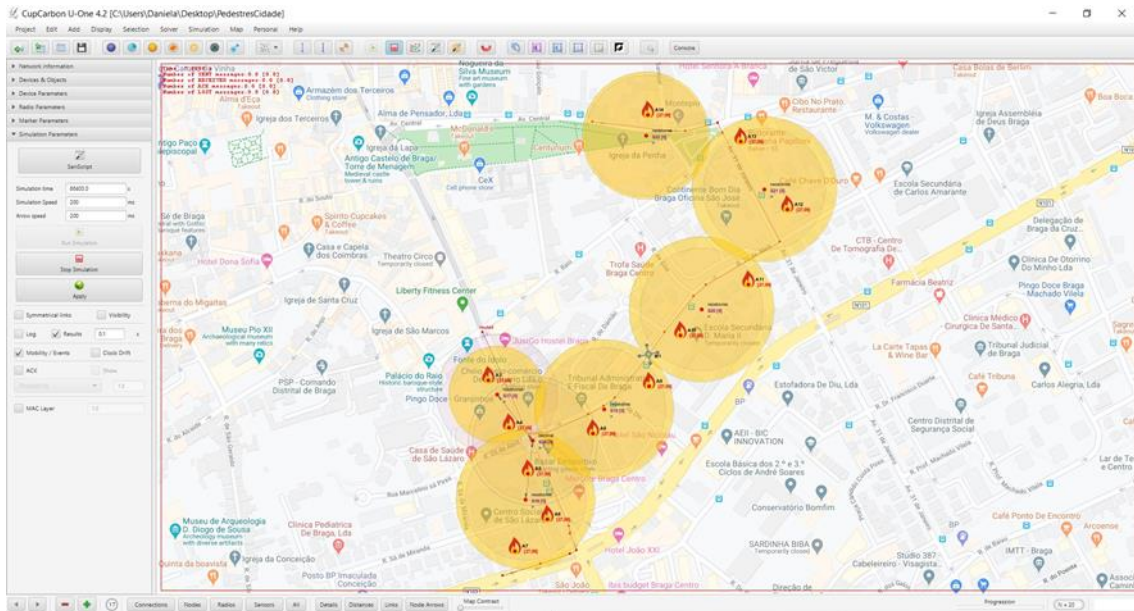


Figure 1. Example of a 2D part of a city as displayed in CupCarbon.

Implementation Block

CupCarbon was designed with a modular structure to simplify the replacement and customization of specific parts of the wireless network, as it is enough to include or remove modules for mobility, network, interference etc. This structure also allows the replication of modules in various parts of the simulated network.

2.2. Installing CupCarbon

To start using CupCarbon it is necessary to install an updated version of the Java software (<https://www.java.com/download/>). The simulator can be downloaded from the website <http://cupcarbon.com/>; the *capcarbon.jar* file in the extracted folder must be run.

CupCarbon comes with the OPEN Street Maps (OSM) and Google Maps API, allowing for running simulations on real locations and using a uniform background. In addition, through the Google Elevation API it is possible to get a view of the elevation of the ground.

2.3. The key features of CupCarbon

The use of simulators goes through three main phases: the creation of the simulation

environment, the execution of the simulation and the analysis of results. For each phase, each simulator presents a set of specific functionalities aligned with the objectives of the simulation. In the case of CupCarbon, the main features are the ones described below. To access detailed information about the simulator's features, please access the user manual listed in section 6.2 of this document.



Figure 2. CupCarbon's menu bar.

To create the simulation environment

- F1. Managing projects to create, open and save simulations (Menu > Project).
- F2. Changing the background map in the simulation environment. The simulator makes available a set of maps (open street maps - OSM - and Google Maps) so that connected objects can be integrated and operations carried out on geographically distributed networks (Menu > Map).
- F3. Adding sensors and mobile devices to the map. The simulator offers a wide variety of sensors for specific purposes, including temperature measurement, light level and traffic intensity. Some of the sensors that can be added to the map simulate events such as gas leaks or fires; others allow for simulating weather conditions. Sensors and devices objects can be interconnected in networks scattered throughout the city. (Menu > Add).
- F4. Parameterizing the network objects. The objects integrated in the simulation environment can be adjusted so that they can produce the desired simulation. The latitude, longitude and radius of devices can be defined/changed:

*Parameter Panel >
Network information;
Devices & Objects;
Device Parameters;
Radio Parameters;
Marker Parameters.*

- F5. Defining the communication between sensors can be programmed (Menu > Personal)

To run the simulation

- F6. Parameterizing the simulation by indicating the duration of the simulation and the speed of execution (Simulation Parameters Panel).

F7. Selecting predefined functions / algorithms to make the simulation more efficient (Menu > Solver) or developing new algorithms to be used in the simulation (Menu > Simulation > SenScript Window).

F8. Generating natural events such as storms or hurricanes (Menu > Simulation > Natural event generator)

F9. Running the simulation (Menu > Simulation > Run Simulation).

F10. Stopping the simulation. When the simulation is stopped, it is possible to know the situation at a given moment in the simulation by analyzing the values obtained up to that moment. (Menu > Simulation > Run Simulation).

To analyze the results

The analysis of the simulation results depends on the objectives defined for the simulation. In general, CupCarbon has the following analysis features.

F11. Analysis of mobility data: analyze data related to maps, distances and time. (Menu > Simulation > Test...mobilities).

F12. Viewing diagrams containing relevant indicators such as energy consumption, air pollution or traffic jams (Menu > Simulation > Energy Consumption)

3. Simulation-based learning: the case of pedestrian routes in the city

In the following sections, learning objectives are defined, the simulation scenario is described and detailed instructions are presented for implementing and testing it in order to develop the learning experience necessary to achieve the defined objectives.

