



Estudo e Melhoria de Processos de
Desenvolvimento de Software Utilizando a
Tecnologia Neo4j

Miguel Ângelo Ferreira Sousa da Silva

Universidade do Minho
Escola de Engenharia





Universidade do Minho
Escola de Engenharia

Miguel Ângelo Ferreira Sousa da Silva

**Estudo e Melhoria de Processos de
Desenvolvimento de Software Utilizando a
Tecnologia Neo4j**

Dissertação de Mestrado
Mestrado em Engenharia de Sistemas

Trabalho efetuado sob a orientação do
Professor Doutor Ricardo Machado
Professor Doutor Lino Costa

DIREITOS DE AUTOR

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 abaixo 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.

Licença concedida aos utilizadores deste trabalho



Atribuição-NãoComercial-SemDerivações
CC BY-NC-ND

AGRADECIMENTOS

Chega assim, ao fim, mais uma etapa da minha vida, onde concluo mais um capítulo do meu percurso acadêmico, percurso esse onde da minha parte houve sempre uma grande ambição e dedicação em realizar o melhor trabalho possível e adquirir novos conhecimentos. No entanto, ao longo destes últimos meses, tive o privilégio de conhecer e privar com quem, mesmo que por vezes, através de pequenos gestos contribuíram para melhorar o meu desempenho acadêmico sempre no sentido de alcançar os meus objetivos a que inicialmente me propus.

Em primeiro lugar, e não podendo ser de outra forma, queria agradecer à minha mãe, avós, familiares e amigos por todo o apoio que me deram ao longo desta jornada, e por terem sempre acreditado em mim, onde me motivavam todos os dias, a querer ser melhor tanto como pessoa quer como aluno.

Ao Professor Doutor Ricardo Machado pela disponibilidade apresentada, que me ajudou a construir este documento com uma maior competência e um maior rigor.

Ao Professor Doutor Lino Costa por todos os ensinamentos prestados, ajudando-me com conhecimento para a orientação da minha dissertação.

Ao Tiago Pereira colega de departamento no Centro de Computação Gráfica, pela disponibilidade e ajuda sempre que precisei, principalmente na troca de conhecimentos.

A toda a equipa do CCG com quem trabalhei, agradeço pela boa disposição que cada um trazia para as reuniões e idas para o escritório, tornando assim todo o trabalho mais fácil, tendo também de agradecer pelo espírito competitivo de querer sempre mais.

Por último, mas não menos importantes e esperando não me esquecer de ninguém, agradeço aos meus colegas que passaram estes últimos meses comigo, que apesar de não poder citar-vos um a um, queria agradecer-vos por todo o apoio que me prestaram tanto em situações académicas, como em situações mais pessoais, sendo que mesmo que conseguisse acabar o meu mestrado sem vos ter ao meu lado, não era tão feliz como sou hoje.

DECLARAÇÃO DE INTEGRIDADE

Declaro ter atuado com integridade na elaboração do presente trabalho académico 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

Estudo e Melhoria de Processos de Desenvolvimento de Software Utilizando a Tecnologia Neo4j

Ao longo dos tempos, as empresas têm vindo a ficar dependentes e de necessitar de sistemas de informação, onde consigam obter respostas em tempo real, deste modo conseguem obter respostas que apoiem nas tomadas de decisão. Nos dias de hoje, a qualidade e a velocidade da informação são dos maiores proveitos que se pode ter num mercado de trabalho que se considera tão competitivo, possibilitando assim, que haja uma melhor gestão das *tarefas* a serem desenvolvidas e da forma como são executadas.

Contudo, estes sistemas têm na sua posse práticas relevantes para toda a organização, práticas que podem ser esquecidas ou perdidas no ecossistema da empresa, isto acontece se os utilizadores não tiverem conhecimento da sua existência, o que leva por vezes ao desenvolvimento de software com objetivos e procedimentos semelhantes.

No âmbito da colaboração entre Iberiana Technial (ITEC) e o Centro de Computação Gráfica (CCG), surge então o projeto “Projeto de Maturidade de Software”, que tem como objetivo principal, melhorar os seus procedimentos diários no que toca aos seus processos de desenvolvimento.

As ontologias, nos últimos anos tem vindo a ser utilizadas para agregar e relacionar conceitos de um dado domínio, com capacidade de padronizar informações relevantes, possibilitando assim, a interoperabilidade semântica da informação contida nesse mesmo domínio, de forma a ser utilizada pelos sistemas de informação.

Deste modo, esta dissertação propõe-se a criar boas práticas a ser utilizadas diariamente pela empresa, onde se utiliza a área das ontologias para que se consiga relacionar dois *frameworks* e retirar o melhor proveito da informação contida em cada um deles.

Palavras chave: Ontologias, CMMI, OpenUp, Guia de boas práticas

ABSTRACT

Study and Improvement of Software Development Processes Using the Technology Neo4j

Over time, companies have become dependent on and need information systems, where they can get answers in real time, so they can get answers that support decision making. Nowadays, the quality and speed of information are one of the greatest benefits that we can have in a labor market that is considered so competitive, where it allows a better management of the tasks to be developed and the way they are executed.

However, these systems hold relevant practices for the whole organization, practices that may be forgotten or lost in the company's ecosystem, this happens if users are not aware of their existence, which sometimes leads to the development of software with similar objectives and procedures.

Within the scope of the collaboration between Iberiana Technial (ITEC) and the Computer Graphics Centre (CCG), the project "Software Maturity Project" arises, whose main objective is to improve their daily procedures about their development processes.

The ontologies, in recent years, have been used to aggregate and relate concepts of a given domain, with the ability to standardize relevant information, which enables the semantic interoperability of the information contained in this same domain, to be used by information systems.

In this way, in this dissertation it is proposed to create good practices to be used daily by the company, where the ontologies area is used so that two frameworks can be related and take the best advantage of the information contained in each one of them.

Keywords: Ontologies, CMMI, OpenUp, Guide to good practices

ÍNDICE

Direitos de autor	iv
Agradecimentos.....	v
Declaração de integridade	vi
Resumo.....	vii
Abstract.....	viii
Lista de Abreviaturas/Siglas	xiii
Lista de Figuras	xvi
Lista de Tabelas	xviii
1. Introdução	1
1.1 Enquadramento.....	1
1.2 Contextualização.....	2
1.3 Objetivos	7
1.4 Abordagem Metodológica.....	8
1.5 Estrutura do Documento	9
2. Análise do Estado da Arte	9
2.1 Introdução.....	9
2.2 Desenvolvimento de Software.....	10
2.3 Framework CMMI	13
2.4 Framework OpenUp.....	18
2.5 Base de dados por Grafos.....	25
2.6 Conclusão	29
3. Caracterização do Problema Processual da ITEC	31
3.1 Introdução.....	31

3.2	Modelação com BPMN	31
3.3	Modelação de Casos de Uso	35
3.4	Requisitos dos Processos de Desenvolvimento	48
3.5	Conclusão	51
4.	Abordagem para a Melhoria dos Processos de Desenvolvimento com Neo4j	53
4.1	Introdução	53
4.2	Análise do CMMI e do OpenUp com Neo4j	53
4.3	Análise do Relacionamento entre o CMMI e o OpenUp	60
4.4	Proposta para a Melhoria dos Processos de Desenvolvimento.....	66
4.5	Conclusão	91
5.	Conclusões e Trabalhos Futuros.....	92
	Bibliografia	94
	Apêndice I	97
	Apêndice II	104
	Apêndice III	105
	Apêndice IV	106
	Apêndice V	107
	Apêndice VI	108
	Apêndice VII	109
	Apêndice VIII	110
	Apêndice IX	111
	Apêndice X	112
	Apêndice XI	113
	Apêndice XII	114
	Apêndice XIII	115

Apêndice XIV	116
Apêndice XV	117
Apêndice XVI	118
Apêndice XVII	119
Apêndice XVIII	120
Apêndice XIX	121
Apêndice XX	122
Apêndice XXI	123
Apêndice XXII	124
Apêndice XXIII	125
Apêndice XXIV	126
Apêndice XXV	127

LISTA DE ABREVIATURAS/SIGLAS

BPMN - Business Process Model and Notation

CA – Capability Area

CAR - Causal Analysis & Resolution

CM - Configuration Management

CMMI - Capability Maturity Model Integration

CONT - Continulty

Cypher – Cypher Query Language

DAR - Decision Analysis & Resolution

DDM – Departamento de desenho mecânico

DE – Departamento de eletrificação

DMM – Departamento de montagem mecânica

DMS - Delivering & Managing Services

DPR – Departamento programação e robótica

EDP - Engineering & Developing Products

ENQ - Ensuring Quality

EPF – Eclipse Process Framework

ESAF - Enabling Safety

ESD – Descarga eletrostática de dispositivos

ESEC - Enabling Security

EST - Estimating

EVSD - Enabling Virtual Solution Delivery

GOV - Governance

GP – Gestor de Projetos

IBM – International Business Machines Corporation

IDE - Integrated Development Environment

IEC - International Electrotechnical Commission

II - Implementation Infrastructure

IoT – Internet of Things

IL - Instruction List

IMP - Improving Performance
IRP - Incident Resolution & Prevention
MBR - Managing Business Resilience
MC - Monitor & Control
MPM - Managing Performance & Measurement
MSS - Managing Security and Safety
MST - Managing Security Threats & Vulnerabilities
MWF - Managing the Workforce
OpenUp - Open Unified Process
OT - Organizational Training
PA – Practice Areas
PAD - Process Asset Development
PC – Personal Computer
PCM - Process Management
PI - Product Integration
PLAN - Planning
PLC - Programmable Logic Controller
PMW - Planning & Managing Work
PQA - Process Quality Assurance
PR - Peer Review
RDM - Requirements Development & Management
RDMS – Sistema de gestão de base de dados
RE – Responsável de equipa
RSK - Risk & Opportunity Management
RUP – Rational Unified Process
SAM - Supplier Agreement Management
SDCI - Single-drop digital communication interface for small sensors and actuators
SDLC - Software Development Lifecycle
SDM - Service Delivery Management
SFC - Sequential Flow Chart
SHP - Sustaining Habit and Persistence
SI - Supporting Implementation

SMS - Selecting & Managing Suppliers

SSS - Supplier Source Selection

STSM - Strategic Service Management

SVN – Sistema de Controlo de Versões

SW – Esquematização do Software

TS - Technical Solution

UML - Unified Modelling Language

VV - Verification & Validation

XP – Extreme Programming

LISTA DE FIGURAS

Figura 1 - Esquema Business Core Baseado na cadeia de valor de Porter.....	3
Figura 2 - Processo de Design Science Research [9)	8
Figura 3 – Adapted Software development life cycle [14].....	11
Figura 4 - Model content relationships [21]	16
Figura 5 - Adaptado das Categorias and associated capability areas do CMMIv2.0 [21].....	17
Figura 6 - Adaptado do Practice Group level characteristics CMMIv2.0 [21]	18
Figura 7 - Disposição do OpenUp [25].....	20
Figura 8 - Ciclo de Vida OpenUp [23]	23
Figura 9 - Quatro fases do ciclo de vida OpenUp [25]	24
Figura 10 - Core Graph Concepts [31]	29
Figura 11 - Modelação BPMN dos processos da ITEC.....	34
Figura 12 - {UC0} – Processos de Desenvolvimento ITEC	36
Figura 13 – {UC1} – Gestão de Projetos.....	36
Figura 14 – {UC2} - Desenvolvimento.....	37
Figura 15 - {UC3} - Controlo de Qualidade/Testes	38
Figura 16 - {UC4} - Serviço Pós-Venda.....	39
Figura 17 - {UC0} – Processos de Desenvolvimento ITEC melhorados	40
Figura 18 - {UC1} - Gestão de Projetos Refinado.....	42
Figura 19 - {UC2} - Desenvolvimento Refinado.....	44
Figura 20 - {UC3} - Workshops	45
Figura 21 - {UC4} - Controlo de Qualidade/Testes Refinados	47
Figura 22 - {UC5} - Serviço Pós-Venda Refinado.....	48
Figura 23 - Mapeamento Níveis de maturidade CMMI	54
Figura 24 - Mapeamento da Relação entre Categorias do CMMI	55
Figura 25 -Mapeamento das Practice Summary da categoria Doing do CMMI	56
Figura 26 - Mapeamento das Practice Summary da categoria Managing do CMMI	57
Figura 27 - Mapeamento das Practice Summary da categoria Enabling do CMMI.....	58
Figura 28 - Mapeamento das Practice Summary da categoria Improving do CMMI.....	59
Figura 29 - Mapeamento das disciplinas do OpenUp (Tenho de alterar esta imagem)	60

Figura 30 - Exemplo de uma Practice Area (CMMI).....	62
Figura 31 - Tarefas do OpenUp por disciplina	63
Figura 32 - Relação entre CMMI e OpenUp	65
Figura 33 – Código Neo4j (Empresa, Colaboradores e Stackholders).....	66
Figura 34 - Código exemplo dos Casos de uso da ITEC	67
Figura 35 - Mapeamento da ITEC.....	68
Figura 36 – Tabela exemplo do cruzamento entre CMMI, OpenUp e ITEC.....	69
Figura 37 - Lista de tarefas da ITEC	70
Figura 38 - Exemplo de preenchimento Tabela.....	73
Figura 39 – Mapeamento da Capability Area DMS, Tarefas OpenUp e Tarefas da ITEC.....	74
Figura 40 – Mapeamento da Capability Area EDP, Tarefas OpenUp e Tarefas da ITEC.....	76
Figura 41 – Mapeamento da Capability Area ENQ, Tarefas OpenUp e Tarefas da ITEC	78
Figura 42 – Mapeamento da Capability Area MBR, Tarefas OpenUp e Tarefas da ITEC.....	80
Figura 43 – Mapeamento das Capability Area MWF, Tarefas OpenUp e Tarefas da ITEC.....	81
Figura 44 – Mapeamento das Capability Area PMW, Tarefas OpenUp e Tarefas da ITEC.....	83
Figura 45 – Mapeamento das Capability Area MSS, Tarefas OpenUp e Tarefas da ITEC.....	85
Figura 46 – Mapeamento das Capability Area SI, Tarefas OpenUp e Tarefas da ITEC.....	86
Figura 47 – Mapeamento das Capability Area IMP, Tarefas OpenUp e Tarefas da ITEC.....	88
Figura 48 – Mapeamento das Capability Area SHP, Tarefas OpenUp e Tarefas da ITEC	90

LISTA DE TABELAS

Tabela 1 - Comparação entre os casos de uso iniciais e os casos de uso refinados.....	41
Tabela 2 - Requisitos de Processo.....	50
Tabela 3 - RP1 Elaborar calendário dos colaboradores	51
Tabela 4 - Níveis de Prioridade.....	61
Tabela 5 - Níveis de cobertura das tarefas da ITEC.....	71
Tabela 6 - Níveis de Prioridade tarefas da ITEC	71
Tabela 7 - Níveis de urgência de realização das tarefas da ITEC	72
Tabela 8 - Cruzamento CA "SDM", Tarefas OpenUp e Tarefas da ITEC	75

1. INTRODUÇÃO

1.1 Enquadramento

Este tema de dissertação retrata a realidade de uma empresa ligada à robótica, a *Iberiana Technical (ITEC)*, sediada em Braga. Fundada em 2006, a ITEC dedica-se essencialmente à comercialização de equipamentos e serviços (programação de máquinas automatizadas) para a indústria eletrónica e automóvel, com especial evidência para as áreas de aparafusamento, doseamento de fluidos, proteção ESD, soldadura, automação e robótica.

Um dos seus principais objetivos visa desenvolver produtos diferenciados, para a criação de soluções tecnologicamente evoluídas, competitivas e inovadoras, para ir ao encontro das necessidades dos seus clientes.

Devido ao facto de a empresa não ter na sua equipa um especialista em métodos de desenvolvimento de software, ao longo dos tempos, têm encontrado algumas dificuldades ao nível da gestão dos seus processos de desenvolvimento. Por consequência, os colaboradores da empresa têm vindo a utilizar métodos que foram implementados para desempenhar essas mesmas funções, contudo, como não têm ninguém que domine esta área, sentiram a necessidade de pedir ajuda externa, dando assim origem a este projeto, o qual designamos de Projeto de Maturidade de Software.

Deste modo, é feito um estudo do desenvolvimento de software, de modo a fazer o enquadramento de acordo com o projeto em estudo. De seguida, realiza-se uma análise e levantamento pormenorizado de dois *frameworks* para o desenvolvimento de software conhecidos por Capability Maturity Model Integration (CMMI) e Open Unified Process (OpenUp), em que o CMMI define o que fazer e o OpenUp define como se deve fazer.

Posto isto, efetua-se um diagnóstico da situação atual da empresa, recorrendo a uma caracterização geral da indústria, onde se abordam as suas áreas de trabalho, assim como os seus departamentos e os *stakeholders*. De seguida, realiza-se um estudo sobre os processos de negócio, na

qual se descreve a situação da empresa com recurso ao *Business Process Model and Notation (BPMN)* [1], ao modelo *Business Core* (baseado na cadeia de valor de *Porter*) [2] e recorrendo ao *UML* para o levantamento dos requisitos de processos de desenvolvimento da empresa. Posteriormente a isto elabora-se e define-se quais as melhorias que são necessárias para otimizar e potenciar o processo de desenvolvimento da empresa. No entanto, estas melhorias resultam da análise efetuada dos seus processos de desenvolvimento e da forma de trabalhar da empresa em estudo, com o principal objetivo de dotar a empresa de melhores processos ao nível de desenvolvimento de software.

De seguida, são também apresentados os requisitos de processos utilizando o método de MoSCow [3], onde a informação apresentada tem por base o levantamento efetuado nos casos de uso modificados. Assim, define-se o que é realizado em cada requisito de processo, bem como os principais intervenientes e qual o nível de prioridade que deve ser realizado.

Por último, é realizada a demonstração de resultados, do mapeamento entre os dois *frameworks* referidos anteriormente e são levantadas as tarefas relativas à empresa em estudo, para que no final do projeto seja possível criar um guia de boas práticas com o objetivo de fornecer uma ferramenta de suporte para os procedimentos diários da empresa. Para a realização deste trabalho são utilizadas tabelas para demonstrar as relações efetuadas de uma forma mais informativa e é também utilizada uma tecnologia de base de dados por grafos, para a realização do mapeamento das relações efetuadas, sendo assim possível observar as relações criadas de uma forma visual e intuitiva, em que também se consegue filtrar as informações que se pretende observar.

1.2 Contextualização

Como referido anteriormente este trabalho é elaborado em parceria com a ITEC, e o motivo do seu surgimento deve-se ao facto da empresa pretender melhorar os processos desenvolvimento, nomeadamente no que diz respeito ao controlo e à forma como os elabora. Assim, surgiu a necessidade de implementar novas ferramentas, dando origem a este projeto.

Dedicando-se ao comércio de equipamentos e serviços para a indústria eletrónica e automóvel, a ITEC tem um papel fundamental no que toca à área da robótica avançada da indústria 4.0, uma vez que, com as soluções robóticas que apresenta consegue desenvolver protótipos em diversos ramos. O

trabalho desenvolvido por esta empresa assenta no desenvolvimento de projetos e têm início quando o cliente faz um pedido, que é personalizado em função das suas necessidades. Após esta fase, inicia-se o processo de planeamento e desenvolvimento, onde se definem os objetivos a atingir com os respetivos prazos e estratégias de desenvolvimento mais adequadas para a programação. Define-se, também, juntamente com as equipas, os materiais a encomendar. Após a conclusão deste processo procede-se à montagem do produto, realizando todos os testes de qualidade necessários. Finalmente, com o produto devidamente testado e aprovado, é instalado no cliente final, onde este tem garantia de suporte contínua. Tendo em consideração este contexto, foi realizado um esquema Business Core baseado na cadeia de valor de Porter, que pode ser observado na Figura 1 [2].

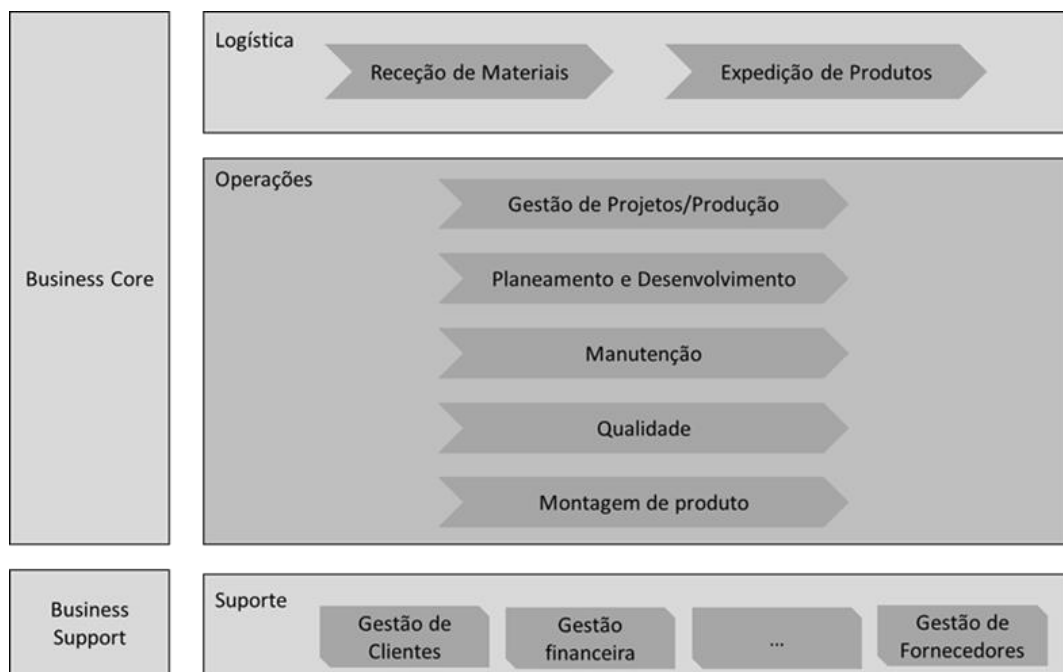


Figura 1 - Esquema Business Core Baseado na cadeia de valor de Porter

Para uma caracterização detalhada de todas as tarefas e processos presentes na empresa recorreremos a dois tipos de modelações distintos: o mapeamento dos macroprocessos da empresa e os seus casos de uso. No que toca aos macroprocessos da empresa utilizamos o BPMN para modelar de forma detalhada as interações entre cada uma das áreas internas da empresa e a forma de trabalhar em cada uma delas.