3.1.Preparing for simulation

The learning promoted by this guide focuses on the competence area – **sensing digitally enabled growth potentials**. As indicated in the VOIL Curriculum, this area focuses on developing the skills needed to seek out and understand external information in order to identify new opportunities for growth. To this end, it is necessary to understand the main digital technologies and their applications, assess their business potentials and monitor technology trends. The technology that will be explored in this guide is IoT; the simulation scenario that will guide the construction of understandings about the technology and its application potential is the recommendation of pedestrian routes in the city environment.

Simulation scenario - recommendation of pedestrian routes

The recommendation of routes for public or private transport is offered by a large number of applications used by an increasing number of people. The recommendation of routes that implies interaction with the specific conditions of a city is less common but equally relevant for choosing a route between two points. When pedestrians intend to go to a certain destination, having enough time, they choose routes with different characteristics such as lighting, few intersections, coverage of the city's Wi-Fi network, attractiveness of the route (shops, parks, restaurants, monuments), route safety and pollution, for example.

With regard to a mobile route recommendation service, it must integrate three essential components, namely user positioning, route calculation, and route communication. In a realistic and versatile simulation scenario, it will be defined a set of sensors placed along predefined routes in order to establish a wireless sensor network that can spread the values associated with the monitored parameters. These parameters correspond to the pedestrian's preferences that are pre-defined.

The wireless sensor network will allow pedestrians to choose the route that best suits them for walking at night in the city. For this, the pedestrian must know the route that offers the best lighting (parameter 1), a pleasant average atmospheric temperature along the route (parameter 2) and an extension of the route in meters that s/he considers acceptable (parameter 3).

3.1.1. Setting the learning objectives

In line with the VOIL curriculum learning goals, the following learning objectives are defined:

- **UNDERSTAND** - Describe the role of IoT in solving mobility challenges.
- **EXTEND** - Solve simple mobility challenges with IoT.

Upon reaching these learning objectives, the learner will be able to design a basic Wireless Sensor Network (WSN) as well as test its behaviour. Part III of this guide offers suggestions for setting learning objectives at the exploring and transferring levels, thus strengthening their competences in understanding and monitoring IoT applications and their business potentials.

3.1.2. Resource to access before simulation

Simulation environments:

Ubiquitous sensor network simulation and emulation environments: A survey.

Sharif, M., & Sadeghi-Niaraki, A. (2017)

Journal of Network and Computer Applications, 93, 150-181

Link: <https://doi.org/10.1016/j.jnca.2017.05.009>

IoT for the digital transformation of business models

Madakam, S., Lake, V., Lake, V., & Lake, V. (2015). Internet of Things (IoT): A literature review. Journal of Computer and Communications, 3(05), 164.

<http://dx.doi.org/10.4236/jcc.2015.35021>

Pflaum, A. A., & Gölzer, P. (2018). The IoT and digital transformation: toward the data-driven enterprise. *IEEE pervasive computing*, 17(1), 87-91.

<https://doi.org/10.1109/MPRV.2018.011591066>

Smart Mobility:

Smart mobility: A survey.

Faria, R., Brito, L., Baras, K., & Silva, J. (2017, July).

2017 International Conference on Internet of Things for the Global Community (IoTGC) (pp. 1-8).

Link: <https://doi.org/10.1109/IoTGC.2017.8008972>

IoT and CupCarbon simulations:

Routing in IoT Network using CupCarbon Simulator.

Johari, R., & Adhikari, S. (2020, February)

2020 7th International Conference on Signal Processing and Integrated Networks (SPIN) (pp. 301-306).

Link: <https://doi.org/10.1109/SPIN48934.2020.9071227>

CupCarbon User Guide.

Link: https://cupcarbon.com/cupcarbon_ug.html

3.1.3. The learning journey timeline

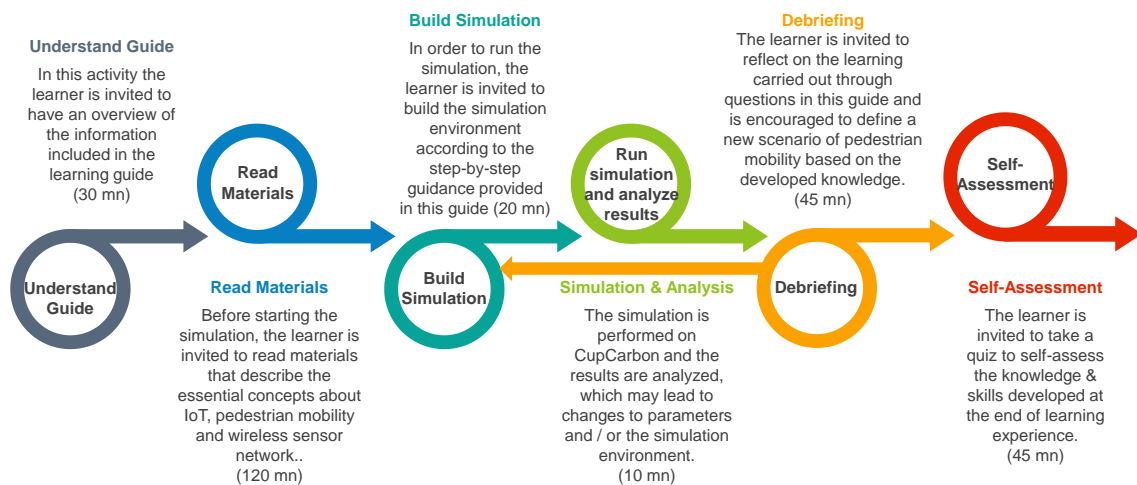


Figure 3. The learning experience timeline.

Although Figure 3 shows an almost linear learning journey that has a total duration of approximately 4 hours and 30mn. The learner may want to return to previous phases to reread materials, change the simulation environment, review parameters of the wireless sensor network, among other changes. These reviews of the learning journey already taken are encouraged to deepen the learning experience; they imply a longer journey than indicated here.

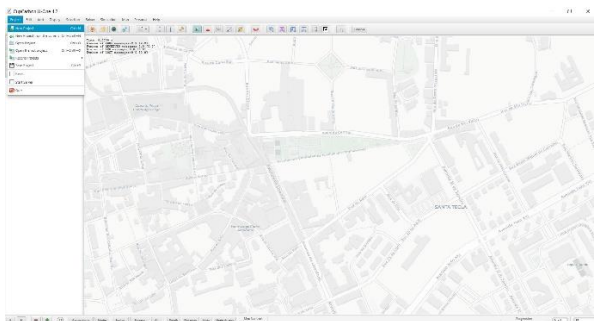
3.2. Building the simulation experience

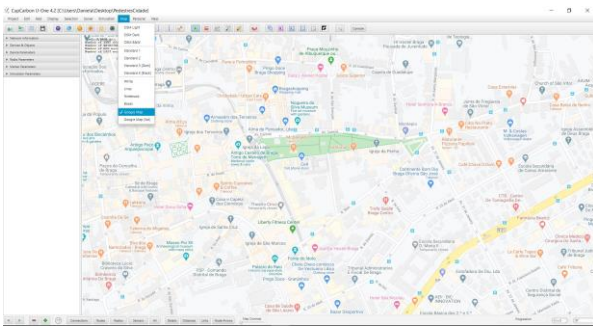
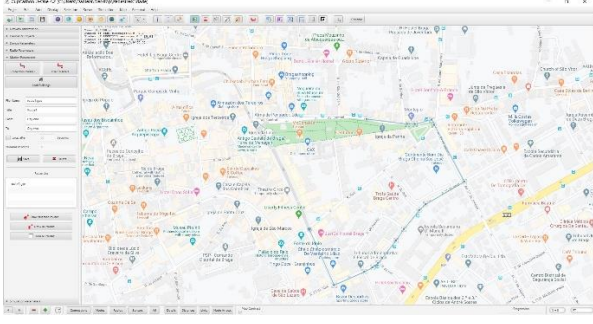
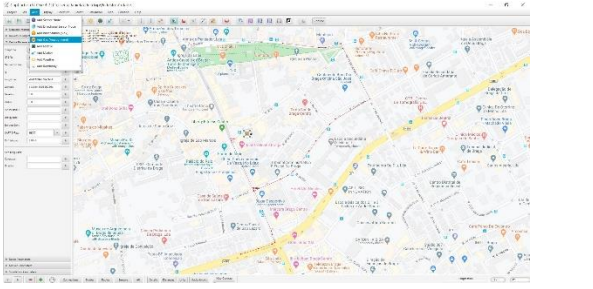
This section provides step-by-step guidance for implementing the simulation scenario. The proposed scenario is to simulate a system that suggests routes that best suits pedestrians for walking at night in a city. Three parameters were suggested: (1) street lighting, (2) atmospheric temperature along the route, and (3) extension of the route.

3.2.1. Building the simulation environment

To create the scenario in the CupCarbon simulator it will be necessary to create the following basic elements: routes, mobile sensor corresponding to the pedestrian, temperature sensor, random temperature event, receiver sensors and a node sensor to determine the best route. The steps that need to be taken are shown below, accompanied by the images that illustrate their execution.

Table 2. Steps to create the scenario in the CupCarbon.

Action	CupCarbon instructions	Image
Create Project	Project → New Project “PedestrianRoutes” → Save	

<p>Select a map</p>	<p>Map → GoogleMap</p>	
<p>Create two routes; Select two points on the map (departure and arrival); save route points</p>	<p>Add → Add Marker Marker Parameters → Route From Markers → Save</p>	
<p>Add a mobile sensor that will represent the pedestrian walking the routes; associate a script to this sensor; save the script and associate it with the mobile sensor</p>	<p>Add→Add Mobile → “pedestrian” Simulation → SenScriptWindow device parameters: Script File → select the script labeled "pedestrian"</p>	<p>SCRIPT “pedestrian”</p> <pre> loop wait read d rdata \$d x n route if(\$x==B) print \$n : \$route set last B route \$route end </pre>
<p>Insert gas sensors to read the temperature at various points in the created routes; Create a random event to generate temperatures along the routes created; associate this event to the various sensors inserted on the</p>	<p>Add → Add Gas Simulation → Natural Event Generator → Mean: 24, Std:4, Period:1 → Generate → Save on each sensor → open the device parameters tab → Natural Evt file → select the event just generated labeled “temperature”.</p>	