No entanto, ao utilizar os casos de uso conseguimos especificar ainda mais cada uma das referidas áreas presentes no BPMN, onde identificamos os atores (Colaboradores/trabalhadores) que

executam as tarefas inerentes às áreas, nomeadamente, no que diz respeito às atividades de gestão de projetos, desenvolvimento de software, controlo de qualidade e serviço de pós-venda. Desta forma é nos possível ter uma visão mais ampla de todo o contexto organizacional da empresa. Posto isto, de seguida, são apresentados todos os intervenientes nos procedimentos da ITEC.

Como em todas as empresas, a ITEC é constituída por uma série de *stakeholders* que passamos a detalhar:

- **Cliente:** É o comprador e quem solicita os produtos desenvolvidos na empresa, onde pode pedir produtos ao nível de programação de máquinas, novos protótipos ou serviços pós-venda/manutenção;
- **Cliente final:** é o beneficiário final dos produtos, com as características solicitadas pelo cliente, e onde o produto é instalado;
- **Comercial:** faz a intermediação entre a empresa e o cliente, onde obtém as informações necessárias para o desenvolvimento do produto. É, também, responsável pelo envio da emissão de ordem de produção para o Gestor de Projeto;
- **Gestor de Projeto:** é responsável pelo planeamento do projeto, nomeadamente, pela análise do caderno de encargos e planeamento temporal. É, também, responsável pela transmissão das datas e informações do projeto ao chefe de equipa e pelas reuniões de *Kickoff* e pela reunião global de projetos;
- **Chefe de equipa:** é responsável por escolher os programadores e os membros da equipa para um determinado projeto, pelo planeamento de tarefas e a respetiva subdivisão entre os vários membros e pela definição de orientações na validação do *software* da máquina e dos testes elétricos e mecânicos, e posterior comprovação dos resultados.
- **Programador:** é responsável pela análise do caderno de encargos e das tarefas de modo a avançar com a esquematização do *software* e efetiva programação. Diariamente deve registar a taxa de realização, atualizar o código no sistema de controlo de versões, e eventualmente comentar com a justificação dos atrasos. Semanalmente deve fazer um levantamento das dificuldades que teve ao longo da semana de trabalho. Numa fase seguinte, segundo as orientações do chefe de equipa, o programador valida o *software* e realiza os testes elétricos e mecânicos. Na fase final, e após todo o produto estar devidamente validado, o programador faz a instalação no cliente final e fornece uma assistência contínua.

Depois de apresentados os principais intervenientes, passamos então a enumerar os departamentos da ITEC, efetuando uma descrição e contextualização dos mesmos. A empresa é constituída por vários departamentos, nomeadamente, o Departamento de Desenho Mecânico (DDM), o Departamento de Montagem Mecânica (DMM), o Departamento de Eletrificação (DE) e o Departamento Programação e Robótica (DPR).

No Departamento de Desenho Mecânico começa-se por fazer um planeamento semanal dos projetos de mecânica que têm no seu plano de trabalhos. De seguida passa-se para a elaboração dos desenhos *3D* e de conjunto. Com isto faz-se, a execução dos ficheiros *2D/IDW* e passa-se essa informação para *PDF/STEP*. Neste departamento realiza-se também o controlo de qualidade. Depois deste trabalho, efetua-se a encomenda das peças necessárias para a construção da máquina, o acompanhamento da execução das peças e a inspeção da conformidade das mesmas a serem utilizadas. Por fim, e apenas se for necessário, pode ocorrer o retrabalho/revisão do desenho *3D*. Após todo este processo avança-se para o Departamento de Montagem Mecânica.

Tal como no Departamento de Desenho Mecânico, no Departamento de Montagem Mecânica também se realiza inicialmente um planeamento semanal. Aqui, com o suporte dos desenhos de conjunto (*2D*), faz-se o acompanhamento da montagem das peças, seguida de uma inspeção da conformidade da montagem e, caso se detete algum erro, é necessário retrabalhar e/ou proceder a uma revisão do trabalho, realizando-se um novo acompanhamento da montagem da peça e respetiva inspeção. Este processo encontra-se em *loop* até estar de acordo com o solicitado, para que possa passar para a etapa seguinte do processo de produção.

Após a parte mecânica estar finalizada, o processo continua no Departamento de Eletrificação que tem uma estrutura de trabalho semelhante aos departamentos de mecânica, com a particularidade de se focar nas exigências da eletrificação da máquina. Desta forma, o processo é iniciado com a identificação das necessidades elétricas do projeto, seguido da encomenda das matérias-primas e sua respetiva receção e armazenamento.

Daqui inicia-se a esquematização elétrica do projeto, com o cuidado de se validar sempre a informação antes de qualquer processo de eletrificação. Tendo esta parte bem definida, é feito um

planeamento semanal de forma a acompanhar a eletrificação da máquina e a respetiva inspeção de conformidade da mesma.

Finalmente, depois da parte de eletrificação estar pronta, a máquina prossegue para o Departamento Programação e Robótica. Este departamento engloba todas as atividades respeitantes ao desenvolvimento da programação das máquinas, desde a formação da equipa técnica de programação até à instalação da máquina no cliente e respetiva assistência em garantia. Este departamento, por ser um dos pontos fulcrais de análise do Projeto de Maturidade de *Software*, é detalhado de forma extensiva no BPMN.

Com vista a alcançar-se esse objetivo identificam-se também, as normas de utilização de *Programmable Logical Controller* (PLC) que são uma forma característica de controlar baseada em microprocessadores que utilizam memória programável para reter informações sobre a forma de instruções que permitem a execução de funções por forma a controlar máquinas e processos de desenvolvimento [4].

No entanto, estes controladores foram projetados para serem simples e intuitivos com objetivo de serem utilizados por engenheiros que não sejam especialistas em computação, que não tenham muito conhecimento das suas linguagens, onde são desenhadas para que apenas os programadores possam configurar e alterar os seus programas [5]. Com objetivo de rentabilizar tempo e recursos, o *International Electrotechnical Commission* (IEC), que é um organismo internacional de normalização, aprovou um conjunto de padrões com a intenção de uniformizar protocolos e linguagens de programação. Foi então que, em 1990, surgiu a norma IEC61131 que resume os requisitos de sistema PLC, referentes ao seu *hardware* e ao sistema de programação [6].

Por sua vez, em 1969 marcou o início da automação e da combinação elétrica com PLC, que ficou conhecido pelo ano da terceira revolução industrial. Atualmente, o mundo depara-se com a quarta revolução industrial, chamada de indústria 4.0 que está relacionada com a conectividade dos parques produtivos com IoT [7].

Ao dedicar-se ao comércio de equipamentos e serviços para a indústria automóvel e eletrónica, a ITEC tem um papel fundamental no que diz respeito à área da robótica avançada da indústria 4.0, uma

vez que através das soluções robóticas que apresenta, consegue desenvolver protótipos em diversos ramos, sendo uma mais-valia para este ramo em que se enquadram.

O trabalho desenvolvido por esta empresa assenta no desenvolvimento de projetos e têm início quando o cliente faz um pedido, que é personalizado em função das duas necessidades. Após esta fase, inicia-se o processo de planeamento e desenvolvimento, onde se definem os objetivos a atingir com os respetivos prazos e estratégias de desenvolvimento mais adequadas para a programação. Define-se também juntamente com as equipas, os materiais a encomendar. Após a conclusão deste procedimento procede-se à montagem do produto, e por fim realizam-se todos os testes de qualidade necessários. Finalmente com o produto devidamente testado e aprovado, é configurado no cliente final que tem garantia de suporte contínuo.

Contudo, surgiu a necessidade de melhorar os seus processos de desenvolvimento, pois não tem ninguém na sua equipa de trabalho que domine esta área. No entanto, é cada vez mais importante desenvolver mecanismos de controlo e gestão de trabalho a serem efetuados, assim como também da produtividade da empresa e todas as áreas que precisam de especial atenção dos projetos na qual estão a trabalhar.

Para uma melhor compreensão do contexto da empresa, são realizadas diversas reuniões com os principais responsáveis pela gestão das equipas da empresa, em que são também solicitados documentos relativos à empresa e são também realizadas entrevistas estruturadas com os *developers* da empresa, com o objetivo de retirar o máximo de informação possível para a elaboração deste projeto.

1.3 Objetivos

Nesta dissertação é realizada uma descrição da ITEC recorrendo ao BPMN e casos de uso, dividindo este último em casos de uso *as is* e *to be*. Baseado nos casos de uso *to be*, são especificados os requisitos de processos de desenvolvimento. Posto isto, é realizada uma sistematização do CMMI e OpenUp, onde se utiliza a ferramenta *Neo4j*, para mapear toda a informação recolhida. Contudo, são

apresentados ao detalhe os dois *frameworks* mencionados, por serem considerados guias de boas práticas, importantes na melhoria dos procedimentos diários da empresa.

Tendo em consideração a contextualização anteriormente efetuada, os objetivos a alcançar com o desenvolvimento desta dissertação são:

1. Modelar os processos de desenvolvimento de software adotados pela ITEC (*as is*) e descrever os requisitos para a correspondente melhoria (*to be*).
2. Sistematizar os *frameworks* CMMI e OpenUp utilizando a ferramenta *Neo4j*.
3. Propor uma abordagem para a melhoria dos processos de desenvolvimento de software da ITEC.

1.4 Abordagem Metodológica

No contexto deste projeto de investigação, a abordagem metodológica a recolhida foi o *Design Science Research*, segundo os autores Vaishnavi e Kuechler, em 2009, este é um novo ponto de vista ou um conjunto de técnicas analíticas que permitem o desenvolvimento de pesquisas em diversas áreas, em particular na engenharia [8]. Esta metodologia, segundo Takeda em 1990, rege-se por cinco fases distintas sendo elas: *awareness, suggestion, development, evaluation e conclusion*, como podemos visualizar na Figura 2 [9]. Numa versão mais recente, elaborada por Peffers, em 2008, sejam mencionados seis passos sendo eles: *identify problem & motivate, define objectives of a solution, design & development, demonstration, evaluation e communication*. Nesta última, na identificação de passos do processo é acrescentada uma fase para demonstração, mas como a versão mais referenciada é a primeira, iremos guiar-nos pelos seus passos.

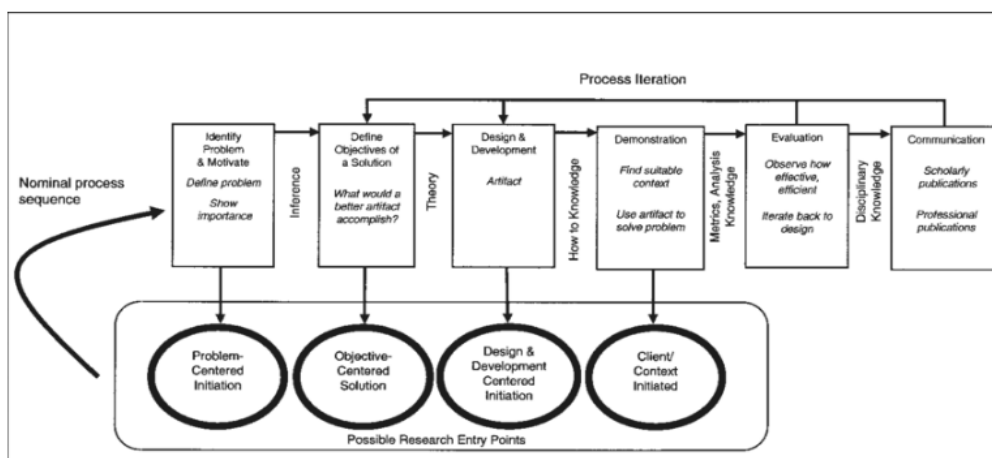


Figura 2 - Processo de Design Science Research [9]

Na primeira fase, é executado o trabalho de obter conhecimento de como estão a ser executadas as tarefas de uma organização, tentando assim perceber a problemática envolvente, para conseguirmos efetuar a melhor formalização do problema, conhecendo as suas fronteiras e quais as soluções pretendidas.

A segunda fase, de acordo com a informação que conseguimos obter, é onde formalizamos uma proposta de solução ao problema, de modo que antes do início do desenvolvimento da mesma não haja fatores que não foram considerados podendo levar a perdas ou excesso de uso de recursos. O resultado da Sugestão é o conjunto de possíveis artefactos e a escolha de um, ou mais, para serem desenvolvidos.

Na terceira fase, o Desenvolvimento corresponde ao processo de constituição do artefacto em si, ou seja, de acordo com o que foi decidido na fase anterior, é iniciado o desenvolvimento do projeto proposto.

Na fase quatro, é feita avaliação do *output* da fase três, verificando assim a performance do artefacto desenvolvido de acordo com o que foi proposto, concedendo uma nota para os esforços realizados em torno do projeto e efetuando uma estimativa do valor adquirido com o projeto.

Na quinta e última fase do *Design Science Research* é realizada uma síntese de todo o trabalho efetuado e dos resultados obtidos, assim como o trabalho que terá de ser realizado posteriormente para a manutenção e melhoria do artefacto.

1.5 Estrutura do Documento

Este documento contém cinco capítulos, sendo que este primeiro refere-se à introdução do trabalho desenvolvido, onde é identificada toda a informação que serve de conhecimento para o desenvolvimento deste tema de dissertação. No capítulo 1 começa-se por demonstrar o enquadramento, fazendo referência à organização na qual este projeto está a ser desenvolvido, bem como da sua história e filosofia. Posto isto, define-se qual a contextualização e objetivos do projeto, onde se identifica o problema em mãos e qual o valor que as base de dados por grafos trazem para a empresa, ao fazer a sistematização do conhecimento do domínio pretendido. De seguida passa-se então para a descrição da abordagem metodológica utilizada e por último define-se a estrutura do documento.

No capítulo 2, encontra-se o desenvolvimento do estado da arte, assente em conceitos que são utilizados para a realização do projeto. Os conceitos essenciais com vista ao desenvolvimento dos

objetivos pretendidos passam pelo ciclo de vida de software, por *frameworks* como o CMMI e OpenUp para o desenvolvimento de software e por base de dados por grafos.

No terceiro capítulo, é realizada uma contextualização da empresa, onde se descreve o seu contexto atual com recurso ao BPMN, ao modelo *Business Core* e ainda modelações de casos de uso. Relativamente aos casos de uso, estes são criados de acordo com as reuniões realizadas, documentos fornecidos e por entrevistas estruturadas aos colaboradores da empresa, posto isto, depois de tudo analisado são sugeridas algumas alterações dos mesmos, dando assim origem os casos de uso *as is* e casos de uso *to be*. Por fim, são utilizados como referência os casos de uso *to be*, com o objetivo de levantar os requisitos de processos presentes na empresa.

No quarto capítulo, são demonstrados os resultados obtidos neste trabalho, onde é dada a conhecer a forma como os dados foram introduzidos no software escolhido para a base de dados por grafos e explicado todos os procedimentos de cada passo, com objetivo final da criação de um documento que servirá como suporte para os procedimentos diários da empresa em questão. Assim, são utilizadas duas ferramentas, uma que seja possível criar tabelas e outra com o objetivo de realizar o mapeamento entre os *frameworks* já mencionadas bem como também as tarefas levantadas relativas à empresa em estudo.

O capítulo 5 é onde são apresentados os resultados obtidos de todo o trabalho desenvolvido, são referidos os problemas encontrados e mencionado o trabalho futuro.

2. ANÁLISE DO ESTADO DA ARTE

2.1 Introdução

Neste capítulo são descritos os conceitos necessários para o desenvolvimento deste tema de dissertação, dando um ponto de vista teórico que serve como base de conhecimento para a elaboração deste documento. Este conhecimento será devidamente referenciado, de modo a poder apresentar o maior nível de confiabilidade da informação contida.

O primeiro conceito a ser exposto é do ciclo de vida de software, onde é explorada a importância da análise e levantamento de requisitos, no âmbito da concepção de novos sistemas em engenharia de software. De seguida, é detalhado ao pormenor e são utilizados estudos sobre o *framework* CMMI, para que seja possível contextualizá-lo e demonstrar que benefícios pode trazer para a empresa, o mesmo acontece para o próximo *framework* em estudo, o *OpenUp*. O CMMI permite-nos perceber o que devemos fazer e o *OpenUp* como devemos fazer.

Por fim, é realizado um estudo sobre bases de dados por grafos, sendo utilizados conceitos sobre este assunto com estudos anteriormente realizados, é feita também uma comparação entre duas tecnologias de base de dados por grafos, conhecidos por *Neo4j* e *Cosmos DB Azure*, em que no final é feita uma análise e seleção de qual dos softwares se deve utilizar neste caso em concreto, tendo como referência estudos anteriormente realizados.

Atualmente, há uma crescente preocupação por parte das instituições em criar guias com normas que permitam o desenvolvimento de *software* de qualidade, como são os casos do *CMMI* e *OpenUp*. Para este contexto optamos por recorrer aos dois *frameworks* mencionados de forma a definir quais as áreas em que os dois se assemelham, usando para o efeito uma tecnologia de bases de dados por grafos (*Neo4j*). Esta análise permite-nos definir, as boas práticas que a empresa deve adotar de forma a garantir a correta implementação dos requisitos identificados ao longo deste projeto.

2.2 Desenvolvimento de Software

Em engenharia de software, a análise de requisitos abrange componentes que lidam com a investigação, definição e âmbito de novos sistemas. No entanto, a análise de requisitos é a parte associada ao processo de desenvolvimento de software na qual se realiza, juntamente com o cliente, uma análise e um levantamento das necessidades da organização, com vista a obter os resultados pretendidos [10]. Portanto, esta fase é umas das primeiras atividades do desenvolvimento de software. O resultado que é esperado nesta etapa é a especificação de requisitos que define o contexto do projeto em duas dimensões: os requisitos funcionais (funcionalidade do sistema) e os requisitos não funcionais (por exemplo, segurança e disponibilidade dos dados). É nesta fase que a equipa de analistas providencia as primeiras reuniões com o cliente ou utilizadores do software para conhecer as funcionalidades do sistema que será desenvolvido.

Nos dias de hoje, existem diversos modelos e normas que servem de referência para apoiar o processo de gestão da configuração do software, sendo que os mais conhecidos são IEEE 828-2012, CMMI-DEV, ISO 10007: 2003 e ISO / IEC 15504. No entanto, a utilização destas normas está dependente das empresas em causa, pois tem que se ter em consideração as suas infraestruturas e os colaboradores [11].

Quando se trata de um ambiente empresarial, que descreve as atividades executadas em cada fase do processo de desenvolvimento e manutenção de software, refere-se ao ciclo de vida de desenvolvimento de software (SDLC) [12]. Por este motivo, é considerado um plano rigoroso em que é descrita e detalhada a forma como o desenvolvimento, manutenção e substituição de software são e devem ser realizados. Contudo, este processo tem como o objetivo desenvolver um sistema de informação, sendo que a resposta deverá corresponder às necessidades e expectativas do próprio cliente, de modo que o projeto aconteça no tempo contratado e com os respetivos custos estimados no início do processo, havendo diversos modelos que descrevam várias abordagens para o SDLC. Com isto, pode-se destacar que é importante diferenciar o significado atribuído ao modelo e metodologia utilizada, em que um modelo descreve o que fazer, enquanto que uma metodologia descreve ao pormenor como fazê-lo [13].

No entanto, a escolha da metodologia pode variar consoante a empresa em questão, uma vez que as realidades presentes também são diferentes, por isso mesmo é necessário, num primeiro momento, perceber qual o contexto do negócio.

Como se pode observar na Figura 3, o ciclo de vida para desenvolvimento de software é composto por sete fases importantes e essenciais para quem vai realizar a análise e o desenvolvimento de *software*: planeamento, análise e levantamento de requisitos, *design* e arquitetura, desenvolvimento e implementação, teste de produto e, ainda, operações e manutenção [14].

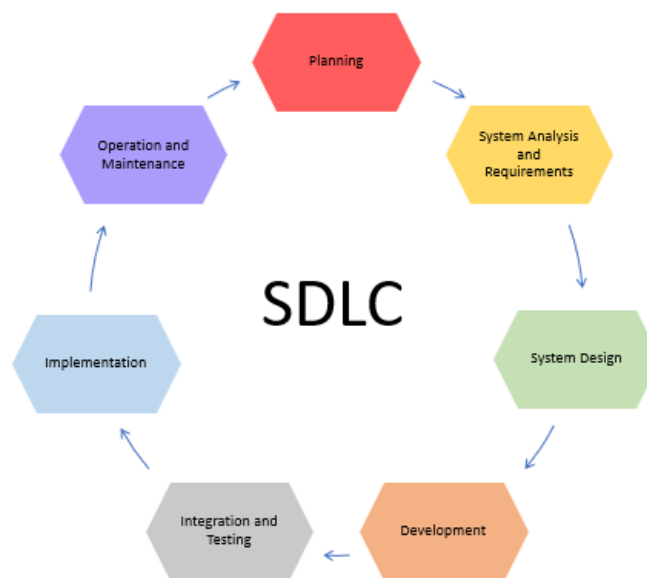


Figura 3 – *Adapted* Software development life cycle [14]

Este ciclo tem início com a fase do *Planning*, em que o objetivo é descobrir qual o contexto do problema e determinar soluções, recursos, custos, tempos e benefícios. Concluída esta fase, passa-se para a *System Analysis* e dos respetivos requisitos, em que são considerados e definidos os requisitos referentes ao projeto. A fase seguinte, a que chamamos de terceira fase, é *Systems Design*, onde descreve com detalhe as especificações, recursos e operações necessárias para responder aos requisitos funcionais, que estarão à disposição após a implementação. Posto isto, passamos então para a fase de desenvolvimento, onde está marcada o fim da parte inicial do processo, portanto, significa que se inicia o processo de desenvolvimento da solução, tendo em consideração todos os requisitos que são levantados nas fases anteriores.

A fase cinco inclui a integração e testes do sistema (de programas e procedimentos), em que geralmente são realizados pelo *Quality Analyst* que determina se o projeto proposto responde ao conjunto inicial de objetivos. Contudo, nesta fase ainda é necessário realizar a verificação e validação para garantir que a conclusão seja bem-sucedida. Na última fase do SDLC requer a manutenção e atualizações regulares. Nesta fase, os utilizadores finais podem ajustar o sistema, caso considerem necessário para aumentar o desempenho, adicionar novos recursos ou considerar requisitos adicionais [15]. Ao longo do tempo, têm surgido diversos modelos de desenvolvimento de software, como exemplo, a metodologia *Waterfall*, *V-Model* e *Scrum* [16].

Desde o surgimento da manipulação de robôs nas fábricas para a ajudar na produção de automóveis, estes têm sido vistos como ferramentas valiosas para o fabrico em ambientes industriais. Sendo que as normas *International Organization for Standardization (ISO)* existentes foram desenvolvidas em parceria com a indústria, e toda a documentação relativa a robôs tem vindo a ser aceite até à data, e é aplicável a robôs industriais [17].

Estas normas foram desenvolvidas com o objetivo de ter um foco industrial, por isso, toda a documentação relacionada com este tema tem sido amplamente aceite até à data. Os robôs, com o passar dos anos, têm vindo a surgir com mais ênfase nas empresas para agilizar e otimizar a produção, sendo por isso considerados como uma ferramenta valiosa. As *tarefas* desempenhadas por estes robôs envolvem levantar peças pesadas, manipular vários componentes metálicos e não metálicos, entre outras, o que faz com que tenham de ser máquinas de grandes dimensões e com muita potência, representando um risco considerável para a segurança do operador [17]. Para assegurar a integridade física do operador foram implementadas várias medidas de segurança, de forma a mantê-lo a uma distância segura, enquanto os robôs estão em funcionamento. Desta forma, foram aparecendo requisitos emergentes no que toca a novas aplicações industriais.

2.3 Framework CMMI

Ao longo dos últimos anos, a utilização de software tem vindo a aumentar consideravelmente em praticamente todas as áreas, e como a procura por sistemas complexos com grande responsabilidade no contexto das empresas tem vindo a crescer, a qualidade do *software* começa a ser vista como um fator essencial no seu desenvolvimento.

O conceito de qualidade tem hoje uma importância fulcral para catapultar a competitividade das empresas e, por isso, a preocupação com a qualidade deixou de ser algo acessório e passou a ser um pré-requisito básico para a participação no mercado. Todavia, um dos primeiros entraves encontrados na definição e implementação de um sistema de informação de qualidade foi entender o que, efetivamente, significa qualidade de *software*. A qualidade de *software* pode ser medida através da quantificação das características desejáveis que um *software* possui. Esta quantificação pode ser apurada, quer por meios qualitativos, como por meios quantitativos, ou uma combinação de ambos [18].

Análise e especificação do *CMMI*

Atualmente, as empresas da área das tecnologias do século XXI começaram a fabricar produtos e serviços mais complexos, sendo raro uma única organização desenvolver todos os componentes que compõem um produto ou um serviço complexo. É mais frequente que alguns componentes sejam construídos internamente e os restantes sejam adquiridos externamente. Desta forma, os problemas que as organizações modernas enfrentam levam a soluções complexas, a nível empresarial, os quais requerem uma abordagem integrada. Deste modo, é fundamental que exista uma gestão eficaz dos ativos organizacionais para que se consiga atingir o tão desejado sucesso empresarial [19].

De forma a colmatar esta situação, atualmente existem modelos de maturidade, *standards*, metodologias e diretrizes que podem ajudar uma empresa a melhorar a forma como efetua a gestão do negócio. Contudo, a maior parte destas ferramentas não adotam uma abordagem sistémica dos problemas que a maioria das organizações enfrenta. Com isto, estes modelos levam as empresas a focarem-se na melhoria de uma das áreas de negócio, perpetuando barreiras que ainda hoje existem nas organizações. Assim surgiu o *CMMI* que proporcionou uma oportunidade de evitar ou eliminar estas

barreiras, sendo constituído pelas melhores práticas que incidem sobre as atividades de desenvolvimento aplicadas a produtos e serviços, incluindo práticas que cobrem o ciclo de vida do produto, desde a sua conceção até à sua entrega e manutenção. Desta forma, os modelos CMMI são uma compilação das melhores práticas no âmbito da melhoria de processos em organizações e empresas. Estes modelos são desenvolvidos por especialistas, com origem em vários ramos da indústria e com as parcerias do governo e do *Software Engineering Institute (SEI)* [20].

O CMMI é constituído por um conjunto integrado de práticas que são recomendadas e que permitem às empresas melhorar o seu desempenho dos seus principais processos de negócio. Este modelo foi desenvolvido por equipas de produção com colaboradores da indústria e do instituto CMMI, a qual atualmente integra o ISACA [21].

A arquitetura e o design do *CMMI V2.0 Product Suite* é um afastamento radical dos seus processos iniciais de desenvolvimento para tornar mais útil e adaptável pelos clientes, sendo que um dos principais inconvenientes de modelos complexos de maturidade são o tempo e os recursos necessários para fazer atualizações e mantê-los assim atualizados com as tendências tecnológicas, comerciais e as exigências do mercado. Para enfrentar este desafio, a arquitetura do *CMMI* foi especificamente concebida para ser flexível, ágil e evoluir à medida que estes e outros fatores se vão alterando. Isto permite um rápido desenvolvimento e adição de novos conteúdos relevantes à velocidade dos negócios, da tecnologia e da mudança [21].

Este fornece orientação para serem aplicadas as melhores práticas numa empresa ou organização, para assegurar a qualidade e soluções atempadas que seduzem os clientes e os utilizadores finais. Assim sendo, cada empresa ou organização pode beneficiar da melhoria do desempenho e da consequente redução do risco. O *CMMI* fornece um guia que orienta para melhorias, desde atividades *ad hoc* até processos disciplinados e consistentes de modo a alcançar os objetivos empresariais relacionados com: desempenho, qualidade, custo, horário e funcionalidade [21].

A utilização do *CMMI* proporciona inúmeros benefícios, sendo que salientamos os seguintes:

- Um retorno positivo dos investimentos de melhoria do desempenho e respetivos processos.
- O cumprimento de compromissos que resultam em:
 - Entregas mais atempadas;
 - Menos *crunches* de última hora;
 - Maior controlo de custos;
 - Aumento da qualidade das soluções.
- Visibilidade de gestão que resulta em:
 - Respostas mais rápidas a questões e riscos;
 - Menos surpresas;
 - Soluções de alta qualidade que satisfaçam as necessidades e expectativas dos clientes;
 - Redução das reclamações dos clientes;
 - Reduzir o retrabalho;
 - Menor rotatividade dos empregados;

O livro do CMMI está dividido em quatro *categorias* (*Doing, Managing, Enabling e Improving*), e cada categoria está dividida em áreas de capacidade, sendo que cada área de capacidade está distribuída em diversas áreas de atuação, de acordo com a Figura 4 [21].

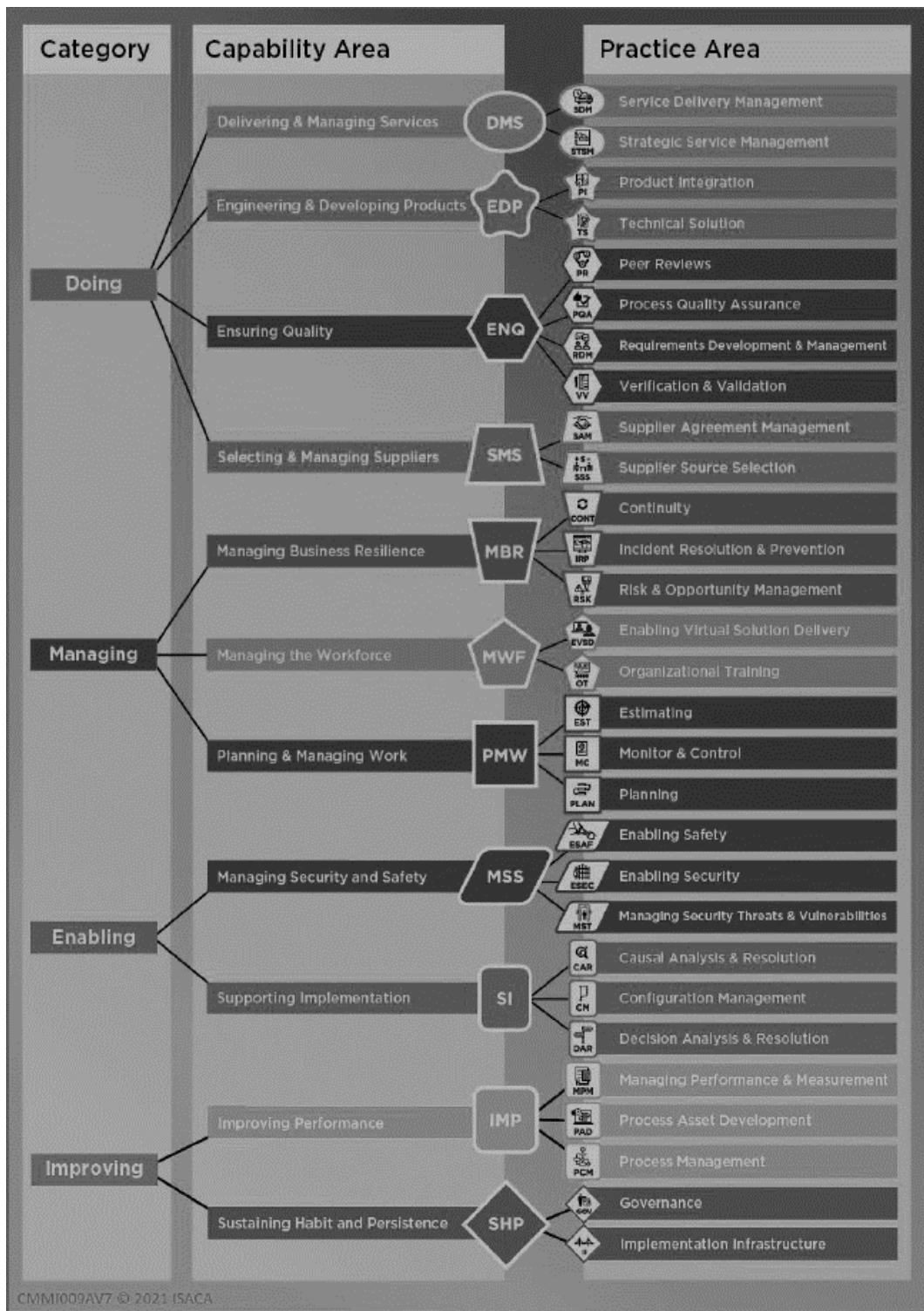


Figura 4 - Model content relationships [21]

As categorias estão divididas em quatro categorias de atuação, sendo que a *Doing* é referente às áreas de capacidade para produzir e fornecer soluções, a *Managing* para o planeamento e gestão da implementação de soluções, a *Enabling* é referente ao apoio à implementação e entrega de soluções e por fim o *Improving* para sustentar e melhorar o desempenho. A Figura 5 mostra-nos as *Categorias* e

associação com as *Capability Areas (CA)*, na qual as CA correspondem às atividades que são desenvolvidas em cada uma das quatro fases presentes no CMMI [21].



Figura 5 - Adaptado das Categorias and associated capability areas do CMMIv2.0 [21]

Relativamente às *Practice Areas*, estas, estão divididas em cinco níveis de maturidade (Figura 6), de acordo com as metas pretendidas pelas empresas que tenham intenção de implementar este guia de boas práticas. Assim, em cada nível de maturidade é notório o aumento da dificuldade de implementação da mesma, consoante estas se aproximam do nível máximo [21].

Quando estamos a falar do nível 1 de maturidade, o resultado da realização de trabalhos é imprevisível e reativa. Neste nível, apesar do trabalho ser concluído, muitas vezes é atrasado e ultrapassa o orçamento, onde a organização está em um momento crítico, pois não tem processos definidos, e também não consegue realizar planos adequados.

De seguida, quando falamos em nível 2 de maturidade, os trabalhos já são geridos no nível de projetos, onde existe planeamento, realização, medição e controlo dos projetos, no entanto neste nível as empresas ainda carregam processos iniciais, mas com a diferença que controlam com facilidade os custos e os prazos dos projetos para entrega de produtos e serviços.

Passando então para o nível 3 de maturidade, tem um nível mais proativo e menos reativo que os anteriores, em que o seu ponto chave é a definição de padrões de processo para toda a organização que fornece orientação, entre projetos, programas e entre outros.

Quando falamos no nível 4 de maturidade, este já é considerado de alta maturidade, em que organizações e projetos de alta maturidade usam análise quantitativa e estatística para determinar, identificar e gerir a tendência.

Por último, temos o nível 5 de maturidade é mais alto nível de maturidade presente no *CMMI*, em que as organizações focam numa melhoria continua para alcançar processos flexíveis capazes de responder às oportunidades e mudanças.

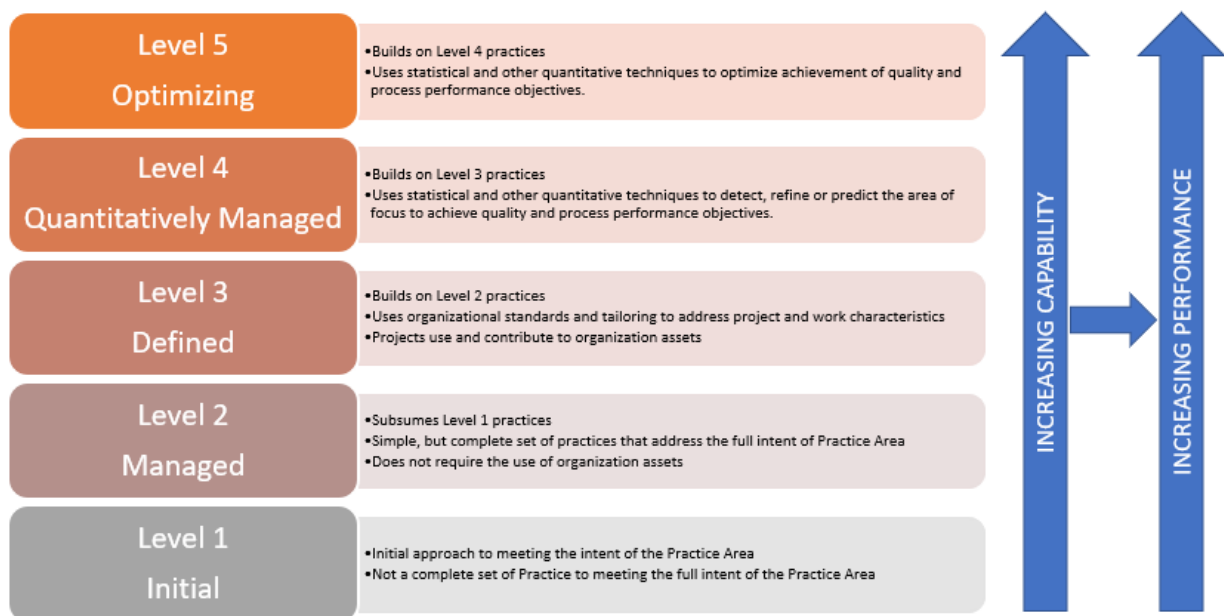


Figura 6 - Adaptado do Practice Group level characteristics CMMIv2.0 [21]

2.4 Framework OpenUp

O OpenUp foi inicialmente desenvolvido pela *International Business Machines Corporation* (IBM) com base no *Rational Unified Process* (RUP) e no *Extreme Programming* (XP), o qual tem como principal objetivo reunir as melhores características de cada uma dessas abordagens. Assim, este processo unificado aplica uma abordagem iterativa e incremental dentro de um ciclo de vida estruturado. Porém, abraça uma filosofia ágil que dá ênfase na natureza colaborativa do desenvolvimento de *software* [22].

Além disso, é um processo que pode ser alargado para se direcionar a uma grande variedade de projetos. É também importante referir que este segue uma linha pragmática, e por isso, tenta ser um caminho de migração para metodologias ágeis, onde preserva alguns formalismos, principalmente no que diz respeito à documentação de requisitos e arquiteturas.

O *OpenUp* é planeado através da ferramenta EPF *Composer* que é uma ferramenta *open-source*, onde tem como suporte a fundação Eclipse, que possibilita a gestão de processos e tem como principais características uma fácil aprendizagem, métodos simples, customização e criação de processos, além de permitir efetuar uma gestão automática de documentos do processo definido [23]. Esta documentação consiste num *website* sendo constituído por cinco elementos básicos:

- **Produtos de trabalho:** são os artefactos produzidos
- **Tarefas:** como executar o trabalho
- **Papel:** quem executa o trabalho
- **Processo:** são utilizados para definir os fluxos de trabalho
- **Diretrizes:** *templates, checklists*, exemplos, guias, conceitos, entre outros....

É uma metodologia que segue o processo unificado da engenharia de *software*, porém com uma cultura aberta e limpa, utiliza a filosofia ágil e que enfatiza a natureza colaborativa de desenvolvimento de *software*. A metodologia é construída através de processos simples os quais podem ser expandidos para quaisquer tipos de *software* [24].

O *OpenUp* mantém o seu núcleo aberto à comunidade, podendo ser facilmente adaptado para qualquer tipo de desenvolvimento de *software*, tornando-se igualmente possível adicionar práticas e processos únicos aplicados pela cultura organizacional da empresa. De acordo com L. Diedrich [25], este processo estimula a colaboração intensiva assim que o sistema é desenvolvido, onde utiliza um ciclo incremental numa determinada equipa.

Através da metodologia são estabelecidos conjuntos de funções, *tarefas* e *artefactos*, sendo estes utilizados pelos membros da equipa durante a execução e gestão de projetos (Figura 7). Como a metodologia é aberta também é disponibilizada uma página a qual contém todo o conteúdo necessário para a sua implementação e utilização [25].

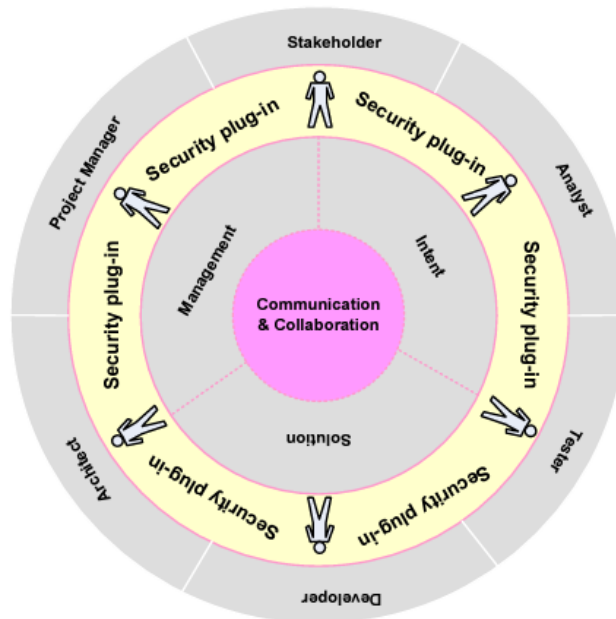


Figura 7 - Disposição do OpenUp [25]

Uma forma de compreender o *OpenUp* é pensar nele como sendo uma ferramenta dirigida para equipas que tem como objetivo, aplicar o processo minimamente necessário que traz valor, evitar que haja sobrecarga com produtos de trabalho numa linha produtiva, utilizar um processo que possa ser adaptado e alargado para necessidades adicionais que possam surgir durante o ciclo de vida de desenvolvimento de software.

Existem várias possibilidades de ambientes de projetos, sendo que o *OpenUp* é um processo que tem as seguintes características:

Mínimo

Apenas está incluído o conteúdo que seja fundamental. É considerado um processo ágil e leve que promove as melhores práticas de desenvolvimento de software, como por exemplo:

- Desenvolvimento iterativo;
- Colaboração da equipa;
- Integração e testes contínuos;
- Entregas frequentes de *software* de trabalho;
- Adaptação às mudanças, e assim por diante.

Outros fatores que também contam, ao determinar a dimensão do material do processo, tais como o número de elementos do método como funções, tarefas, artefactos e guias. O *OpenUp* fornece descrições de responsabilidades, aptidões e competências dos papéis fundamentais de uma equipa.

Este disponibiliza os artefactos essenciais necessários para captar e comunicar decisões. Em última análise, o processo não é regido pela criação de artefactos. Por exemplo, pensar no design é mais importante do que documentar o design; avaliar e planear uma iteração é promover a colaboração da equipa, em vez de criar planos que são escritos em pedra. Além disso, para cada artefacto, o *OpenUp* sugere representações ou modelos informais como pontos de partida para as equipas que não precisam de criar os seus próprios modelos. A equipa decide qual é a forma mais apropriada a ser aplicada.

As *tarefas* no *OpenUp* são claras e centradas nos resultados. O texto é geralmente breve e objetivo, descrevendo como os indivíduos devem colaborar para alcançar as metas. Os passos são descrições curtas do que se deve alcançar, e apontam para orientações externas sobre a forma como o devem fazer, o qual recomenda a menor quantidade de orientação de processos que uma equipa deve utilizar para ser bem-sucedida. As equipas podem ter nomes ou responsabilidades diferentes para os papéis, bem como realizar *tarefas* ou ter representações diferentes para os artefactos, mas ainda assim querem seguir os princípios e práticas descritos no *OpenUp* para aumentar as hipóteses de sucesso do projeto.