<p>routes</p>		
<p>Place receiver sensors to cover all gas sensors;</p> <p>These sensors will receive information about temperature and lighting so you need to associate a script to each of them;</p> <p>Associate the script to the sensors.</p>	<p>Add → Add Sensor Node – “receiver”</p> <p>Simulation--->SenScript Window</p> <p>select the sensor and open the device parameters tab: Script File → select the script labeled "receiver"</p>	<p>SCRIPT “receiver”</p> <pre> loop wait read s1 rdata \$s1 temp tempMin tempMax per_light id areadsensor v if(\$v!=X) rdata \$v a b temp2 int temp2 \$temp2 if(\$temp2 < \$tempMin) set tempMin \$temp2 end if(\$temp2 > \$tempMax) set tempMax \$temp2 end plus temp2 \$temp2 \$temp div temp2 \$temp2 2 randb light 0 1 if(\$light2==1) plus per_light \$per_light 1 end plus id \$id 1 print \$temp2 \$per_light data p \$temp2 \$tempMin \$tempMax \$per_light \$id send \$p \$id end delay 1000 </pre>
<p>Place a sensor at the fork in routes;</p> <p>The sensor will decide which is the best route taking into account the pedestrian's preferences, so it is necessary to associate a script;</p> <p>Associate the sensor with the script.</p>	<p>Add--->Add Sensor Node</p> <p>Simulation-->SenseScript Window</p> <p>select the sensor and open the device parameters tab: Script File ---> select the script that was labeled “central”</p>	<p>SCRIPT “central”</p> <pre> loop set pef B wait read v rdata \$v temp1 tempMin1 tempMax1 per_light1 wait read s1 rdata \$s1 temp2 tempMin2 tempMax2 per_light2 if(\$pef == B) set total1 25 set total2 25 set numSenR1 5 set numSenR2 2 set distance1 1514 set distance2 1389 div per_light1 \$per_light1 \$numSenR1 mult per_light1 \$per_light1 100 div per_light2 \$per_light2 \$numSenR2 mult per_light2 \$per_light2 100 if(\$temp1 > 20) plus total1 \$total1 35 end if(\$distance1 < 600) plus total1 \$total1 20 </pre>

		<pre> end if(\$tempMax1 <25) plus total1 \$total1 10 end if(\$temp2 > 20) plus total2 \$total2 35 end if(\$distance2 < 600) plus total2 \$total2 20 end if(\$tempMax2 <25) plus total2 \$total2 10 end print \$total1 \$total2 if(\$total1 > \$total2) data d B M2 route1 send \$d 225 else data d B M2 route2 send \$d 225 end </pre>
--	--	---

The final environment will look similar to the figure below. Please note that your environment will be adjusted to the choices you made regarding the map, routes and placement of sensors.

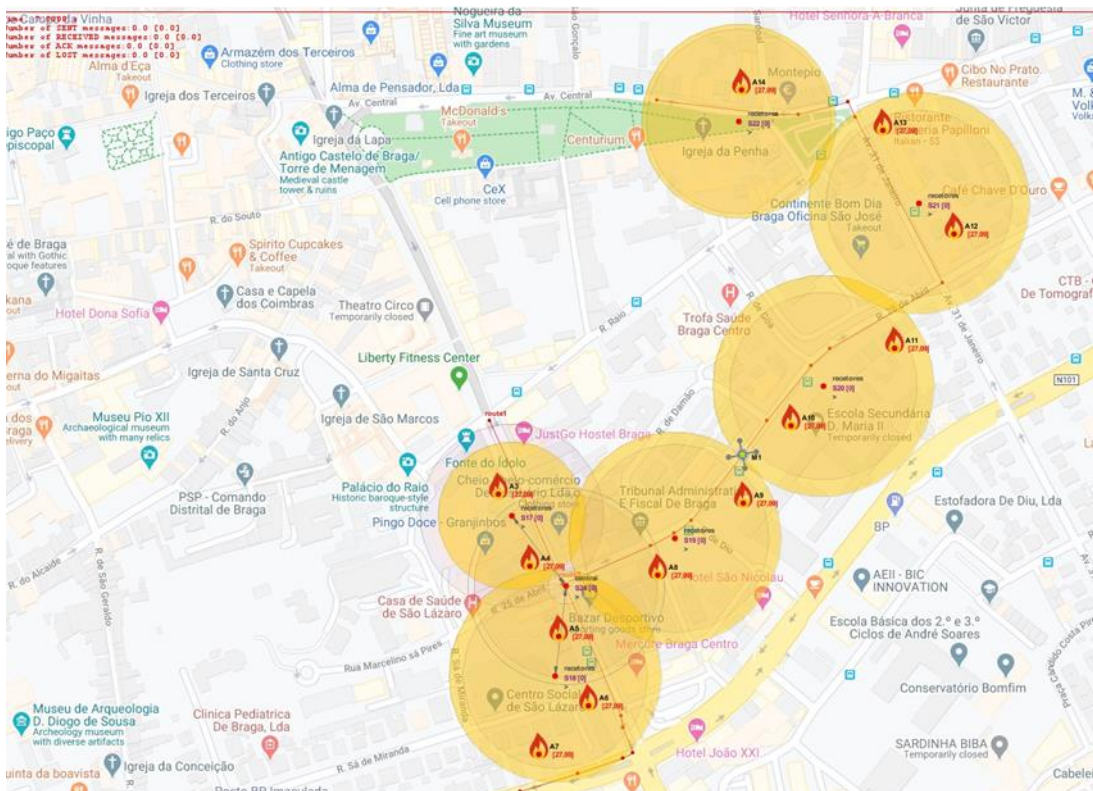


Figure 4. The simulation environment.

3.2.2. Running the simulation and interpreting the results

To perform the simulation of a pedestrian route recommendation system, the following commands must be executed:

Simulation Parameters → Simulation Speed: 200, Arrow Speed: 200 → Run Simulation

When running the simulation, it is possible to observe that the mobile device starts its route on the route and when it reaches the fork, it chooses the route that presents the best temperature and lighting values as it is the one that best suits you (script).

It is also possible to observe that each gas sensor is showing the temperatures (within the interval defined in the respective script), which were randomly generated.

3.2.3. Changing simulation parameters to explore new possibilities

Taking into account the implemented scenario, we can, for example, change the temperature function, so that the temperature affects different intervals; the change may be introduced only in the sensors of route2.