Completo

Pode ser manifestado como um processo completo de construção de um sistema, porque cobre as disciplinas essenciais de um ciclo de vida de desenvolvimento de *software*, orientando a equipa nas seguintes atividades de alto nível:

- As necessidades e requisitos dos clientes são informados e capturados com o envolvimento contínuo dos mesmos;
- É desenvolvida uma arquitetura robusta para o sistema, abordando os riscos técnicos e promovendo a organização da equipa;
- Para cada requisito é concebida, implementada e testada uma solução técnica, em conformidade com as decisões arquitetónicas;

- O sistema é avaliado por testes que validam os requisitos do cliente;
- É dada prioridade ao trabalho, as iterações são planeadas e avaliadas, e os membros da equipa assumem o trabalho a ser feito.

Contudo, este assume que a equipa de projeto não é responsável por certas atividades e decisões que são atribuídas a outras áreas da organização, como por exemplo:

- A criação do caso empresarial é tratada pela direção que decide se vale ou não a pena investir no projeto e qual é o retorno do investimento;
- Configuração do ambiente - algumas questões organizacionais podem não estar no âmbito da equipa, tais como: instalação, configuração e licenciamento de ferramentas de desenvolvimento e ferramentas de gestão de configuração; personalização e publicação do processo de desenvolvimento;
- Implementação e operação - Abordadas a nível organizacional, em vez de a nível de equipa.

Extensível

Pode ser usado como base sobre a qual o conteúdo do processo pode ser adicionado ou adaptado conforme necessário. Pode ser adotado por projetos tal como está, *out-of-the-box*, porque é mínimo e completo. No entanto, projetos diferentes podem ter necessidades diferentes. As organizações podem estar interessadas em personalizar alguns aspetos do *OpenUp* para o adaptar aos seus projetos. Estes são alguns exemplos de uma possível personalização:

- Adicionar novos papéis ou renomear os existentes;
- Adicionar etapas às *tarefas* existentes;
- Criar diretrizes sobre uma dada técnica;
- Remover uma determinada área de conteúdo;
- Modificar os modelos existentes ou adicionar novos modelos aos artefactos;
- Modificar ou criar um ciclo de vida do processo;
- Adicionar ou remover o conteúdo do processo;
- Acrescentar orientações sobre como alcançar a conformidade;
- Adicionar orientação específica a uma tecnologia a ser utilizada;

- Substituir ou aumentar uma das camadas com novo conteúdo (por exemplo, alterando ou acrescentando material à forma como a gestão é efetuada).

Ciclo de Vida do Projeto

Assim como o Rational Unified Process (RUP), o *OpenUp* baseia-se num ciclo iterativo e incremental, o qual se foca em criar um modelo ágil e preciso. Para além disso possui iterações e micro incrementos, em que este micro incremento é uma pequena unidade de trabalho e o conjunto de atividades que são executadas numa interação [23].

Sendo que o seu funcionamento completo pode ser visualizado na Figura 8:

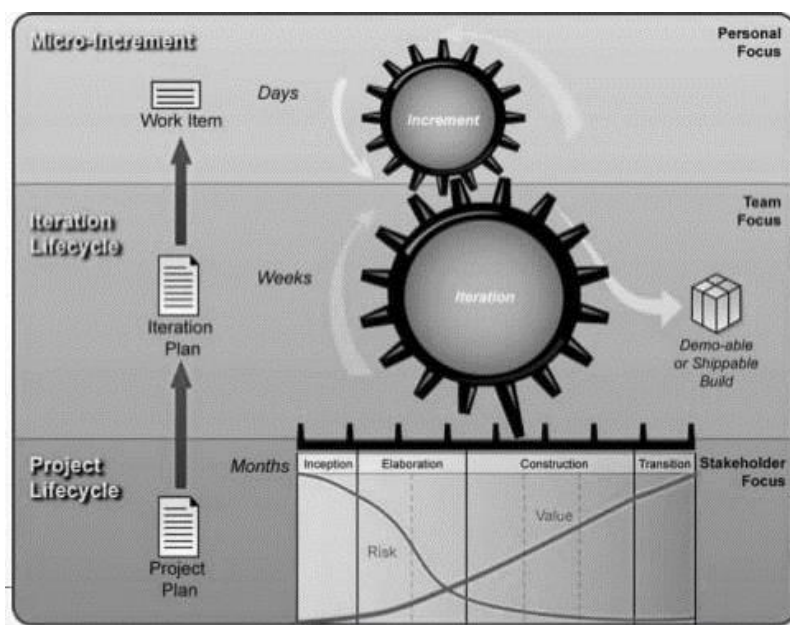


Figura 8 - Ciclo de Vida OpenUp [23]

Durante as quatro fases é necessário que alguns artefactos sejam respondidos e completos, a não existência do mesmo ou que este esteja incompleto, significa que o projeto pode atrasar ou ter problemas. De seguida, serão apresentadas as quatro fases [26]:

- **Inception (Fase Conceção):** neste ponto todos devem concordar com o detalhe de todo o trabalho necessário para entregar o produto final, dentro das expectativas do cliente e objetivos finais do projeto.

- **Elaboration (Fase de elaboração):** na elaboração é necessário que haja uma arquitetura concreta e funcional, além de ter riscos mitigados e possuir valores aceitáveis para a entrega final.
- **Construction (Fase de construção):** neste ponto é realizado o desenvolvimento do sistema, que permitirá no final efetuar entregas incrementais onde demonstra valores funcionais ao *stakeholder*.
- **Transition (Fase de transição):** se a aplicação conseguir sair da fase de construção é necessário que ela seja totalmente funcional e sem problemas para ser colocada em produção.

O ciclo de vida do projeto contempla um determinado período, e todas as datas de atualização e fases a serem seguidas são declaradas num artefacto designado de Plano de Projeto. O ciclo de vida do projeto é baseado em quatro fases (Figura 9), dependendo em qual das quatro fases o projeto se encontra, é possível e necessário tomar diferentes ações de acordo com o risco identificado. De acordo com L. Diedrich [25], o ciclo de vida do projeto fornece uma visão geral com transparência e com mecanismos para gerir os custos do projeto, riscos, valores, e outros aspetos do processo.

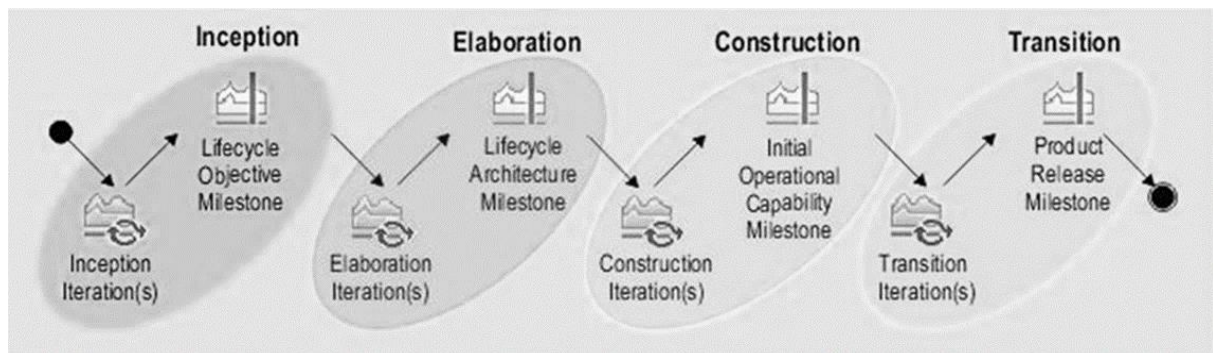


Figura 9 - Quatro fases do ciclo de vida OpenUp [25]

Ciclo da Iteração

Segundo L. Diedrich [25], as iterações do *OpenUp* permitem é manter a equipa focada na entrega de algo de valor para o cliente a cada micro- incremento, durante as quais é lhe entregue um protótipo funcional da aplicação. Deste modo, todas as semanas é possível detetar erros ou requisitos mal levantados. O plano da iteração, o tempo gasto e progresso são centrados em produtos entregues, estes possuem prioridades e quanto mais alta a prioridade do caso de uso mais cedo deve entrar num plano de iteração [25].

Micro Incrementos

Os micro incrementos são pequenas unidades de trabalho de um ou mais membros da equipa. As unidades de trabalho podem representar algumas horas a alguns dias de trabalho, em algum produto de acordo com o pré-estabelecido no plano de iteração. Este prevê um retorno extremamente rápido dos envolvidos no projeto o que torna possível tomar pequenas decisões que podem corrigir erros.

De acordo com L. Diedrich [25], o conceito de um micro incremento ajuda ao membro da equipa a repartir o seu trabalho em unidades menores, as quais, quando terminadas, este entrega uma pequena quantidade de valor. É importante que a cada micro incremento terminado tenha um novo artigo de trabalho entregue e que na conclusão, sejam adicionados os resultados numa lista comum para que todos os membros da equipa vejam o resultado da iteração com transparência.

2.5 Base de dados por Grafos

Para representar dados, é bastante utilizada uma teoria que estuda as relações dos objetos de um determinado conjunto, conhecida por grafos, sendo esta fundamental em ciências da computação e umas das estruturas mais úteis para modelar objetos e iterações [27].

De forma mais simplista, quando falamos em grafos G este par (V, E) , onde V representa um conjunto finito de “nós” e a letra E representa um conjunto finito de arestas que conectam pares de “nós”. Naturalmente, as arestas podem ser direcionadas ou não, embora consideremos apenas os casos direcionados que são os mais gerais [28].

Por volta dos anos 90, os dados semiestruturados e as bases de dados de objetos obtiveram um especial destaque, pois forneceram áreas frutíferas para o estudo de modelos de grafos e linguagem de consulta. No entanto, na última década, as redes semânticas da web e sociais, começaram assumir-se como possíveis áreas-chave das abordagens baseadas em grafos [28].

Contudo, o modelo existente de base de dados relacional, que surgiu no final dos anos 60, provou fornecer mecanismos de persistência, controlo de concorrência e integração. Os sistemas de

gestão de base de dados relacional (RDMS) mantém tabelas definidas por conjuntos de linhas e colunas, nas quais uma linha pode ser percebida como um objeto, enquanto as colunas seriam atributos/propriedades desses objetos [29]. Todavia, uma das fraquezas do modelo relacional é a sua capacidade limitada de capturar explicitamente a semântica de requisitos.

O armazenamento, a recuperação e manipulação de dados complexos tornaram-se dispendiosos quando, são utilizadas abordagens RDMS tradicionais, com esses modelos de dados baseados em esquemas, por conta própria. Para tal, isto implica que haja um processo manual para redesenhar o esquema para assim adaptar-se a novos dados, onde o RDMS é otimizado para dados agregados, as bases de dados por grafos são otimizadas para dados altamente conectados. Um grafo é uma estrutura de dados composta por arestas e vértices [30], onde uma tecnologia de base de dados por grafos se destaca como uma ferramenta eficaz para um projeto de um modelo de dados. Através de grafos, podemos representar diversos objetos e relacionamentos entre si, sendo um tipo de grafo comum suportado pela maior parte dos sistemas é um grafo de propriedades, que pode ser atribuído, classificado e direcionado para sistemas de múltiplos grafos [31].

Antes do ano de 2002, Angles e Gutierrez desenvolveram uma pesquisa sobre modelos de dados por grafos. Posteriormente, os autores sintetizaram a noção de “modelo de base de dados de grafos” e compararam as propostas disponíveis até à data [32].

É importante salientar que a maioria dos estudos revistos pelos autores seguiu, acima de tudo, um interesse teórico e não tanto num desenvolvimento prático. Com relação aos desenvolvimentos recentes na área, foi realizada uma pesquisa com o objetivo de analisar e comparar as principais bases de dados por grafos existentes. Com base num *ranking* de 31 sistemas até setembro de 2018, os 3 principais identificados são o *Neo4j* com 40.10%, em segundo lugar o *Microsoft Azure Cosmos DB* com 19,18% e por último o *Datastax Enterprise* com 7.76%, em que estes são mais utilizados entre os sistemas de gestão de base de dados estudado [33]. Como podemos observar, existe uma diferença significativa entre o primeiro e o segundo lugar no *ranking*. O *Neo4J* ocupa o primeiro lugar com uma diferença de 20,92%, muito mais alto que o DBMS que ocupa a segunda posição, o *Microsoft Azure Cosmos DB*. Embora a solução da *Azure* tenha características muito interessantes, ela também apresenta limitações significativas. É um DBMS que para suportar construções de grafos necessita de um API, o *Gremlin*, portanto não é considerada uma tecnologia de base de dados por grafos nativa. Para

além disto, a *Azure* não disponibiliza uma versão *open source*. Por último, o paradigma *Neo4j* é umas das maneiras de representar a visualização gráfica das organizações e contextos específicos [33].

Neo4J vs Cosmos BD Azure

Para o cruzamento entre os dois *frameworks* anteriormente mencionados (*OpenUp e CMMI*), o estudo poderá ser realizado tendo como opção duas tecnologias de base de dados por grafos bem cotadas, no entanto, bem distintas entre si. Relativamente, ao *Neo4j* é descrito como "*The world's leading Graph Database*", sendo uma ferramenta que armazena os seus dados em nós conectados através de relacionamentos direcionais e ainda possuem propriedades. É um armazenamento de grafos de alto desempenho com inúmeros recursos e com uma plataforma robusta.

No caso do *Neo4J* é uma base de dados por grafos nativa o que facilita a criação de toda a componente semântica. A linguagem utilizada pelo *Neo4J* é o *Cypher Query Language (Cypher)*. Esta é uma linguagem de consulta de grafos do *Neo4j* que permite aos utilizadores armazenar e recuperar dados da base de dados por grafos. A sintaxe do *Cypher* apresenta uma forma visual e lógica de corresponder aos padrões de nós e relacionamentos nos grafos. É uma linguagem declarativa e inspirada em SQL para descrever os padrões visuais dos grafos utilizando a sintaxe ASCII-Art. É possível indicar o que queremos selecionar, inserir, atualizar ou excluir dados do grafo sem uma descrição exata de como fazê-lo. Os utilizadores podem criar consultas expressivas e eficientes para lidar com a criação, leitura, atualização e exclusão das funcionalidades necessárias.

Em contrário, temos o *Azure Cosmos DB* descrito como sendo "*A fully-managed, globally distributed NoSQL database service*". Esta tecnologia é, no entanto, um serviço de base de dados *NoSQL* que pode ser totalmente gerido e permite também um desempenho rápido e previsível.

Relativamente ao *Azure Cosmos DB*, como é uma base de dados *NoSQL*, é necessário a utilização da API do *Gremlin* para que possa representar modelações por grafos. O *Gremlin* é a linguagem do *Apache TinkerPop* e permite aos utilizadores expressar sucintamente percursos complexos (ou consultas) dos grafos de propriedades das suas aplicações. Todo percurso do *Gremlin* é composto por uma sequência de etapas. Uma etapa executa uma operação atômica no fluxo de dados. Cada etapa é

uma etapa do mapa (transformando os objetos no fluxo), uma etapa do filtro (remoção de objetos do fluxo) ou uma etapa *sideEffect* (estatísticas de computação sobre o fluxo).

De seguida, será explicado o motivo de optar entre uma das tecnologias acima mencionadas.

Análise e escolha da tecnologia Neo4J

De acordo com a análise das duas ferramentas de bases de dados por grafos, a solução *Neo4j* foi a selecionada devido à sua posição no *ranking* ser consistente ao longo dos anos, e ao facto de ser uma tecnologia *OpenSource* e à extensa comunidade de trabalho existente.

A seguir, os conceitos de bases de dados por grafos subjacentes à solução *Neo4j* são apresentados com o intuito de dar a conhecer a forma como um grafo é composto e de que modo as operações são realizadas.

Tecnologia Neo4J

Neo4j é uma base de dados de código aberto baseado no conceito *Nosql*. Aqui, a informação (dados) não é armazenada em tabelas, mas na forma de grafos e as suas estruturas são representadas de modo que o conhecimento seja representado por conceitos matemáticos da Teoria dos Grafos. Esta é uma estrutura simples e robusta para a construção de modelos ou para a construção de sistemas complexos que podem variar do mapeamento de processos industriais à logística, sistemas de comunicação, fluxo de rede, seleção de rotas, entre muitas outras abordagens [34]. Um grafo é um objeto que contém nodos e relacionamentos. Os nodos são compostos por propriedades e são organizados por relações que também têm suas propriedades específicas. Ao navegar pelos grafos, os caminhos que ordenam os nodos são identificados. A Figura 10 ilustra os componentes individuais de um grafo e dá uma representação visual de como os nodos se relacionam uns com os outros [31].

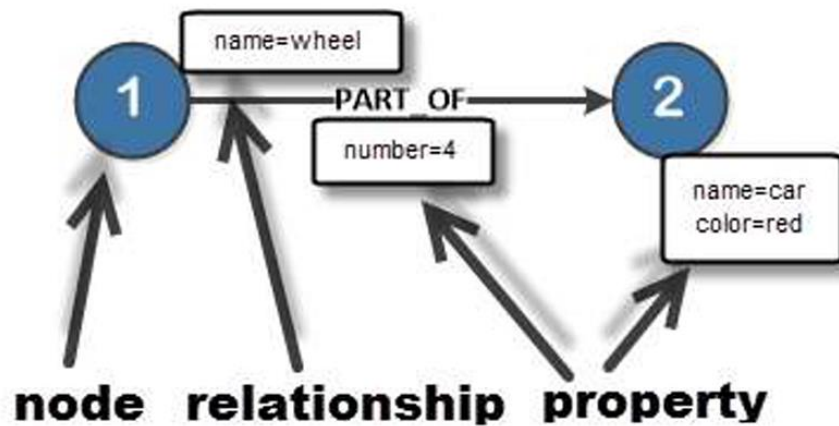


Figura 10 - Core Graph Concepts [31]

A representação por grafos fornece uma visão mais dinâmica das dependências e das relações entre os diferentes processos, subprocessos, atividades e artefactos. O uso desta ferramenta de representação permite uma análise eficiente de vários aspetos, a saber, as relações entre processos, uma vez que permite traçar um curso inteiro, com o intuito de verificar se o processo modelado está a ser efetivamente executado.

Esta abordagem permite-nos validar se poderia ser mais fácil navegar no processo, modelado em *BPMN*, selecionando os artefactos que as equipas de desenvolvimento da empresa necessitam de criar e utilizar. De referir ainda que também permite acompanhar e comparar as práticas que realmente acontecem atualmente, também pode agir na direção oposta, isto é, selecionar atividades e processos e assim saber quais artefactos devem ser gerados.

Assim sendo, o processo organizacional, que vem da modelação *BPMN* já validada pela empresa, poderá ser mapeado em *Neo4j*, alimentado esse mapeamento com as informações relevantes quer de *OpenUp* como do *CMMI*.

2.6 Conclusão

Neste capítulo podemos concluir que, na área da engenharia de software, é de extrema importância que haja uma boa análise dos requisitos para que os processos de desenvolvimento do software sejam realizados com o maior rigor possível. Para isso é preciso uma análise e levantamento de necessidades, com o objetivo de obter os resultados desejados.

No capítulo seguinte, são apresentadas algumas modificações que consideramos importantes para que haja uma melhoria contínua na empresa. Relativamente aos *frameworks* apresentados, concluímos que estes possuem boas ferramentas e procedimentos que servem de suporte para os processos de desenvolvimento presentes na ITEC. Para realizar o mapeamento destes *frameworks* é utilizada a ferramenta *Neo4j* para a construção da nossa base de dados por grafos.

Estes dois *frameworks* são importantes, pois o CMMI ajuda a empresa a aprender o que fazer em cada procedimento e com a ajuda do *OpenUp* permite-nos perceber como devemos proceder para executar as *tarefas* com a melhor qualidade possível.

3. CARACTERIZAÇÃO DO PROBLEMA PROCESSUAL DA ITEC

3.1 Introdução

Neste capítulo é realizada a contextualização da ITEC, com o objetivo principal de apresentar as melhorias pretendidas. Tendo em consideração toda esta informação é elaborado um BPMN, onde se tem em consideração todos os macroprocessos presentes na empresa. Depois de algumas reuniões realizadas na ITEC, são construídos os casos de uso, onde são apresentadas as suas atividades e intervenientes (atores), à qual chamamos de casos de uso iniciais. De seguida foram realizadas mais reuniões e entrevistas estruturadas, com o objetivo de retirar o máximo de informação possível e de apresentar sugestões de melhoria, que chamamos de casos de uso refinados. Assim sendo, depois de fechados os casos de uso refinados, estes são utilizados para a elaboração dos requisitos de processos, que são devidamente descritos e explicados.

Para uma caracterização detalhada de todas as tarefas e processos presentes na empresa recorreremos a dois tipos de mapeamentos distintos: o mapeamento dos macroprocessos e dos casos de uso. No que toca aos macroprocessos, utilizamos o BPMN para mapear de forma detalhada as interações entre cada uma das áreas internas da empresa e a forma de trabalhar em cada uma delas. Já com o mapeamento dos casos de uso conseguimos especificar ainda mais cada uma das referidas áreas presentes nos BPMN, identificando os atores (colaboradores) que executam as tarefas inerentes às áreas, nomeadamente, no que diz respeito às atividades de gestão de projetos, desenvolvimento de *software*, controlo de qualidade e serviço pós-venda. Desta forma é-nos possível ter uma visão mais ampla de todo contexto organizacional da empresa em estudo.

3.2 Modelação com BPMN

Como referido na contextualização, com o propósito de se entender melhor os processos de negócio e do departamento de desenvolvimento de *software*, foi realizado um BPMN, respetivamente demonstrado na Figura 11. O processo de desenvolvimento de *software* inicia-se no departamento comercial com a emissão da ordem de início do projeto que é passada ao gestor de projetos (GP) que, por sua vez, inicia o planeamento do projeto. Dentro do planeamento resultam dois documentos: o

Caderno de Encargos e o Planeamento Temporal do Novo Projeto. Este último documento é enviado para o responsável de equipa (RE), para que se inicie o processo de escolha do programador. Caso o programador seja escolhido a tempo, participará na reunião *Kickoff*. Caso contrário irá depois participar na reunião de planeamento designada internamente, na empresa, como “*Scrum Planning*”.

Na reunião *Kickoff* participam o gestor de projetos e os responsáveis de equipa, com a possibilidade da participação do programador, consoante o que já foi dito anteriormente. Nesta reunião, com base nos documentos “Planeamento Temporal do Projeto” e “Caderno de Encargos”, faz-se uma análise de modo a definir-se os parâmetros e objetivos funcionais que se pretende obter com a máquina, parâmetros estes que são depois inseridos no *OneNote* pelo Gestor de Projeto. À posteriori efetua-se uma reunião global de projetos que é realizada uma vez por semana (às segundas-feiras). Nesta reunião discute-se o ponto de situação, podendo fazer-se uma adaptação dos objetivos, tarefas e prazos dos vários projetos, cujos parâmetros são novamente inseridos no *OneNote*. Daqui, e com base nas tarefas a realizar, o RE faz o planeamento das tarefas e a subdivisão das mesmas, inserindo o seu planeamento no *Jira/Trello* de forma detalhada. No entanto, às terças-feiras, realiza-se a “Reunião *Scrum Planning*”, sendo que por vez esta reunião acontece no início do projeto. Nesta reunião participam o RE juntamente com os programadores, com o objetivo de se analisar o planeamento e a alocação das tarefas. Esta atualização (informação com as pessoas alocadas) é inserida no *Jira/Trello*. O programador, após ter acesso ao caderno de encargos e às tarefas, faz uma análise deste caderno, esquematização do *Software* (SW) de controlo e desse modo inicia a programação da máquina. Ao longo desta fase, o programador faz diariamente o registo da taxa de realização, que é inserido no documento de monitorização e desempenho. Caso haja atraso da tarefa, o programador tem de registar nos comentários do *Jira/Trello* o motivo do incumprimento das tarefas. Ao mesmo tempo, o programador tem de realizar uma atualização diária do código para o sistema de controlo de versões (SVN).

Semanalmente, o programador escreve o seu *feedback* e/ou levantamento de dificuldades encontradas. Este *feedback* serve de input para a reunião semanal “*Daily Scrum*”, realizada às quintas-feiras, com o intuito de se analisar e resolver as questões levantadas. Depois, o RE prepara um documento com um conjunto de orientações sobre como o programador deve validar o SW e os testes elétricos e mecânicos, procedendo o programador com esta informação à validação do SW e à realização dos testes. Caso sejam detetados erros, ele deve comunicar a avaria ao RE com conhecimento do GP, fazer o retrabalho/revisão e uma nova validação do mesmo. Após a validação do programador, o RE faz

a validação das funcionalidades da máquina e, não havendo erros a apontar, o programador inicia a instalação no cliente final. Caso o cliente faça um pedido de suporte/reclamação (pedido esse que é inserido no documento Monitorização e Desempenho), o programador tem de avançar com a assistência criando uma solução. Se a solicitação não resolver o problema faz-se o retrabalho da mesma até ficar resolvido. Após estar resolvido o pedido é dado como concluído, sendo também registado no documento de Monitorização e Desempenho.

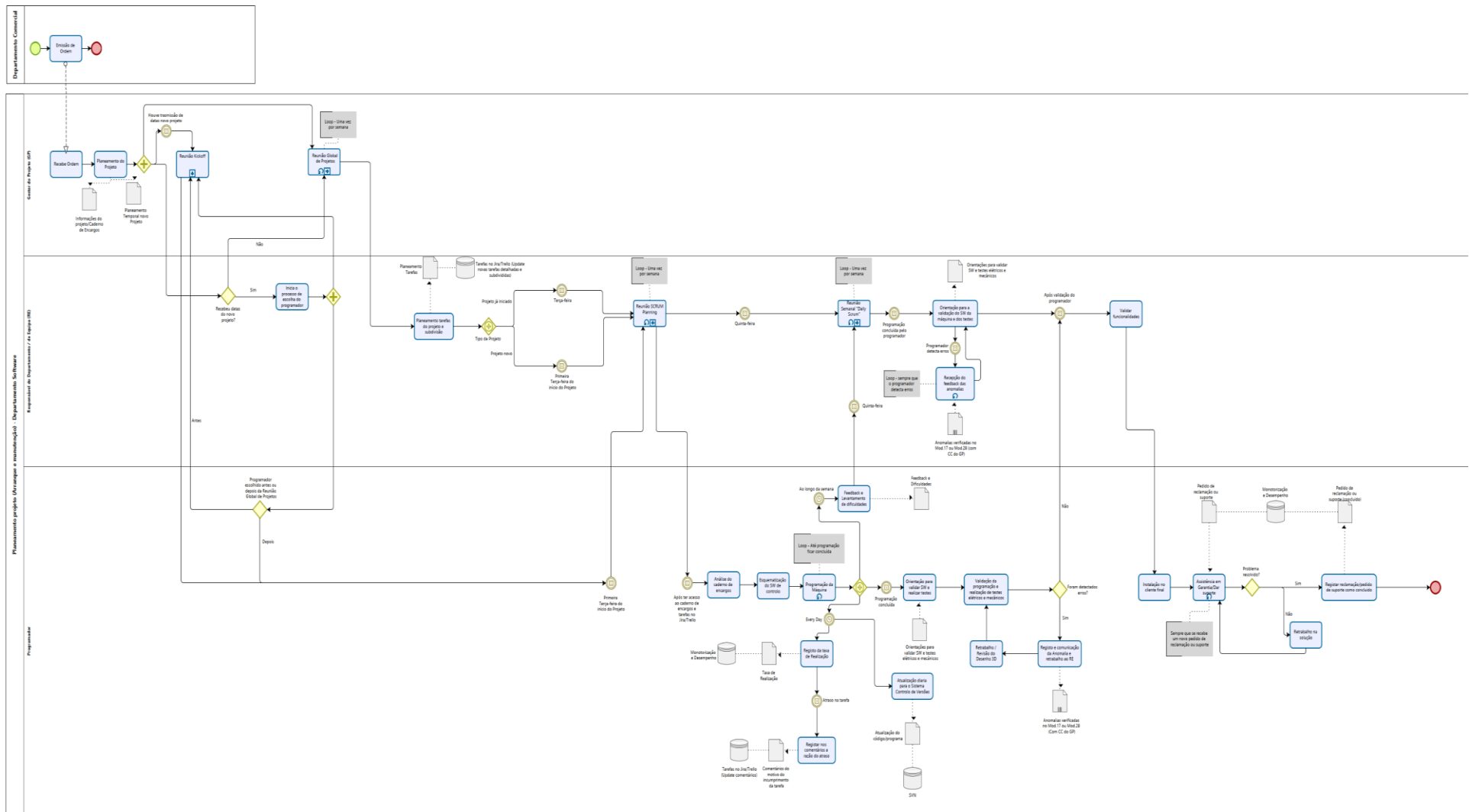


Figura 11 - Modelação BPMN dos processos da ITEC

3.3 Modelação de Casos de Uso

De forma a representar o estado atual da empresa, recorreremos aos casos de uso aos quais chamamos de casos de uso iniciais. Contudo, foram implementadas melhorias nos processos da empresa, que são apresentados nos casos de uso refinados, que servem como suporte para a criação dos requisitos de processos.

Casos de uso iniciais

A modelação dos casos de uso iniciais, tem particular importância ao nível do estado atual da empresa, uma vez que nos permite ter uma ideia concreta acerca das áreas da empresa, assim como as relações entre os vários atores e as suas funções e tarefas. A estrutura destes modelos é baseada na definição de atividades (casos de uso), que são colocadas dentro da caixa central, sendo depois ligadas aos vários atores que nelas participam [35]. Na Figura 12 é possível observar-se o panorama geral, nomeadamente os “ {UC0} – Processos de desenvolvimento ITEC”, sendo constituído pelo “{UC1} - Gestão de Projetos”, “{UC2} – Desenvolvimento”, “{UC3} - Controlo de Qualidade/Testes” e “{UC4} - Serviços Pós-venda”, e os atores que atuam em cada um deles. Os casos de uso apresentados correspondem às principais áreas de interesse respeitantes ao Departamento de Software, departamento no qual o projeto está focado.

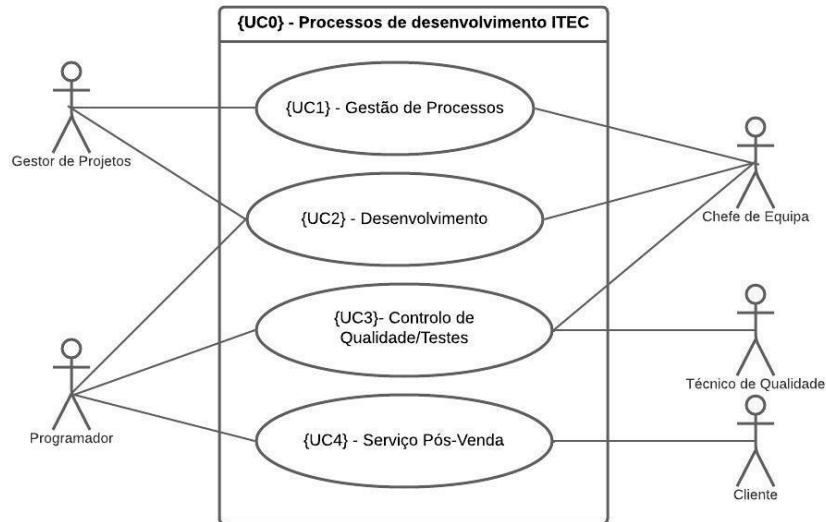


Figura 12 - {UC0} – Processos de Desenvolvimento ITEC

No que toca ao “{UC1} -Gestão de Projetos”, este é composto pelas atividades “{UC1.1} -Planear principais prazos e datas do projeto”, “{UC1.2} - Analisar o caderno de encargos (1ª análise)”, “{UC1.3} - Organizar reunião *kickoff*” e “{UC1.4} - Reunião global de projetos”. Estas atividades são divididas entre o Gestor de Projetos e o Chefe de Equipa, que correspondem aos intervenientes associados nesta área. Na Figura 13, pode ser observado uma visão global deste caso de uso.

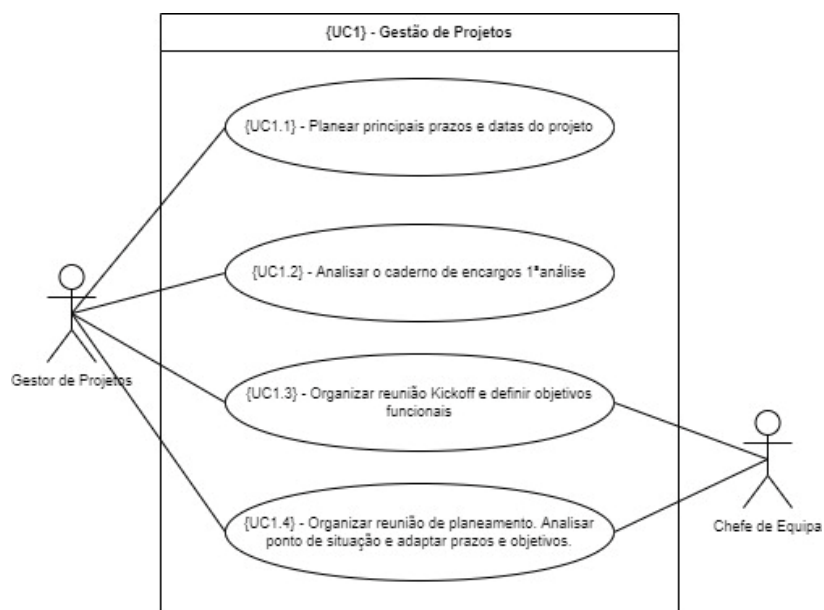


Figura 13 – {UC1} – Gestão de Projetos

Relativamente ao “{UC2} – Desenvolvimento”, este caso de uso aborda a temática do desenvolvimento ao longo do projeto e está dividido nas atividades “{UC2.1} - Planejar tarefas e escolher programadores”, “{UC2.2} - Reunir semanalmente e alocar tarefas”, “{UC2.3} - Analisar o caderno de encargos”, “{UC2.4} - Planejar e esquematizar o software” e “{UC2.5} - Programar máquinas”. Esta fase é dividida entre dois intervenientes, nomeadamente, o chefe de equipa e o programador, como se pode observar na Figura 14.

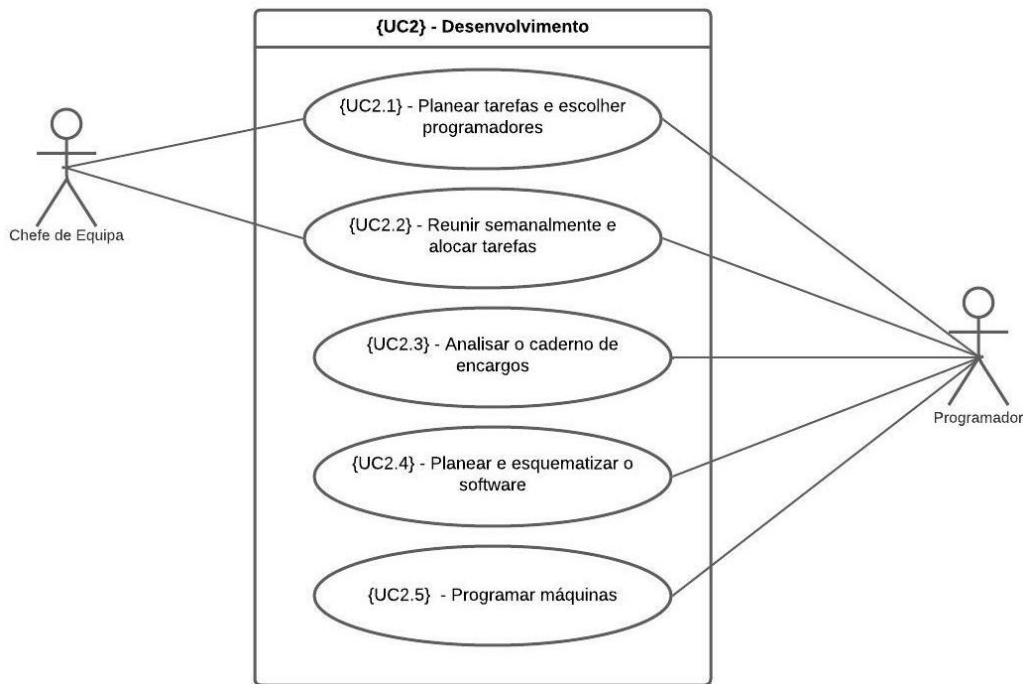


Figura 14 – {UC2} - Desenvolvimento

Seguindo agora, para o caso de uso “{UC3} - Controlo de Qualidade/Testes”, que pode ser visualizado na Figura 15, este tem como intervenientes o Programador, o Chefe de Equipa e o Técnico da Qualidade. As atividades desta área da empresa passam por “{UC3.1} - Atualização diária do controlo de versões”, “{UC3.2} - Registo das taxas de realização e justificar atrasos” e “{UC3.3} – Registrar dificuldades”, “{UC3.4} - Organização da reunião semanal de dificuldades”, “{UC3.5} – “ Preparar Testes”, “{UC3.6} – Realizar Testes”, “{UC3.7} – Registrar e Visualizar anomalias”, “{UC3.8} – Validar funcionalidades e testes (antes da fase final)”, “{UC3.9} – Validar testes (fase final)”, “{UC3.10} – Retrabalho/Rever desenho 3D” e por último temos “{UC3.11} – Realizar uma validação intermédia da máquina”.

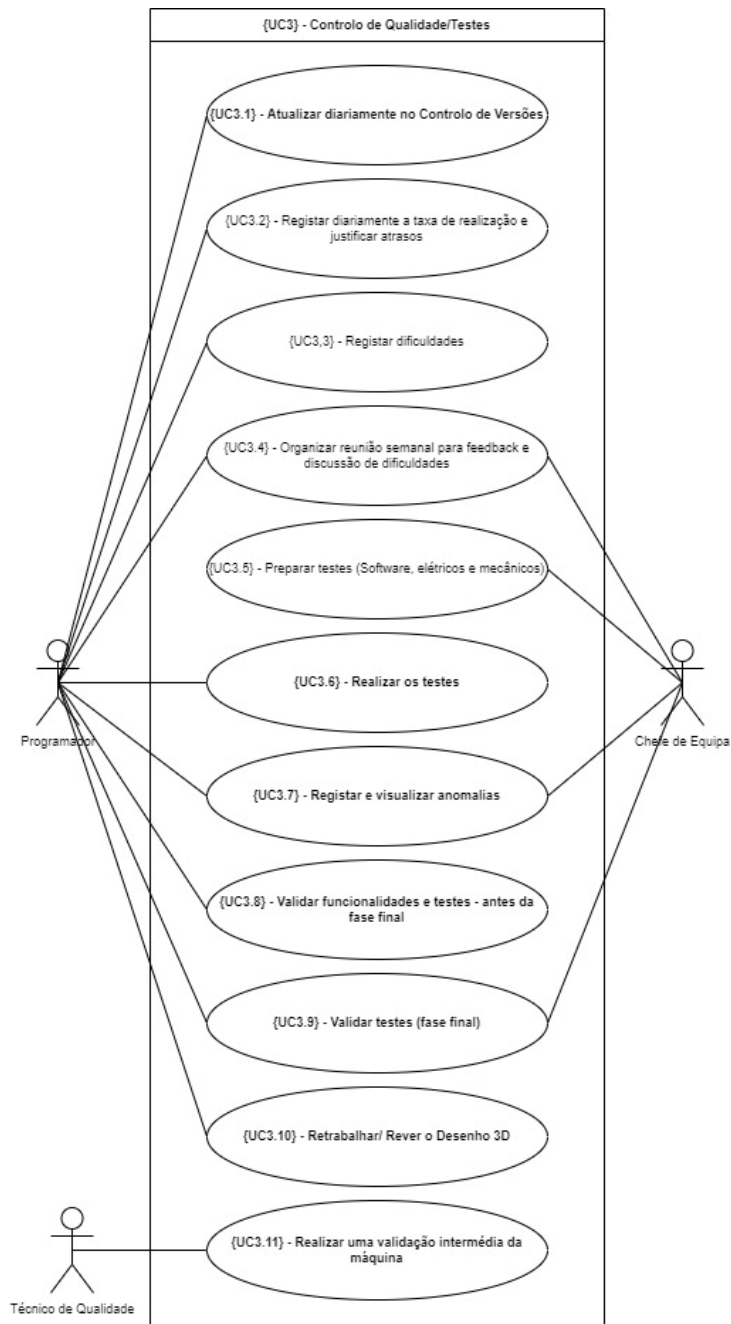


Figura 15 - {UC3} - Controle de Qualidade/Testes

Em relação ao “{UC4} -Serviços Pós-venda”, Figura 16, este caso de uso tem como principais atores o Gestor de Projetos, o Programador/Assistente Técnico e o Cliente, que têm associado as atividades “{UC4.1} - Registrar e gerir pedidos de assistência em garantia”, “{UC4.2} - Alocar o pedido a um programador”, “{UC4.3} - Instalar a máquina no cliente final” e “{UC4.4} - Desenvolver nova solução após pedido de assistência”.

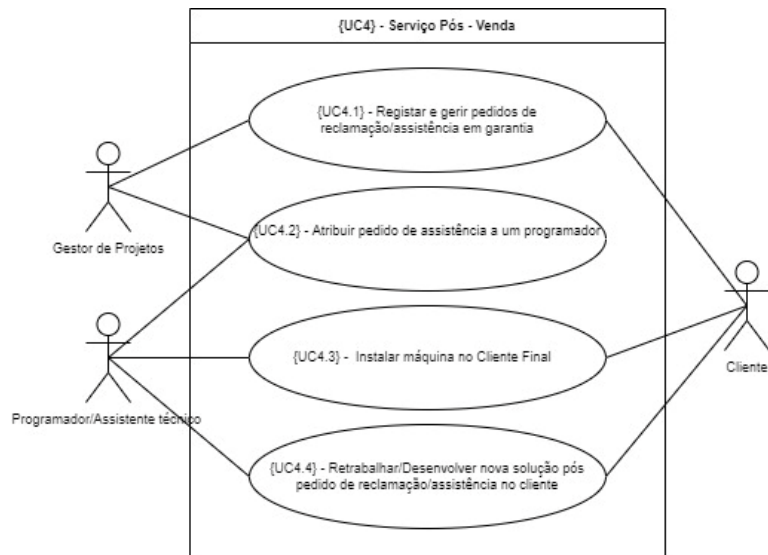


Figura 16 - {UC4} - Serviço Pós-Venda

Depois de levantados todos os casos de uso do estado atual da empresa, é elaborado um plano de reestruturação dos mesmos e feito a implementação de novos para melhorar os procedimentos e gestão da empresa, com o objetivo de obter melhores resultados. Por este motivo surgem os casos de uso refinados, que são apresentados de seguida.

Casos de uso melhorados

Este tipo de modelação (casos de uso melhorados) é de extrema importância, pois permite comparar o estado atual da empresa com as propostas de melhoria por nós efetuadas. Na Figura 17, podemos observar o caso de uso “{UC0} - Processos de desenvolvimento ITEC melhorados” referente à empresa, onde temos um misto de azul e verde são os casos de uso que se mantêm inalterados e a cor verde temos o caso de uso que surge como sugestão de melhoria. Os casos de uso apresentados correspondem às principais áreas e tarefas desempenhadas no Departamento de software, sendo esta a principal área em que o projeto incide.