The Natural Event Generator Mean: 24, Std: 4, Period: 1 will be replaced by different parameters such as Mean: 13, Std: 4, Period: 1 associated only to the sensors in route2.

A change similar to the one in Figure 5 below will happen

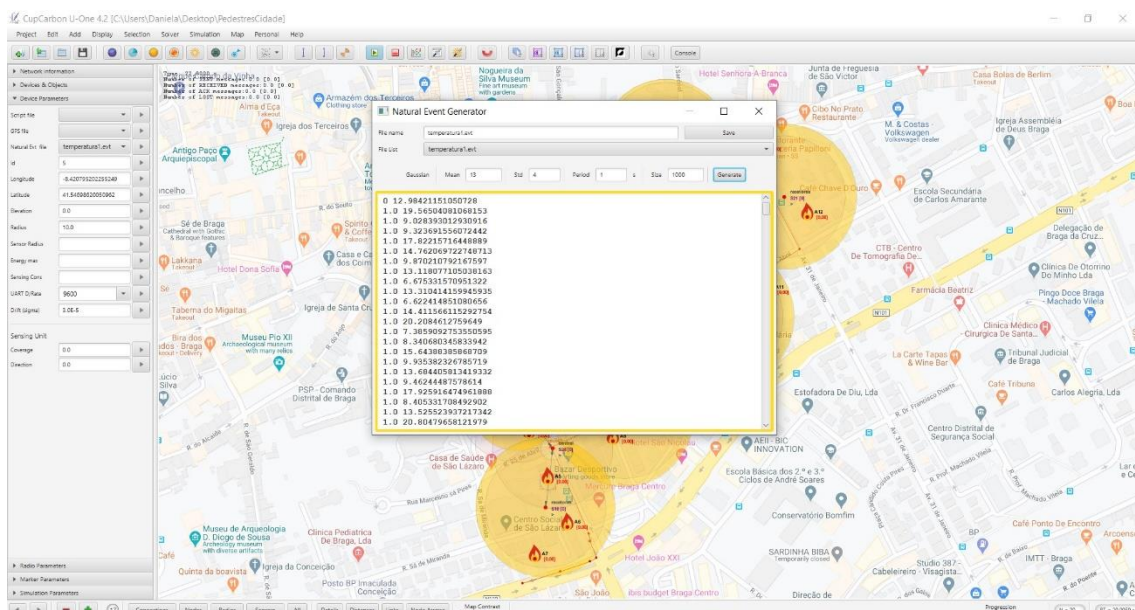


Figure 5. Changing parameters to run a new simulation.

The next figure shows that the sensors are showing different temperatures between route1 and route2 due to the change of parameters.

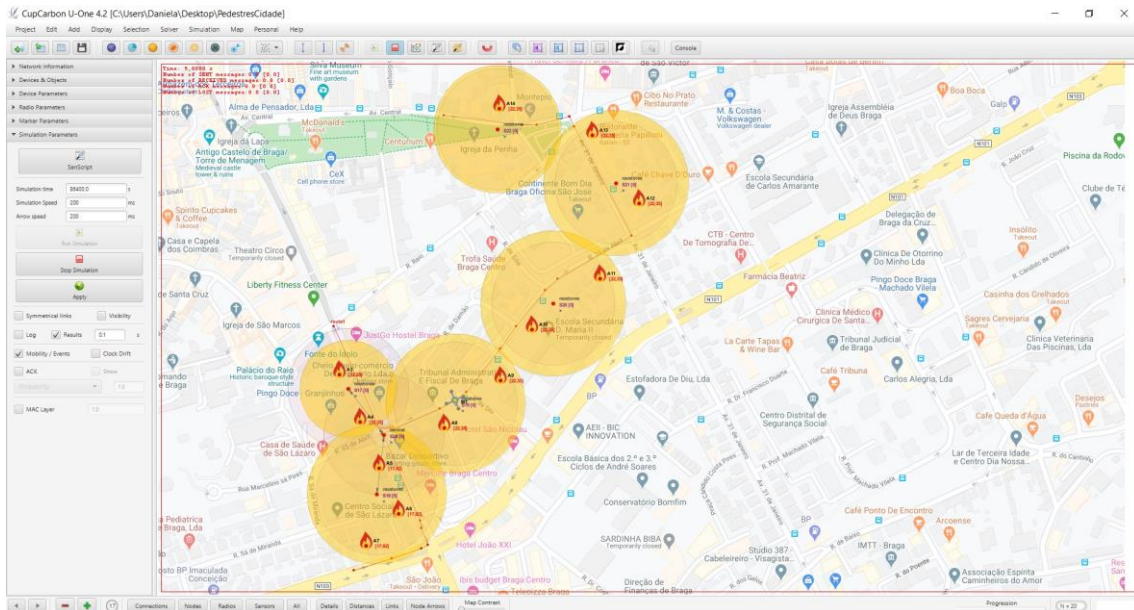


Figure 6. Routes display different temperatures.

4. Debriefing

Analyzing simulation experiences is increasingly understood as a crucial step to clarify and consolidate understandings as well as learning related to the use of simulators to support decisions. This is the reason for including in the simulation-based learning experience a phase called debriefing, which aims to stimulate reflective thinking in order to integrate the developed understandings, skills and attitudes with pre-existing knowledge (Decker et al., 2013; Shinnick et al., 2011).

Debriefing is the process by which users develop their reasoning through reflection and metacognition, linking theory to practice, allowing them to think critically and intervene professionally in complex situations (Mariani et al, 2013).