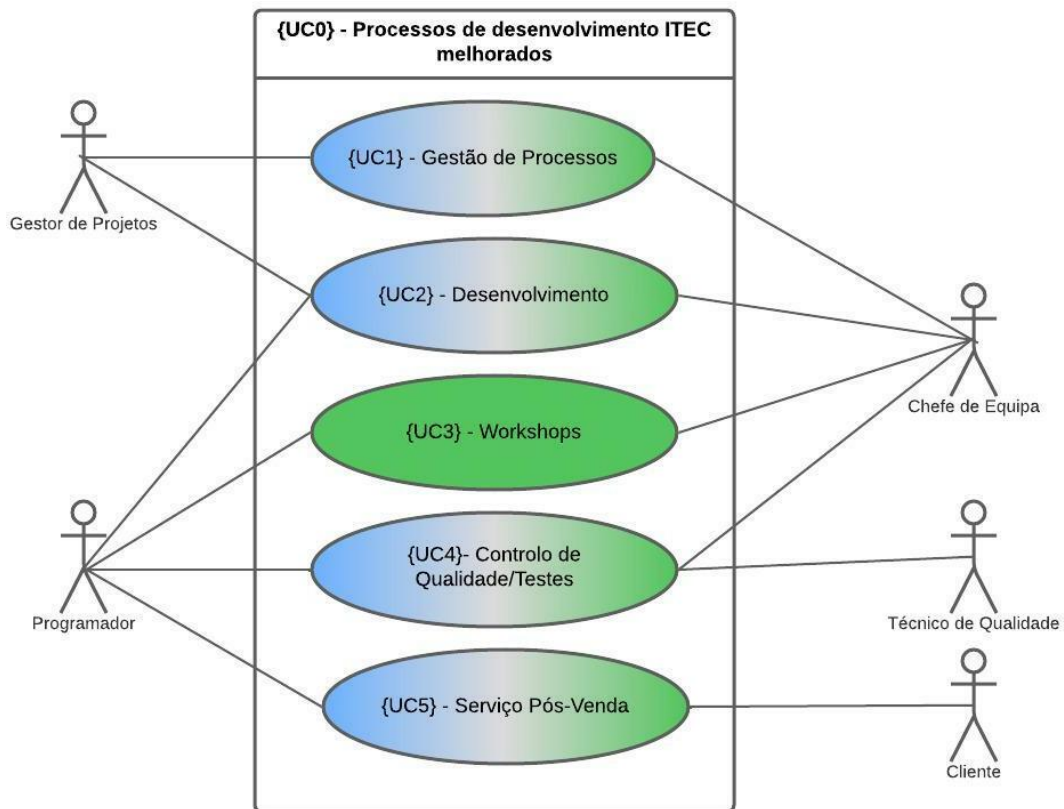


Figura 17 - {UC0} – Processos de Desenvolvimento ITEC melhorados

Na Tabela 1, são apresentados os casos de uso iniciais já descritos anteriormente e os casos de uso refinados que serão seguidamente descritos. Anteriormente, tinham sido efetuadas as primeiras modelações de casos de uso relativas ao seu contexto atual, onde de seguida serão apresentadas as alterações que achamos que fazem sentido serem incrementadas com o objetivo de haver uma melhoria.

Com a observação comparativa da tabela podemos ter uma melhor noção do que as melhorias propostas incrementam no desenvolvimento de software.

	Casos de Uso iniciais	Casos de Uso melhorados
{UC1} Gestão de Projetos	<ul style="list-style-type: none"> ○ Planear principais prazos e datas do projeto; ○ Analisar o caderno de encargos 1ª análise; ○ Organizar reunião Kickoff e definir objetivos funcionais; ○ Organizar reunião de planeamento. Analisar ponto de situação e adaptar prazos e objetivos. 	<ul style="list-style-type: none"> ○ Planear principais prazos e datas do projeto; ○ Analisar o caderno de encargos 1ª análise ○ Organizar reunião <i>Kickoff</i> e definir objetivos funcionais, considerando pesos e tempos mais realistas; ○ Organizar reunião de planeamento, feita projeto a projeto – analisar ponto de situação e adaptar prazos e objetivos.

<p>{UC2} Desenvolvimento</p>	<ul style="list-style-type: none"> ○ Planear tarefas e escolher programadores; ○ Reunir semanalmente e alocar tarefas; ○ Analisar o caderno de encargos; ○ Planear e esquematizar o software; ○ Programar máquinas. 	<ul style="list-style-type: none"> ○ Planear tarefas (atribuindo um peso/valor á dificuldade da mesma e escolha de programadores); ○ Ler e considerar o calendário de todos os membros antes das reuniões; ○ Reunir semanalmente (<i>Scrum</i>) e alocar tarefas, projeto a projeto em separado; ○ Analisar o caderno de encargos de forma normalizada (em tempo); ○ Planear e esquematizar o software realizado de forma normalizada (em método e em tempo); ○ Considerar o peso das tarefas extra, em termos de dificuldade e tempo, com possibilidade de rejeição; ○ Programar máquinas.
<p>{UC3} Workshops</p>	<p>-</p>	<ul style="list-style-type: none"> ○ Planear e preparar workshops de novas linguagens/máquinas e de técnicas de gestão de forma cíclica ○ Participar nos workshops ○ Realizar exame e obter certificado
<p>{UC4} Controlo de Qualidade/Testes</p>	<ul style="list-style-type: none"> ○ Atualizar diariamente no Controlo de Versões; ○ Registar diariamente a taxa de realização e justificar atrasos; ○ Registar dificuldades; ○ Organizar reunião semanal para feedback e discussão de dificuldades; ○ Preparar testes (Software, elétricos e mecânicos); ○ Realizar os testes; ○ Registar e visualizar anomalias; ○ Validar funcionalidades e testes – antes da fase final; ○ Validar testes (fase final); ○ Retrabalhar/ Rever o desenho 3D; ○ Realizar uma validação intermédia da máquina. 	<ul style="list-style-type: none"> ○ Atualizar diariamente no Controlo de Versões; ○ Registar diariamente a taxa de realização e justificar atrasos; ○ Registar dificuldades; ○ Requisitar, convidar e realizar reunião breve para feedback e discussão de dificuldades; ○ Preparar testes (Software, elétricos e mecânicos); ○ Realizar os testes; ○ Registar e visualizar anomalias; ○ Validar funcionalidades e testes – antes da fase final; ○ Validar testes (fase final); ○ Retrabalhar/ Rever o desenho 3D; ○ Escrita regular, continua e normalizada de comentários no código ○ Definir e escrever (com atualizações cíclicas) o guia de boas praticas na codificação, para efeitos de se atingir standard ○ Validar de forma continua e cíclica do código
<p>{UC5} Serviço pós-venda</p>	<ul style="list-style-type: none"> ○ Registar e gerir pedidos de reclamação/assistência em garantia; ○ Atribuir pedido de assistência a um programador; ○ Instalar máquinas no Cliente Final; ○ Retrabalhar/Desenvolver nova solução pós pedido de reclamação/assistência no cliente. 	<ul style="list-style-type: none"> ○ Recolher feedback dos clientes de forma proativa; ○ Registar e gerir pedidos de reclamação/assistência em garantia; ○ Atribuir pedido de assistência a um programador; ○ Instalar máquinas no Cliente Final; ○ Retrabalhar/Desenvolver nova solução pós pedido de reclamação/assistência no cliente.

Tabela 1 - Comparação entre os casos de uso iniciais e os casos de uso melhorados

Relativamente ao “{UC1} - Gestão de Projetos”, passamos a apresentar os casos de iniciais coloridos a azul e os que representam as propostas de alteração ou implementação de procedimentos por parte da empresa coloridos a verde, na qual chamamos de casos de uso refinados. Podemos observar na Figura 18, que existem dois intervenientes nos processos da gestão de projetos: o gestor de projetos, que está envolvido em todas as atividades presentes, e o chefe de equipa, que também está presente em duas das atividades. Igualmente se pode observar a atividade relativa à gestão de projetos, onde se realiza o planeamento de prazos e datas do projeto, a análise inicial do caderno de encargos, a leitura e consideração do calendário de todos os membros antes das reuniões, a organização da reunião *Kickoff* e a definição dos objetivos funcionais, a consideração de pesos e tempos mais realistas e, por fim, a realização de uma reunião de planeamento feita projeto a projeto, onde se faz ponto de situação, adapta-se prazos e objetivos.

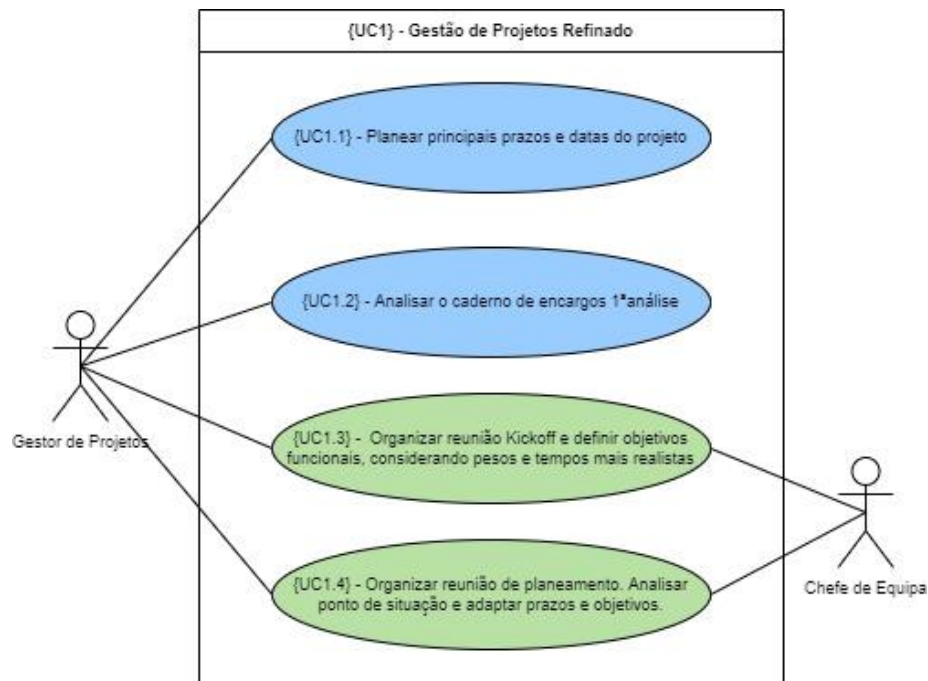


Figura 18 - {UC1} - Gestão de Projetos Refinado

O próximo caso de uso a considerar, especificamente o “{UC2} Desenvolvimento Refinado”, encontra-se dividido entre dois intervenientes, respetivamente, entre programador e o chefe de equipa. Esta fase inicia-se após a reunião de *Kickoff*, nomeadamente, após o chefe de Equipa ter acesso a todas as informações e objetivos funcionais. Com base nesta informação, o chefe de equipa escolhe os programadores de forma a compor a sua equipa de trabalho e planeia as tarefas a serem realizadas.

Estas tarefas são depois transmitidas aos programadores nas reuniões *Scrum* realizadas, por norma, às terças-feiras, com o intuito de serem alocadas pelos vários elementos da equipa. Na fase intermédia/final do projeto, caso o cliente efetue um novo pedido, a este será atribuído um peso, em termos de dificuldade e tempo. Com base neste peso, o chefe de equipa e a sua equipa decidem qual o melhor rumo a tomar em relação ao pedido, o qual pode ser rejeitado. No que diz respeito às restantes tarefas de {UC2}, estas estão a cargo do programador, que inicia o desenvolvimento com a análise do caderno de encargos, planeamento e esquematização do *software* de forma normalizada. Entende-se esta normalização pela imposição de um limite mínimo em termos de tempo para a realização das tarefas mencionadas, assim como um *standard* na realização da estrutura esquematizada. Após a realização destas tarefas, o programador inicia efetivamente a programação da máquina. No {UC2} temos várias atividades no seu desenvolvimento, como podemos observar pela Figura 19, como o planeamento das tarefas, considerar o calendário de todos os membros antes das reuniões, reunião semanal (*Scrum*) de alocação das tarefas, análise do caderno de encargos de forma normalizada (em tempo), fazer o planeamento e esquematização do software realizado de forma normalizada (em método e em tempo), considerar os pesos em termos de dificuldade e tempo no caso de haver novos pedidos durante a fase intermédia/final, onde há possibilidade de rejeição, e por fim é realizada a programação da máquina.

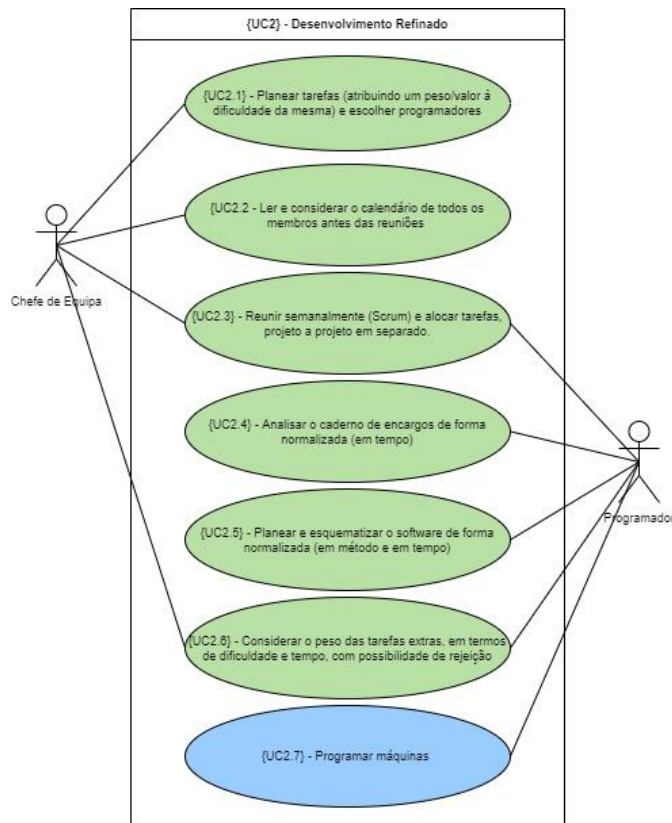


Figura 19 - {UC2} - Desenvolvimento Refinado

A proposta feita em {UC3} visa promover o incentivo à aprendizagem contínua sob a forma de *workshops*. Estes devem abordar tópicos técnicos ao nível de linguagens de programação e de vários tipos de robô e PLC, assim como treinos de processos de gestão e dar a conhecer técnicas *Agile (lean, Scrum, Kanban, etc)*. Os responsáveis pela promoção e preparação destes *workshops* são os chefes de equipa onde em conjunto com os membros da sua equipa, participam nos projetos e com isto obter eventuais certificados. Numa fase posterior, os próprios participantes iriam aplicar esses conhecimentos regularmente nos projetos, principalmente ao nível de técnicas de gestão. Ainda, e após estas fases iniciais estarem consolidadas, os membros com certificados passam a ser os responsáveis por dar os próximos *workshops* a membros novos ou a quem não teve ainda oportunidade de participar. Desta forma cria-se a consolidação contínua de toda a informação teórico-prática na empresa, ao passar conhecimento a novos participantes e ao testar a capacidade de ensino e aplicabilidade de quem já participou. Na Figura 20, está representada a atividade para o desenvolvimento de *workshops* na empresa, onde se tenciona efetuar um planeamento e preparação dos mesmos com novas linguagens e/ou máquinas, de forma cíclica. Aqui, terá de haver participação nos *workshops* por parte de todos os

intervenientes e, por fim, cada participante pode realizar um exame, de forma a obter um certificado do *workshop* frequentado.

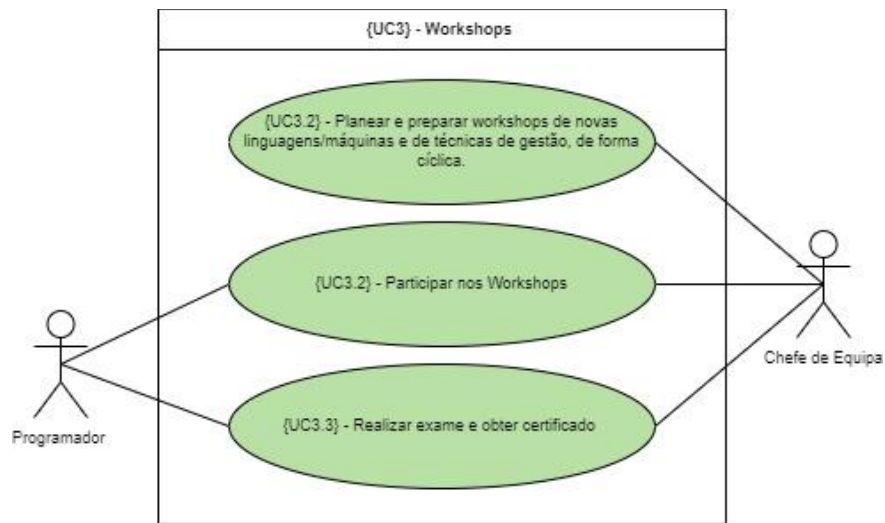


Figura 20 - {UC3} - Workshops

Ao nível de controlo de Qualidade/Testes, que pode ser visualizado na Figura 21, foi estabelecida a mudança da reunião semanal de *feedback* e discussão de dificuldades para uma reunião diária e breve, ou cíclica com maior frequência. Este cenário surgiu após a comprovação, via entrevistas estruturadas e análise dos processos, que ter uma reunião semanal para este tipo de questões acaba por tornar o seu propósito obsoleto ou impraticável, visto que ninguém espera e guarda dúvidas especificamente para serem discutidas uma vez por semana, num dia definido. Outra nova tarefa para o programador passa pelo registo regular e contínuo de comentários perceptíveis e explicativos à medida que codifica. Esta tarefa tem como finalidade, que se atinja o maior rigor e standard ao nível de boas práticas de programação e assim permitir um fácil retorno aquando da leitura do código no futuro. Desta forma é possível conceder a outros programadores, que devem trabalhar nesse mesmo código, uma boa compreensão dessas mesmas funcionalidades. Finalmente, foram ainda consideradas duas novas tarefas, nomeadamente, a escrita e manutenção do guia de boas práticas e a validação cíclica do código, ambas realizadas pelo técnico da qualidade. Estas tarefas em conjunto têm como principal finalidade atingir-se um nível *standard* no que toca ao controlo de qualidade, sendo estabelecido que essa responsabilidade deve ser feita por equipas externas aos programadores que trabalham nos seus próprios códigos. Desta forma, devia haver um documento oficial e padrão que indicia as práticas normalizadas na programação, garantindo que todas as equipas de desenvolvimento trabalhem da

mesma forma. Haver também, controlo de qualidade regulares sobre o trabalho desenvolvido pelos programadores, garantindo que as suas tarefas estão a ser respeitadas como, por exemplo, a escrita regular de comentários no código.

Em jeito de resumo, nesta fase, os atores devem atualizar diariamente o controlo de versões com todas as notas necessárias para um bom trabalho, realizar um registo diário das taxas de realização e justificar os atrasos. Devem também, fazer o registo de dificuldades na qual é preciso também requisitar, convidar e realizar reuniões breves para *feedback* e discussão de dificuldades (quando requisitado esta deverá ser feita no dia seguinte, o mais tardar) e preparar testes (*software*, elétricos e mecânicos).

Depois dos produtos estarem prontos, é necessário realizar testes, onde deve-se fazer o registo e visualizar as anomalias, validar as funcionalidades e testes antes de chegar a fase final do produto. Chegando essa fase, deve-se validar os testes (fase final) e, caso seja necessário, fazer o retrabalho/rever todo o Desenho 3D, escrever de forma regular, contínua e normalizada comentários no código e definir e escrever (com atualizações cíclicas) o guia de boas práticas na codificação.

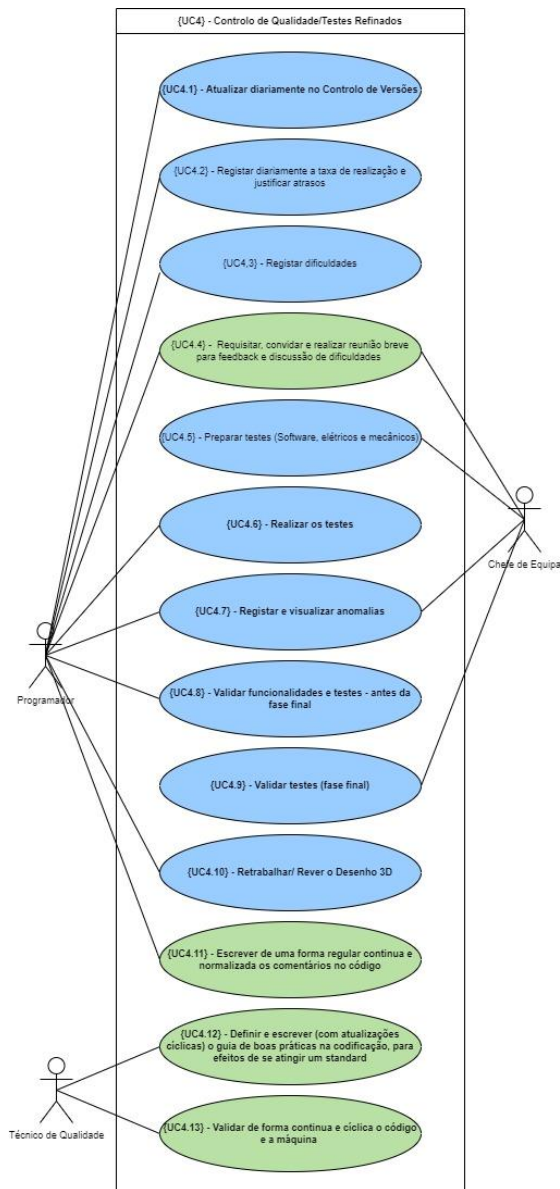


Figura 21 - {UC4} - Controlo de Qualidade/Testes Refinados

Em relação ao Serviço Pós-Venda, as tarefas são as mesmas das apresentadas nos casos de uso iniciais, com a exceção da “{UC5.1} - Proatividade na recolha de *feedback* do cliente”. Esta tarefa tem como propósito criar uma relação mais saudável e normalizada após a instalação dos produtos finais, com o objetivo de procurar a opinião do cliente, seja por questionário, seja por contacto direto. Desta forma é possível dar a oportunidade ao cliente de fornecer uma opinião fresca. Por outro lado, quando se espera que o cliente voluntariamente transmita algum tipo de *feedback*, por norma serão sempre informações de foro negativo. Isto acontece porque, quando o produto se encontra a funcionar como o esperado, o cliente, por esquecimento ou por não sentir a necessidade, não irá dar qualquer

feedback. Desta forma, o cliente irá só entrar em contacto com a empresa apenas se algo acontecer, sob a forma de pedido de assistência em garantia.

Na Figura 22, tem-se presente a atividade dos serviços pós-venda para a ITEC, onde deve ser realizada a recolha de *feedback* dos clientes de forma mais proativa, fazer o registo e gerir os pedidos de reclamação/assistência em garantia, atribuir o pedido de assistência a um programador, realizar a instalação da máquina no cliente final e, caso necessário, fazer o retrabalho e/ou desenvolver uma nova solução após pedido de reclamação/assistência por parte do cliente.

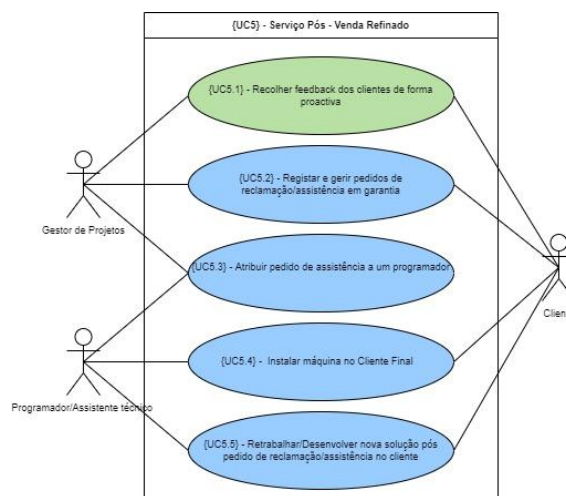


Figura 22 - {UC5} - Serviço Pós-Venda Refinado

3.4 Requisitos dos Processos de Desenvolvimento

Os requisitos de um sistema definem o que ele deve fazer e como ele deve operar. Os requisitos de processo definem a forma como o processo deve ocorrer. Para cada requisito é atribuída uma prioridade, de acordo com uma técnica de agrupamento que consiste na atribuição de diferentes prioridades aos requisitos. O método utilizado neste trabalho é o *MosCow*, que foi desenvolvido por Dai Clegg em 1994 [3] e é uma técnica de priorização usada em desenvolvimento de *software* para se chegar a um entendimento comum com as partes interessadas sobre a importância de cada requisito. Apesar de todos os requisitos serem importantes, é necessária à sua priorização de forma a fornecer os melhores e mais imediatos benefícios no início do projeto. O termo *MosCow* é um acrónimo vindo do inglês derivado da primeira letra de cada um dos quatro grupos de prioridades distintos [36].

- **Must Have:** requisitos com prioridade elevada, que são críticos para que o prazo de entrega seja cumprido;
- **Should have:** requisitos a serem considerados como importantes, mas não necessários para o prazo de entrega pretendido. Embora estes requisitos possam ser tão importantes como os anteriores não são tão críticos no que toca ao cumprimento do prazo de entrega;
- **Could have:** requisitos desejáveis, mas não imprescindíveis, podendo ser incluídos se o tempo e os recursos assim o permitirem. São requisitos que podem melhorar a experiência do utilizador ou a satisfação do cliente por um pequeno custo de desenvolvimento;
- **Won't have:** requisitos que podem ser considerados futuramente uma vez acordado entre as partes interessadas como itens menos críticos, de menor retorno ou interesse, ou não apropriados naquele momento.

Colocado este cenário de prioridades, são utilizados inicialmente todos os requisitos “*Must have*”, “*Should have*” e “*Could have*”, sendo que se houver qualquer tipo de imprevisto que comprometa o prazo de entrega, os requisitos “*Should have*” e “*Could have*” serão os primeiros a ser removidos. Simplificando, estas categorias são importantes para fazer com que os clientes entendam melhor o impacto da definição de prioridades.

Para uma melhor compreensão e contextualização dos requisitos identificados no contexto, foi definida uma estrutura onde cada uma se caracteriza pelos seguintes conjuntos de propriedades:

- **ID:** identificador do requisito;
- **Nome:** nome do requisito;
- **Descrição:** descrição do requisito;
- **Fonte do Requisito:** permite identificar as fontes de onde o requisito pode ser extraído, tais como, Gestores de Projetos, Chefe de equipa, Programadores, Técnicos de Qualidade, entre outros;
- **Interfaces:** composta pelos Utilizadores Finais, que correspondem às partes interessadas que serão parte integrante do novo processo a ser implementado.
- **Dependências:** permitem a identificação de:
 - a) partes direta ou indiretamente interessadas no cumprimento do requisito;
 - b) que Fontes são necessárias para recolher os dados que cumprem o requisito.

- **Identificação do Problema:** identifica os possíveis constrangimentos ou problemas para cumprir o requisito (por exemplo, falta de acesso a uma fonte de dados específica);
- **Prioridades:** permite a definição da prioridade dos requisitos através do método *MoSCoW*.

Deste modo, apresentamos na Tabela 2 os requisitos para a definição de um novo processo de gestão para desenvolvimento de *software*, ou seja, os requisitos que são efetivamente traduzidos em novos métodos e tarefas que serão suportadas pelo novo processo, permitindo aos colaboradores usufruir de boas práticas relativamente ao desenvolvimento de *software*.

ID	Requisitos	Prioridade
RP1	Elaborar calendário dos colaboradores	Should
RP 2	Definir objetivos atribuindo peso ao tempo	Must
RP 3	Realizar reunião de planeamento por projeto	Must
RP 4	Planear e alocar tarefas atribuindo peso à dificuldade	Must
RP 5	Analisar o caderno de encargos de forma normalizada	Must
RP 6	Esquematizar e planear o <i>software</i> de forma normalizada	Must
RP 7	Analisar pedidos extra atribuindo peso à dificuldade e tempo	Should
RP 8	Planear e preparar <i>workshops</i>	Must
RP 9	Participar em <i>workshops</i>	Must
RP 10	Obter certificado de frequência nos workshops	Should
RP 11	Realizar reunião breve de <i>feedback</i> e dificuldades	Must
RP 12	Escrever de forma regular comentários no código	Must
RP 13	Definir e escrever o guia de boas práticas	Must
RP 14	Validar código de forma contínua	Must
RP 15	Obter feedback dos clientes	Must

Tabela 2 - Requisitos de Processo

Na Tabela 3, podemos retirar a informação detalhada para o requisito de processos 1 (RP1), relativo à elaboração do calendário dos colaboradores, em que temos a descrição do requisito, as suas fontes, as interfaces, quem depende da realização deste requisito, os problemas identificados e por último no nível de prioridade que o requisito deve ser realizado. O mesmo acontece para os restantes

requisitos levantados, onde podemos observar a informação dos mesmos no Apêndice I, em que também temos uma descrição detalhada de cada um deles, os intervenientes e qual o seu nível de prioridade.

ID: RP1 - Elaborar calendário dos colaboradores
Descrição: Seria importante que fosse adotado pela empresa, a leitura e consideração do calendário de todos os membros da equipa antes das reuniões <i>Scrum</i> , onde ficando a cargo do Chefe de Equipa promover a gestão desta tarefa todas as semanas, sempre com a confirmação do gestor de projetos.
Fonte do Requisito: Chefe de Equipa e Gestor de Projetos.
Interfaces: N/A
Dependência: <ul style="list-style-type: none"> ● Stakeholders <ul style="list-style-type: none"> ○ Chefe de Equipa ○ Programadores
Problemas Identificados: N/A
Prioridades: Should

Tabela 3 - RP1 Elaborar calendário dos colaboradores

3.5 Conclusão

Com base nos detalhes descritos e analisados ao longo deste capítulo foi-nos possível elaborar e pensar num conjunto de requisitos para o novo processo de gestão de ciclo de vida e qualidade para produtos de software.

Assim, alguns dos pontos a melhorar podem passar pela divisão da reunião de planeamento em várias reuniões, uma para cada projeto, a realização de um *daily scrum* efetivamente diário e incutir o espírito de autocritica na capacidade de conclusão das tarefas.

Estas soluções têm como finalidade promover uma melhor gestão dos projetos e dos respetivos prazos, assim como deixar questões relativas a dificuldades diárias das tarefas fora do contexto de planeamento.

Com as interações que promovemos na ITEC foi-nos possível concluir a não utilidade de uma reunião semanal fixa para dúvidas, onde é aconselhável trocar para uma reunião breve diária. Também concluímos que existe uma falta de *standard*, ao comprovar que durante a reunião dedicada ao planeamento de projetos é possível que surjam dúvidas técnicas individuais de projetos específicos.

Por fim a análise de todo o ciclo de vida dos projetos, bem como das diferentes áreas, permitiu-nos delinear os requisitos de processo que entendemos serem adequados para promover uma melhoria ao nível dos processos de desenvolvimento de software.

Um dos pontos mais importantes deste trabalho, é a sugestão dos *Workshops*, com o intuito de criar uma espécie de academia dentro da ITEC, para que haja partilha de conhecimento, para que haja pessoas cada vez mais interessadas e motivadas, pois estão constantemente em aprendizagem contínua e a poder encontrar desafios ao longo dos projetos que tenham em mãos.

Sendo que esta parte do trabalho desenvolvido, é muito importante para que se consiga no capítulo seguinte elaborar um trabalho que vá de acordo com o pretendido pela empresa em estudo.

4. ABORDAGEM PARA A MELHORIA DOS PROCESSOS DE DESENVOLVIMENTO COM NEO4J

4.1 Introdução

Nesta secção são apresentados os procedimentos necessários para o mapeamento dos *frameworks* em estudo e também o mapeamento com as informações recolhidas relativas à ITEC, utilizando o *Neo4j*. Sendo o objetivo final e trabalho futuro a criação de um guia de boas práticas, que irá ajudar a empresa nos procedimentos diários.

No entanto, são utilizadas duas ferramentas distintas para efetuar o nosso estudo, o *Neo4j* para realizar as relações entre os dois *frameworks* e atribuir pesos às relações existentes, o mesmo é feito para a relação com as tarefas já desenvolvidas pela empresa e outra ferramenta para a construção das respetivas tabelas. Contudo, são atribuídos níveis de cobertura, níveis de prioridade e níveis de urgência das tarefas a realizar pela empresa.

Sendo que inicialmente é utilizada uma ferramenta de criação de tabelas, para a realização desse mesmo estudo, para que essa informação sirva de suporte e haja informações mais completas, para permitir que seja possível introduzir toda a informação no *Neo4j*. Por fim, neste capítulo são demonstrados todos os procedimentos da criação das tabelas, em que são explicados todos os procedimentos e o mesmo acontece com o mapeamento de toda a informação que foi colocada no software de visualização da base de dados por grafos.

4.2 Análise do CMMI e do OpenUp com Neo4j

Nesta fase do trabalho, são mapeados os dois *frameworks* em estudo, e para que seja de mais fácil compreensão são explicados os procedimentos e todos os elementos presentes em cada um dos *frameworks* que são utilizados para este estudo. Posto isto, de seguida são apresentados os níveis de maturidade presentes no CMMI, e que são utilizados no estudo.

Mapeamento dos Níveis de Maturidade do CMMI utilizando o Neo4j

Após a análise da empresa em estudo e pelo facto de estarmos perante uma pequena média empresa, chegamos à conclusão que relativamente ao *framework* em estudo (CMMI), ao utilizar os processos que o nível 2 de maturidade proporciona, este faz com que a empresa obtenha os objetivos desejados. Podemos observar na Figura 23, o mapeamento dos níveis de maturidade no CMMI, que são utilizados no decorrer deste trabalho e que munem a empresa com processos para desempenhar as suas tarefas diárias com a melhor qualidade possível.

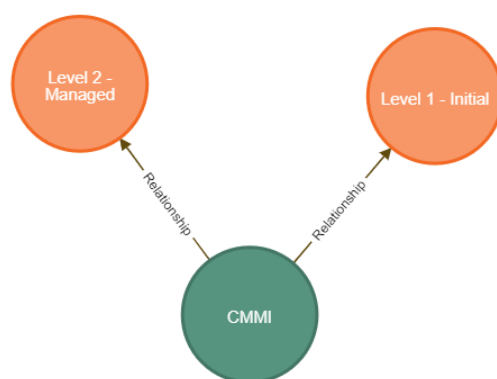


Figura 23 - Mapeamento Níveis de maturidade *CMMI*

Depois de demonstrados os dois níveis de maturidade que são utilizados neste projeto, no ponto seguinte são apresentadas as relações existentes entre as Categorias do CMMI, tendo em consideração a informação que foi apresentada no Capítulo 2 no Estado da Arte, recorrendo à tecnologia *Neo4j*.

Representação das relações das Categorias no CMMI utilizando o Neo4j

As categorias presentes no CMMI, estão divididas em quatro categorias distintas, na qual têm diferentes áreas de capacitação. Contudo, existe uma relação entre as categorias, para que haja uma sequência certa dos procedimentos a serem realizados, para que o produto final seja entregue com a melhor qualidade possível. Podemos observar essa relação na Figura 24.

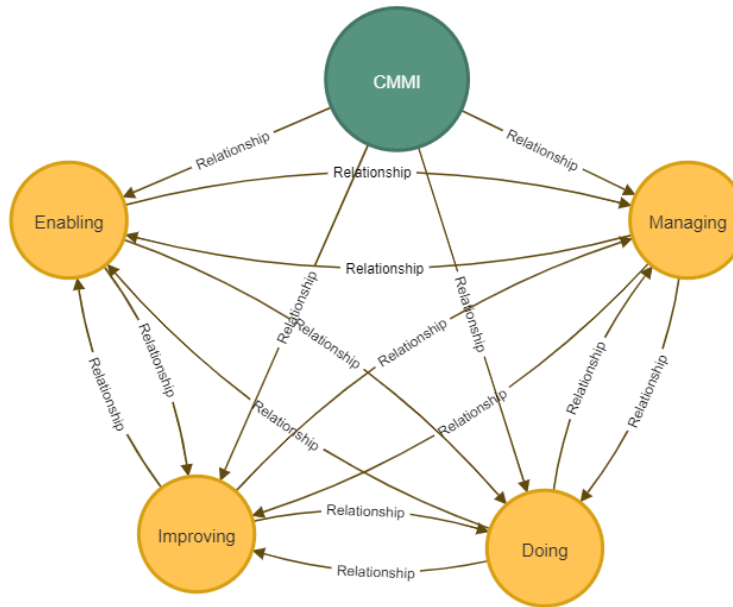


Figura 24 - Mapeamento da Relação entre Categorias do CMMI

Após efetuar a relação entre as categorias, de seguida são demonstrados os mapeamentos realizados em cada uma das quatro categorias existentes, onde se tem consideração as áreas de atuação, nível de maturidade e as “*practice summary*”, tendo em consideração a informação apresentada no Capítulo 2. Na primeira fase, será apresentada a Categoria “*Doing*” e todas as suas relações.

Mapeamento da Categoria *Doing* e respetivas áreas de atuação

Na Figura 25, é realizado o mapeamento da categoria “*Doing*”, que está dividida em quatro áreas de capacidade. Temos também as suas respetivas áreas de atuação e as “*Practice Summary*” de cada área de atuação. Temos então a verde o CMMI, a amarelo as categorias presentes no CMMI, a azul representa as quatro áreas de capacidade existentes nesta categoria, a vermelho as áreas de atuação presentes em cada uma das áreas de capacidade, a castanho-claro as “*Practice Summary*” e por último, temos a laranja os níveis de maturidade presentes.

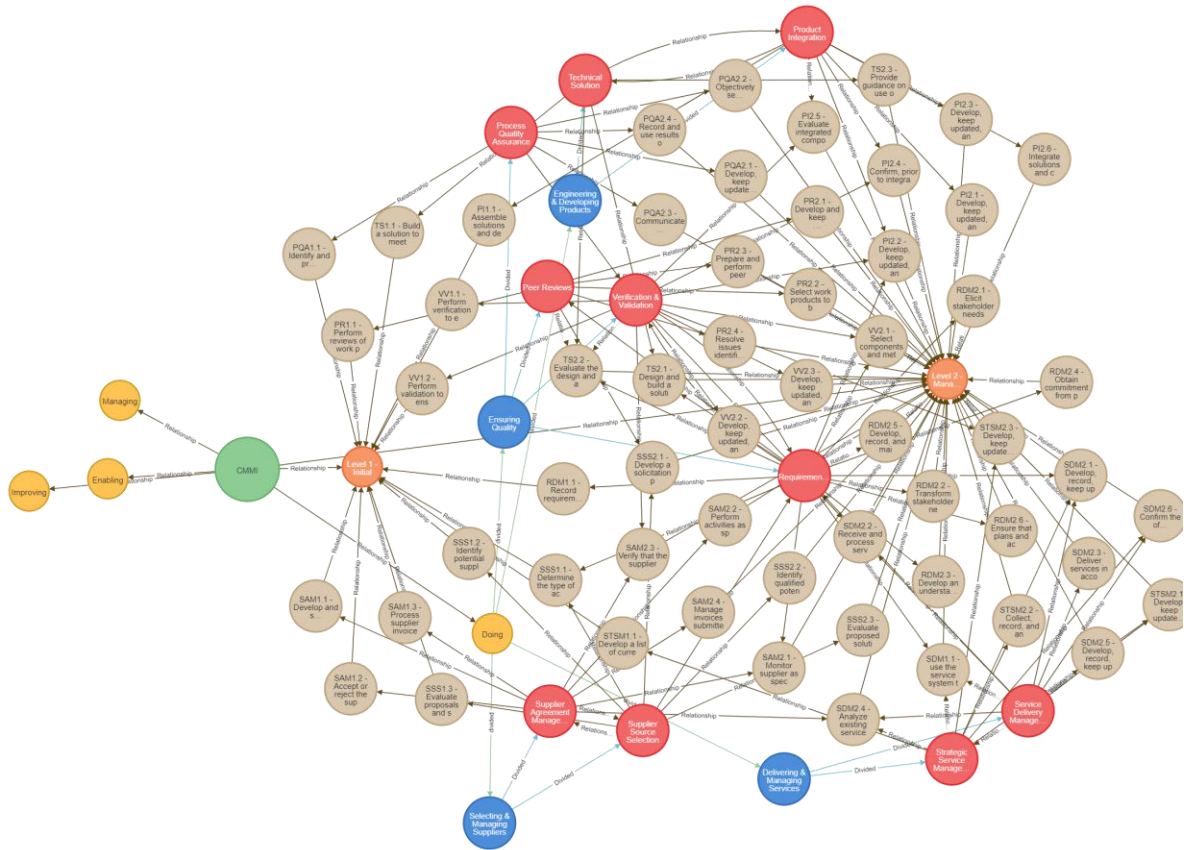


Figura 25 - Mapeamento das Practice Summary da categoria Doing do CMMI

De seguida, o mesmo será realizado para a categoria “*Managing*”, onde são apresentadas todas as relações existentes, que são utilizadas neste estudo.

Mapeamento da Categoria *Managing* e respetivas áreas de atuação

Na Figura 26, temos o mapeamento realizado da categoria “*Managing*”, que é realizado da mesma forma da anteriormente apresentada, em que temos a verde a representação do CMMI, a amarelo as categorias, a azul representa as três áreas de capacidade existentes, a vermelho temos as áreas de atuação presentes em cada um das áreas de capacidade, a castanho-claro as “*Practice Summary*” e a laranja os níveis de maturidade presentes.

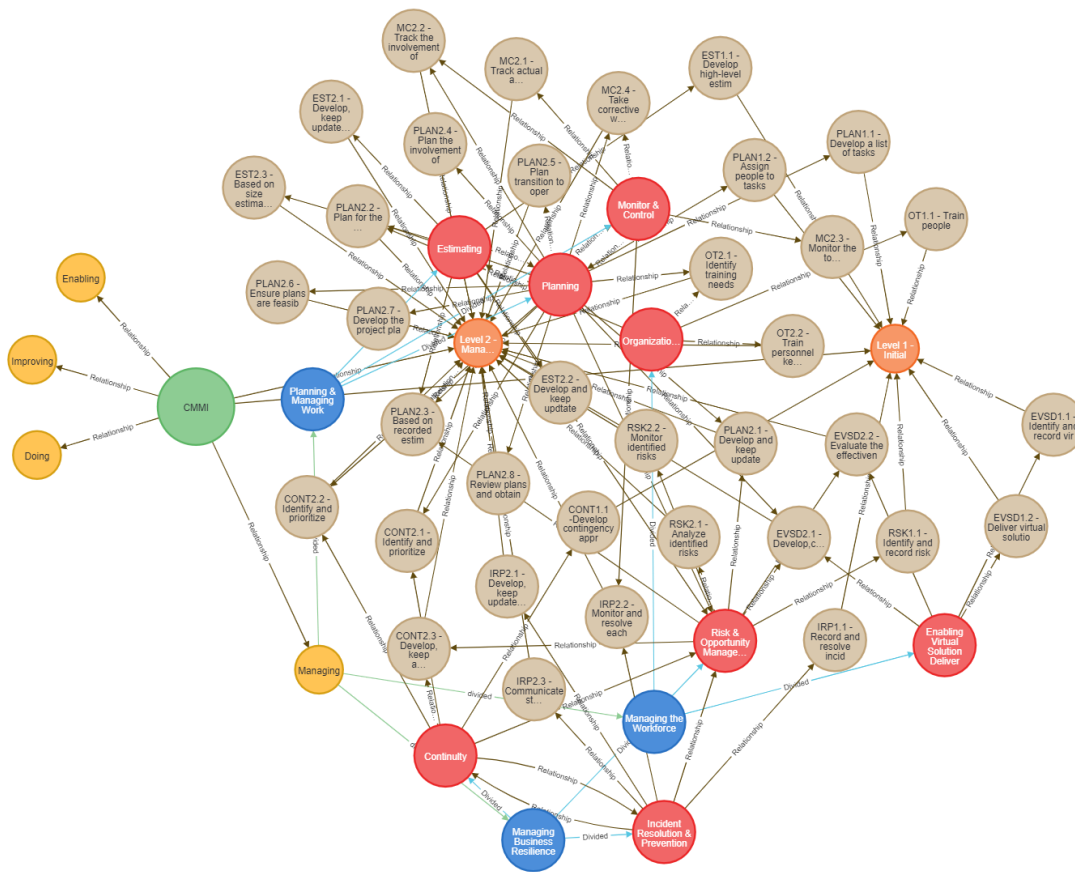


Figura 26 - Mapeamento das Practice Summary da categoria Managing do CMMI

De seguida, o mesmo será realizado para a categoria “*Enabling*”, onde são apresentadas todas as relações existentes, que são utilizadas neste estudo.