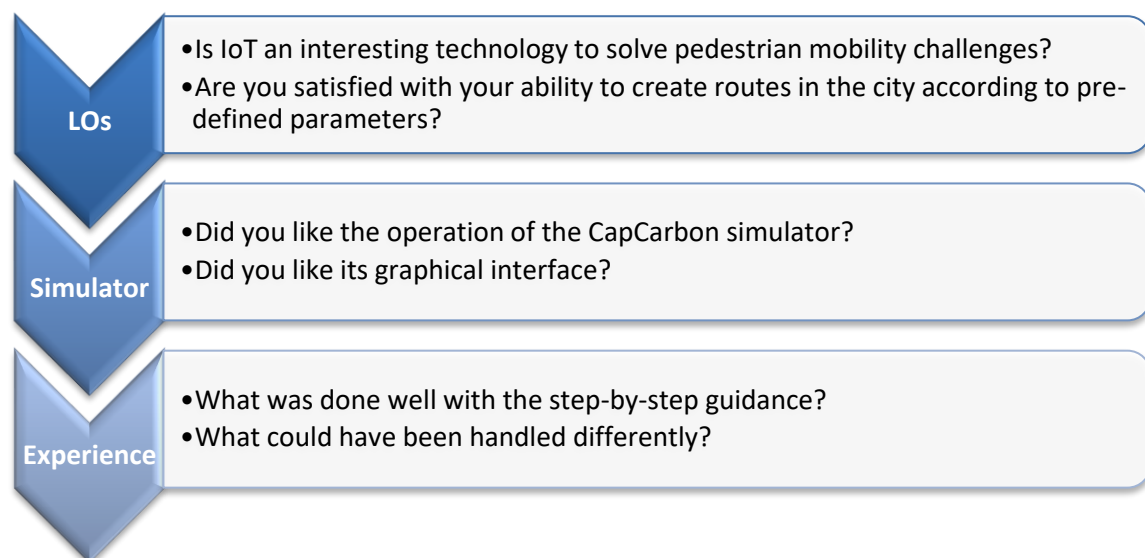
Debriefing focuses on reflection after each simulation, so it may happen more than once during the learning experience necessary to achieve the objectives defined for it (Abulebda et. al, 2019). This process comprises three main phases: reaction/description, analysis/understanding, and application/summary. In general, the phases integrate the following main activities:

- *PHASE 1 – Reaction/Description.* This phase aims to decompress and includes open-ended questions about the satisfaction felt with the simulation experience.
- *PHASE 2 – Understanding/Analysis.* At this stage, the learning objectives are revisited in order to understand how the simulation experience contributed to achieving them. The concepts and practices to be learned are explored in close connection with the simulation steps. The questions that guide this phase are associated with the understanding of what happened during the simulation and the reasons for the problems or decisions made.
- *PHASE3 – Application/Summary.* In this phase, the learning experience is applied to define the next simulation to be carried out to extend understandings and skills, thus consolidating the learning journey needed to fully achieve the defined learning objectives. At this stage learners are helped to formulate the questions they wish to see answered by the next simulation iteration.

Debriefing is a collective process, facilitated by the trainer. This learning guide assumes a self-directed learning journey. Thus, in the next sections specific suggestions are presented to facilitate the autonomous debriefing of the user of this guide.

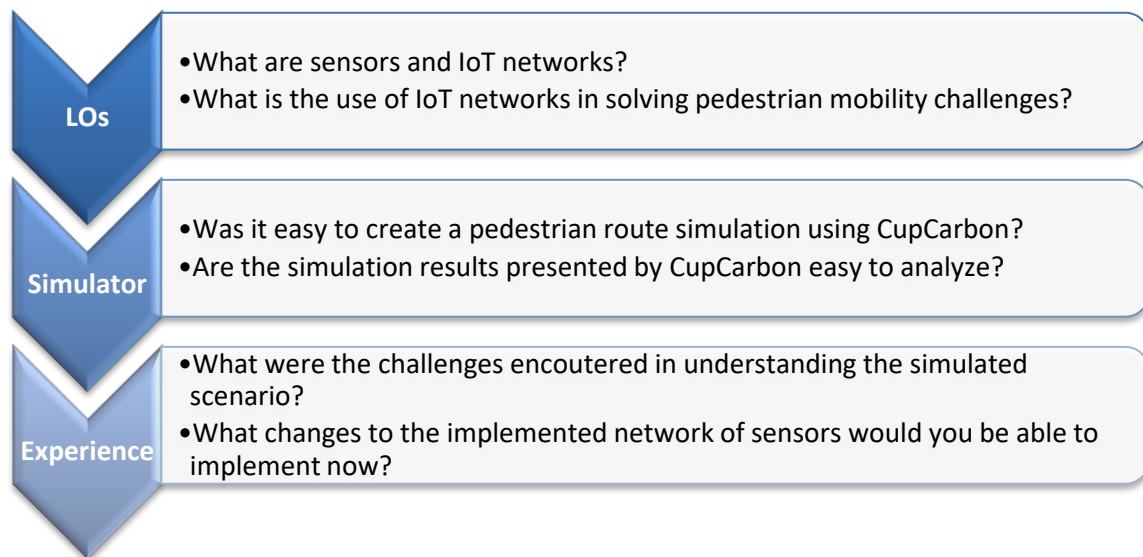
Reflecting on the learning experience - Reaction/Description Phase

At this stage the reflection addresses the overall simulation experience. We recommend the following reflection points:



4.1. Reflecting on changes in perspective and skills - Understanding/Analysis Phase

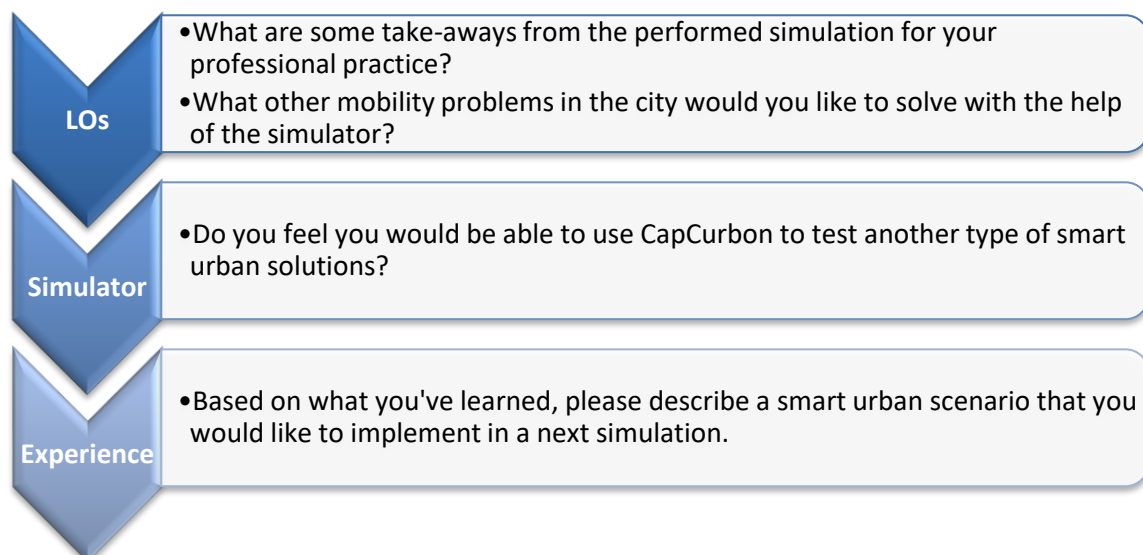
At this stage the reflection addresses the developed understandings and skills. We recommend the following reflection points:



4.2. Defining new learning objectives and corresponding simulation scenarios

Application/Summary Phase

At this stage the reflection addresses the ability define new problems and simulation scenarios. We recommend the following reflection points:



5. Self-assessment approach

Simulation exercises are used to develop the learner's knowledge, skills and reasoning and critical thinking skills. The evaluation of the skills developed focuses on their ability to define and test solutions and decisions with relevance in a given real-life domain.

In this guide, the self-assessment proposals were defined to determine (1) the learner's understanding about sensor networks and their use to solve mobility challenges, and (2) the learner's ability to define mobility decisions/solutions that can be tested using CupCarbon. It is proposed that the answer to the quiz in section 5.1 and the implementation of the scenarios described in Section 5.2.

Quiz to self-assess the developed knowledge

The role of IOT in transforming organizations and business

1. Organizations digitally transform their business processes and models to:
 - a. Integrate technological advances in the industry to which they belong
 - b. To remain competitive in turbulent environments
 - c. Keep employees motivated
 - d. Not sure

2. To sense digitally enabled growth potentials organizations
 - a. implement mechanisms and processes for scanning, observing and understanding changes in the business environment
 - b. develop an understanding of how digital solutions will help to deliver the firm's objectives and review the strategy accordingly
 - c. deploy initiatives to innovate value creation, value capture, and/or value offer
 - d. not sure

3. The Internet of Things (IoT) is:
 - a. All about connecting disparate devices together in some manner using a single app to control every aspect of a home, business, or industrial setting

- b. An open and comprehensive network of intelligent objects that have the capacity to auto-organize, share information, data and resources, reacting and acting in face of situations and changes in the environment
 - c. A decentralized, distributed ledger that records the provenance of a digital asset
 - d. Not sure
4. A data-driven organization employs IoT solutions to:
- a. To generate data insights for decision makers
 - b. Collect data from the operational processes
 - c. Create value from data and selling data embedded into smart services
 - d. Not sure

Answers: 1b, 2a, 3b, 4c

Wireless sensor networks for mobility management

1. A WSN is:
- a. A self-configured and infrastructure-less wireless network to monitor physical or environmental conditions and to cooperatively pass the data through the network to a main location or sink where the data can be analyzed
 - b. A set of sensors that communicate among themselves to produce a dataset to be analyzed
 - c. A network of smart objects that communicate among themselves to produce a dataset to be analyzed
 - d. Not sure
2. Which of the following are components of a WSN?
- a. Sensor nodes
 - b. Sensors
 - c. Gateways
 - d. All the above
3. Smart mobility concept envisages the use of WSN to improve:
- a. Pedestrian routes in a city
 - b. The management of product transportation to customers

- c. Traffic management in cities taking into account a wide range of modes of mobility, including streetcars, bicycles, buses, light rail trains, subways, taxis, autonomous vehicles, walking.
- d. Not sure

Answers: 1a, 2d, 3c

Simulation with CupCarbon

1. CupCarbon's objective is:
 - a. To design, visualize, debug and validate distributed algorithms for monitoring, environmental data collection and to create environmental scenarios such as fires, gas, mobiles for educational and scientific purposes. educational and scientific projects
 - b. To perform experiments in order to understand basic trade-offs and relations in process flow analysis with consideration of financial and operational key performance indicators
 - c. To develop and test traffic management software
 - d. Not sure
2. From the options below, what simulation could not be implemented in CapCarbon?
 - a. Adaptive waste collection for smart cities
 - b. Intelligent parking management in urban environments
 - c. Intelligent warehouse management
 - d. All the above
3. What are the mobiles that can be simulated with CpCarbon:
 - a. Humans
 - b. Vehicles
 - c. Flying objects
 - d. All the above

Answers: 1a, 2c, 3d

5.1.Scenarios to be implemented in CupCarbon

The resources in urban areas are increasingly being managed in a smart way in order to provide a better quality of life to citizens while ensuring the sustainability of the city. Pedestrian traffic presents specific challenges to the smart management of urban resources, due to the need to guarantee the safety and quality of the experience for

pedestrians. In this scenario, the learner is asked to simulate a recommendation system that allows the user to know which route is the most pleasant in terms of temperature, good Wi-Fi coverage and tourist points of interest.

6. Additional resources

Bibliography

Sotiriadis, S., Bessis, N., Asimakopoulou, E., & Mustafee, N. (2014, May). Towards simulating the internet of things. In 2014 28th International Conference on Advanced Information Networking and Applications Workshops (pp. 444-448). IEEE.

Sharif, M., & Sadeghi-Niaraki, A. (2017). Ubiquitous sensor network simulation and emulation environments: A survey. *Journal of Network and Computer Applications*, 93, 150-181.

Al-Ahmadi, H. H. (2020). Real-Time Simulation of Traffic Monitoring System in Smart City. *Global Journal of Computer Science and Technology*.

Bounceur, A., Marc, O., Lounis, M., Soler, J., Clavier, L., Combeau, P., ... & Manzoni, P. (2018, January). Cupcarbon-lab: An iot emulator. In 2018 15th IEEE Annual Consumer Communications & Networking Conference (CCNC) (pp. 1-2). IEEE.

Bounceur, A. (2016, March). CupCarbon: a new platform for designing and simulating smart-city and IoT wireless sensor networks (SCI-WSN). In *Proceedings of the International Conference on Internet of things and Cloud Computing* (pp. 1-1).

Cheng A, Grant V, Dieckmann P, Arora S, Robinson T, Eppich W, Faculty Development for Simulation Programs: Five Issues for the Future of Debriefing Training. *Simulation in healthcare: journal of the Society for Simulation in Healthcare*. 2015 Aug

Chernyshev, M., Baig, Z., Bello, O., & Zeadally, S. (2017). Internet of things (iot): Research, simulators, and testbeds. *IEEE Internet of Things Journal*, 5(3), 1637-1647.

Johari, R., & Adhikari, S. (2020, February). Routing in IoT Network using CupCarbon Simulator. In 2020 7th International Conference on Signal Processing and Integrated Networks (SPIN) (pp. 301-306). IEEE.

Lopez-Pavon, C., Sendra, S., & Valenzuela-Valdés, J. F. (2018). Evaluation of CupCarbon Network Simulator for Wireless Sensor Networks. *Netw. Protoc. Algorithms*, 10(2), 1-27.

Patel, N. D., Mehtre, B. M., & Wankar, R. (2019, December). Simulators, Emulators, and Test-beds for Internet of Things: A Comparison. In 2019 Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC) (pp. 139-145). IEEE.

Web resources

Blogs

Open Source for U. Simulating Smart Cities with CupCarbon .
<https://www.opensourceforu.com/2019/09/simulating-smart-cities-with-cupcarbon/>

Medium (Portuguese) Simular uma rede de sensores (Mesh) com protocolo Lora no software CupCarbon (Envio de Temperaturas). <https://medium.com/@daeynasvistas/a-iot-internet-das-coisas-surgiu-como-a-nova-gera%C3%A7%C3%A3o-da-internet-que-permitiu-a-telem%C3%B3veis-e-a-3d25bdef111b>

PPLWARE (Portuguese). CupCarbon: Simulador para smartcities e rede de sensores.
<https://pplware.sapo.pt/tutoriais/networking/cupcarbon/>

Books

Software Architecture: 11th European Conference, ECSA 2017, Canterbury, UK, September 11-15, 2017, Proceedings.
https://books.google.pt/books?id=X3syDwAAQBAJ&dq=cupcarbon&source=gbs_navlinks_s

Mobile, Secure, and Programmable Networking: 4th International Conference, MSPN 2018, Paris, France, June 18-20, 2018, Revised Selected Papers.
https://books.google.pt/books?id=qpaDDwAAQBAJ&dq=cupcarbon&source=gbs_navlinks_s

Video tutorials

CupCarbon Tutorial 1: Hello World. <https://www.youtube.com/watch?v=Lw9-Nk3Uzco>

DEMO: From CupCarbon Simulation to real WSN (ANR PERSEPTEUR).
<https://www.youtube.com/watch?v=KTrh4s19NLQ>

CupCarbon: From WSN simulation to Arduino/XBee real Network.
<https://www.youtube.com/watch?v=mifOP2KVTsY>

Camera Mobile Detection with CupCarbon.
<https://www.youtube.com/watch?v=He3zRXPKIgM>

CupCarbon: LEACH. <https://www.youtube.com/watch?v=epB3N0pFvY0>

WSN & Smart-City & Visibility: CupCarbon (ANR PERSEPTEUR).
<https://www.youtube.com/watch?v=pyC9Sknrpng>

CupCarbon Linear Routing (ANR PERSEPTEUR).
<https://www.youtube.com/watch?v=K0F6PIS-Go0>

Other IoT Simulators

The Network Simulator NS-2, [online] Available from: <http://www.isi.edu/nsnam/n>

OMNET++ discrete event simulator, [online] Available from: <http://www.omnetpp.org>

WSNet, [online] Available from: <http://wsnet.gforge.inria.fr/overview.html>

TOSSIM. Available from: <http://networksimulationtools.com/tossim/>

DPWSim. Available from: <https://sourceforge.net/projects/dpwsim/>

iFogSim. Available from: <https://opensourceforu.com/2018/12/ifogsim-an-open-source-simulator-for-edge-computing-fog-computing-and-iot/>

Cooja. Available from: <https://sourceforge.net/projects/contiki/files/Instant%20Contiki/>