Mapeamento da Categoria *Enabling* e respetivas áreas de atuação

Na Figura 27, temos o mapeamento realizado da categoria “*Enabling*”, que é realizado da mesma forma da anteriormente apresentada, em que temos a verde a representação do CMMI, a amarelo as categorias, a azul representa as duas áreas de capacidade existentes, a vermelho temos as áreas de atuação presentes em cada um das áreas de capacidade, a castanho-claro as “*Practice Summary*” e a laranja os níveis de maturidade presentes.

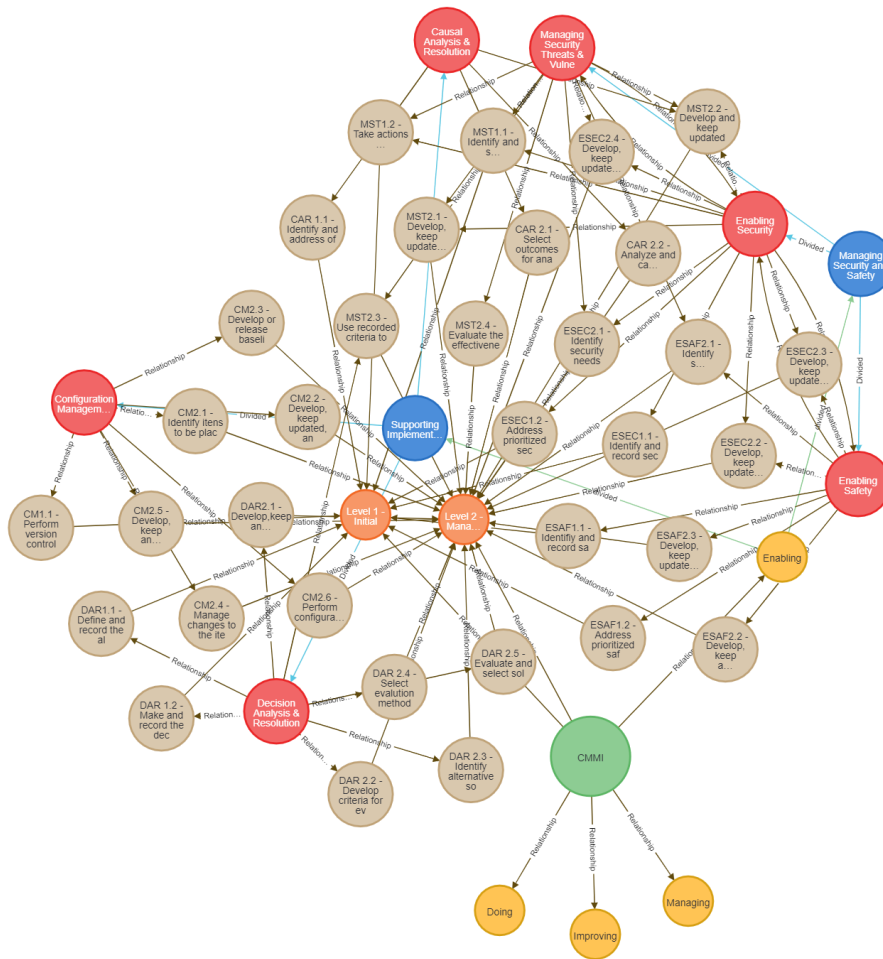


Figura 27 - Mapeamento das Practice Summary da categoria Enabling do CMMI

De seguida, o mesmo será realizado para a categoria “*Improving*”, onde são apresentadas todas as relações existentes, que são utilizadas neste estudo.

Mapeamento da Categoria *Improving* e respetivas áreas de atuação

Na Figura 28, temos o mapeamento realizado da categoria “*Improving*”, que é realizado da mesma forma da anteriormente apresentada, em que temos a verde a representação do CMMI, a amarelo as categorias, a azul representa as duas áreas de capacidade existentes, a vermelho temos as áreas de atuação presentes em cada um das áreas de capacidade, a castanho-claro as “*Practice Summary*” e a laranja os níveis de maturidade presentes.



Figura 28 - Mapeamento das Practice Summary da categoria Improving do CMMI

OpenUp – Tarefas OpenUp

Por último, é demonstrando o mapeamento relativo ao *OpenUp*, e no caso do estudo o que nos interessa apresentar são as *tarefas* presentes no mesmo e que são utilizadas para relacioná-las com a informação em cima demonstrada relativa ao CMMI.

Relativamente ao *OpenUp* são mapeadas as suas respetivas *tarefas*, onde estão divididas em áreas de atuação, com o objetivo de ser de mais fácil utilização e compreensão por parte do utilizador. Sendo também mais fácil realizar a relação entre este *framework* com o CMMI.

Na Figura 29, podemos observar a vermelho o *OpenUp*, a verde-claro temos a referência às disciplinas, castanho-escuro temos as diferentes áreas de atuação das *tarefas* e por fim, com a cor azul-escuro temos os nomes das *tarefas* que são desempenhadas em cada área de atuação.

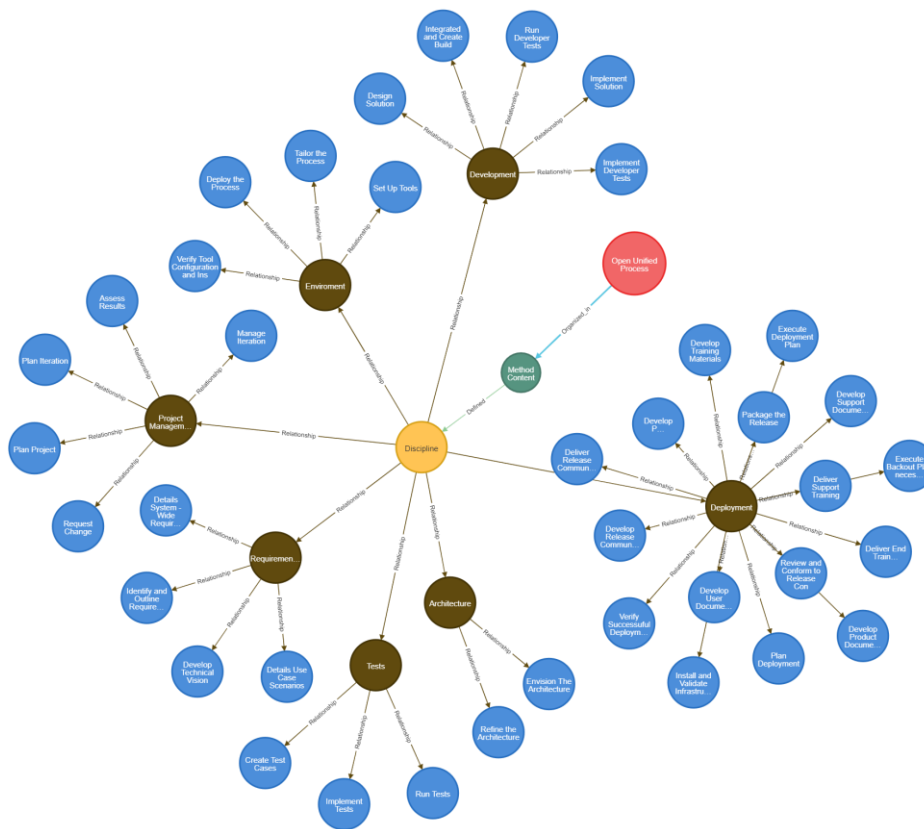


Figura 29 - Mapeamento das disciplinas do OpenUp (Tenho de alterar esta imagem)

Em forma de resumo, todas as representações apresentadas anteriormente, servem como base para o mapeamento que é mostrado no ponto seguinte, onde nesta fase é realizado o mapeamento entre os dois referenciais e demonstrado todos os processos que foram criados e quais os objetivos dos mesmos.

4.3 Análise do Relacionamento entre o CMMI e o OpenUp

Com vista á criação de um guia de boas práticas para utilização futura da empresa, foram estudadas as relações entre os dois *frameworks*, em que para a relação entre os dois foram elaboradas com ajuda de tabelas e mapeamento em *neo4j*. Para que fosse possível definir níveis de prioridade, foram criados quatro níveis que achamos que são os melhores para representar o nosso estudo.

Níveis de cobertura

Por isso, tendo como objetivo a melhoria contínua da empresa, em que atribuímos com H+ (verde-claro), quando a relação entre a *Practice Summary* do CMMI e as disciplinas do *OpenUp* é de elevada importância e de máxima urgência, onde se a empresa não efetuar essa tarefa não consegue de forma alguma entregar o produto. Quando atribuímos com H, é uma tarefa de alta importância, em que também tem de ser realizada para que o produto possa ser entregue e com a qualidade pretendida, mas com menos importância que a anterior. Quando atribuímos com a letra M é quando a tarefa tem uma importância média, em que a tarefa em causa consideramos que deveria ser realizada, mas que se não for elaborada não vai trazer grande impacto à entrega do produto final. Por fim, quando atribuímos com a letra L, é no caso de a tarefa ser de muito baixa importância, porém, consideramos que deveria ser realizada. Todavia, não consideramos obrigatória, podemos observar os diferentes níveis propostos na Tabela 4.

H+	Prioridade Altíssima
H	Prioridade Alta
M	Prioridade Média
L	Prioridade Baixa

Tabela 4 - Níveis de Cobertura

De seguida, é apresentada como foi construída a tabela com a informação relativa ao CMMI e ao *OpenUp*, construída com a finalidade de ser possível utilizar os níveis de prioridade em cima apresentados.

Construção da tabela relação CMMI e OpenUp

De modo apresentar como as tabelas estão estruturadas, é apresentada a informação presente em cada uma delas, com o objetivo de facilitar a compreensão da mesma. Na Figura 30, podemos observar um exemplo de uma “*Practice Area*” presente no CMMI, onde também contém a informação da “*Category*”, “*Capability Area*” e “*Practice Summary’s*”, com a informação do respetivo nível de maturidade. Como já referido anteriormente, a empresa esta dividida em dois níveis de maturidade, em estão representados por ML1 e ML2 respetivamente, pois ao estudarmos o CMMI chegamos à conclusão de que estes níveis tem o necessário para implementar na empresa e atingir os objetivos desejados nos

procedimentos diários da empresa. A Figura 30, serve apenas como referência, pois foram feitas 29 tabelas iguais tendo em consideração cada *Practice Area* presente no CMMI.

Doing	Ensuring Quality (ENQ)	Requirements Development and Management (RDM)	Level	Practice Summary	
			ML1		RDM 1.1 Record requirements.
			ML2		RDM 2.1 Elicit stakeholder needs, expectations, constraints, and interfaces or connections.
					RDM 2.2 Transform stakeholder needs, expectations, constraints, and interfaces or connections into prioritized customer requirements.
					RDM 2.3 Develop an understanding with the requirements providers on the meaning of the requirements.
					RDM 2.4 Obtain commitment from project participants that they can implement the requirements.
					RDM 2.5 Develop, record, and maintain bidirectional traceability among requirements and activities or work products.
					RDM 2.6 Ensure that plans and activities or work products remain consistent with requirements.

Figura 30 - Exemplo de uma Practice Area (CMMI)

De modo a realizar o cruzamento é introduzido em cada tabela já mencionada, as disciplinas presentes no *OpenUp*, onde é feita a separação em tarefas, como podemos observar na Figura 31, onde estas foram divididas por cores para ser mais fácil conseguirmos distinguir as áreas presentes neste *framework*.

Refine the Architecture	Architecture
Envision the Architecture	
Assess Results	Project Management
Manage Iteration	
Plan Iteration	
Plan Project	
Request Change	Requirements
Identify and Outline Requirements	
Detail Use-Case Scenarios	
Detail System-Wide Requirements	
Develop Technical Vision	Development
Implement Developer Tests	
Implement Solution	
Run Developer Tests	
Integrate and Create Build	
Design the Solution	Environment
Deploy the Process	
Tailor the Process	
Set Up Tools	Test
Verify Tool Configuration and Installation	
Create Test Cases	
Implement Tests	Deployment
Run Tests	
Develop Product Documentation	
Develop User Documentation	
Develop Support Documentation	
Deliver end user Training	
Deliver Support Training	
Develop Training Materials	
Deliver Release Communications	
Execute Backout Plan (if necessary)	
Execute Deployment Plan	
Package the Release	
Verify Successful Deployment	
Develop Backout Plan	
Develop Release Communications	
Install and Validate Infrastructure	
Plan Deployment	
Review and Conform to Release Controls	

Figura 31 - Tarefas do *OpenUp* por disciplina

Tendo em consideração a informação que já foi anteriormente referida, serão apresentados alguns exemplos de relacionamentos, tendo como referência os níveis de prioridade da Tabela 4. Podemos observar na Figura 32, a relação entre os dois *frameworks* em estudo, tendo em consideração a *Practice Area* “*Requirements Development and Management (RDM)*” que pertence à *Capability Area* “*Ensuring Quality (ENQ)*” que por sua vez pertence a *Category* “*Doing*” referente ao CMMI e relativamente ao *OpenUp* iremos ter as disciplinas onde foram atribuídas coberturas aos cruzamentos encontrados. A seguinte imagem serve apenas como exemplo de como foram realizadas as tabelas de cruzamentos de referências e atribuições dos níveis de cobertura, pois foram efetuados os mesmos procedimentos para todas as 29 “*Practice Area*” presentes no CMMI, sendo que na secção seguinte serão apresentados com maior detalhe.

Doing		Ensuring Quality (ENQ)		Requirements Development and Management (RDM)		Level	Level
ML2		ML1		Level		Level	Level
RDM 2.6 Ensure that plans and activities or work products remain consistent with requirements.		RDM 2.5 Develop, record, and maintain bidirectional traceability among requirements and activities or work products.		RDM 2.4 Obtain commitment from project participants that they can implement the requirements.		RDM 2.3 Develop an understanding with the requirements providers on the meaning of the requirements.	
RDM 2.2 Transform stakeholder needs, expectations, constraints, and interfaces or connectors into prioritized customer requirements.		RDM 2.1 Elicit stakeholder needs, expectations, constraints, and interfaces or connectors.		RDM 1.1 Record requirements.		Practice Summary	
						Refine the Architecture	
						Envision the Architecture	
H						Assess Results	
M		M				Manage Iteration	
						Plan Iteration	
		H		M		Plan Project	
				M		Request Change	
				H		Identify and Outline Requirements	
				H		Detail Use-Case Scenarios	
				H		Detail System-Wide Requirements	
		H		H		Develop Technical Vision	
						Implement Developer Tests	
						Implement Solution	
						Run Developer Tests	
						Integrate and Create Build	
H		H		M		Design the Solution	
H				H		Deploy the Process	
				H		Tailor the Process	
						Set Up Tools	
						Verify Tool Configuration and Installation	
M						Create Test Cases	
M						Implement Tests	
H						Run Tests	
		H				Develop Product Documentation	
		H				Develop User Documentation	
		H				Develop Support Documentation	
						Deliver end user Training	
						Deliver Support Training	
						Develop Training Materials	
						Deliver Release Communications	
H		M				Execute Backout Plan (if necessary)	
						Execute Deployment Plan	
						Package the Release	
H						Verify Successful Deployment	
						Develop Backout Plan	
						Develop Release Communications	
						Install and Validate Infrastructure	
		H				Plan Deployment	
H						Review and Conform to Release Controls	

Figura 32 - Relação entre CMMI e OpenUp

4.4 Proposta para a Melhoria dos Processos de Desenvolvimento

Nesta secção começamos por demonstrar o mapeamento realizado relativo à informação obtida acerca da ITEC, onde também utilizamos a informação que foi recolhida no Capítulo 3, e que serviram para mapear os casos de uso refinados e respetivos requisitos de processos.

Mapeamento da ITEC

Em primeiro lugar, é criada a base de dados relativa à ITEC, na qual se coloca a informação sobre a empresa, quais os principais intervenientes e principais interessados.

De seguida são apresentadas as nossas propostas de melhoria dos casos de uso levantados e por fim também são introduzidos todos os requisitos de processos levantados.

Na Figura 33, é apresentado o código utilizado na ferramenta *neo4j*, em que são introduzidos os nodos relativos à organização, aos colaboradores da empresa (*Sources of Requirements*) e os principais interessados (*stackholders*), esta informação é necessária para que seja possível referir quem desempenha as atividades na empresa.

```
// CRIAÇÃO DA ORGANIZAÇÃO (ITEC)
CREATE (n:Organization{name:"Iberiana Technical",id:"ITEC",Fundada:"2006",Ramo:"Electronica e automovel"});

// Criar Sources Of Requirements
CREATE (n:Sources_Of_Requirements{name:"Team Manager",id:"CE"}); // Chefe de equipa
CREATE (n:Sources_Of_Requirements{name:"Project Manager",id:"GP"}); // Gestor de Projetos
CREATE (n:Sources_Of_Requirements{name:"Developer",id:"Prog"}); // Programador
CREATE (n:Sources_Of_Requirements{name:"Quality Manager",id:"TQ"}); // Técnico de Qualidade
CREATE (n:Sources_Of_Requirements{name:"Customers",id:"C"}); // Cliente

// _____

// Criar Partes Interessadas
CREATE (n:Stackholders{name:"Team Manager",id:"CE"}); // Chefe de equipa
CREATE (n:Stackholders{name:"Project Manager",id:"GP"}); // Gestor de Projetos
CREATE (n:Stackholders{name:"Developer",id:"Prog"}); // Programador
CREATE (n:Stackholders{name:"Quality Manager",id:"TQ"}); // Tecnico de Qualidade
CREATE (n:Stackholders{name:"Customers",id:"C"}); // Cliente
```

Figura 33 – Código Neo4j (Empresa, Colaboradores e Stackholders)

Posto isto, na Figura 34, podemos observar parte do código dos casos de uso a que chamamos de refinados, que surgiram da análise realizada e demonstrada no Capítulo 3, onde são apresentados os casos de uso refinados e servem como suporte para a definição dos requisitos de processos levantados.

```

// CASOS DE USO ITEC

// Use Cases - Project Management
CREATE (n:UseCasesTobe{name:"Project Management",id:"GPS"});

MATCH (UCT:UseCasesTobe), (Org:Organization) // Ligação entre ITEC e Use Cases Project Management
WHERE UCT.id="GPS" AND Org.id="ITEC"
CREATE (Org) - [:Contains]- > (UCT);

.....
// Activities - Use Cases - Project Management
//.....
CREATE (n:Activities{name:"Deadline",id:"P"}); // Actividades Realizadas - Prazos

MATCH (A:Activities), (UCT:UseCasesTobe) // Ligação entre Deadline e Use Cases Project Management
WHERE A.id="P" AND UCT.id="GPS"
CREATE (UCT) - [:has]- > (A);

.....
CREATE (n:Activities{name:"Project Dates",id:"DP"}); // Actividaes Realizadas - Datas dos Projetos

MATCH (A:Activities), (UCT:UseCasesTobe) // Ligação entre Project Dates e Use Cases Project Management
WHERE A.id="DP" AND UCT.id="GPS"
CREATE (UCT) - [:has]- > (A);

.....
CREATE (n:Activities{name:"1st Analysis of the Specifications",id:"IACE"}); // Actividades realizadas - 1ª Análise do Caderno de Encargos

MATCH (A:Activities), (UCT:UseCasesTobe) // Ligação entre 1st Analysis of the Specifications e Use Cases Project Management
WHERE A.id="IACE" AND UCT.id="GPS"
CREATE (UCT) - [:has]- > (A);

.....
CREATE (n:Activities{name:"Organize Kickoff Meeting", id:"ORK"}); // Actividades Realizadas - Organizar reunião kickoff

MATCH (A:Activities), (UCT:UseCasesTobe) // Ligação entre Organize Kickoff Meeting e Use Cases Project Management
WHERE A.id="ORK" AND UCT.id="GPS"
CREATE (UCT) - [:has]- > (A);

.....
CREATE (n:Activities{name:"Planning Meeting",id:"RP"}); // Actividades Realizadas - Reunião de Planeamento

MATCH (A:Activities), (UCT:UseCasesTobe) // Ligação entre Planning Meeting e Use Cases Project Management
WHERE A.id="RP" AND UCT.id="GPS"
CREATE (UCT) - [:has]- > (A);

```

Figura 34 - Código exemplo dos Casos de uso da ITEC

Depois de introduzido o código relativo à informação em cima referido, o mapeamento observado na Figura 35, temos a vermelho o nodo relativo à ITEC, a azul-claro são os colaboradores que desempenham as tarefas da empresa, a azul – escuro temos as cinco áreas encontradas na empresa, a amarelo são as atividades que são desempenhadas em cada uma das áreas apresentadas a azul escuro, a castanho escuro temos os *stackholders*, a verde claro temos os requisitos de processos levantados e por fim temos a castanho-claro os níveis de prioridade com que os requisitos de processos devem ser executados.



Figura 35 - Mapeamento da ITEC

No entanto, chegamos à conclusão que seria interessante acrescentar à tabela anteriormente apresentada relativa às relações entre CMMI e *OpenUp* (Figura 32), a informação relativa à ITEC, que é apresentada de seguida.

Melhoramento da tabela da relação entre CMMI e OpenUp

Depois da relação efetuada na secção 4.2, chegou-se à conclusão que seria importante acrescentar informação relativa às atividades realizadas pela empresa, por esse motivo surgiu a Figura 36, em que foi adicionada a informação sobre as *tarefas* desempenhadas pela ITEC, qual o nível de cobertura que essa *tarefas* tem que ser realizadas para que o produto esteja pronto e com a qualidade pretendida. Por fim, utilizando a relação entre o nível de cobertura e o nível de prioridade surgiram os níveis de urgência, com a qual cada *tarefas* deve ser desempenhada para que o produto passe por todos os procedimentos necessários para que seja entregue com a máxima qualidade possível.

Maturity Level	Process Summary	Process Summary										Tasks ITEC	Detailed tasks	Coverage	Priority																					
		Business	System	Project Management	Requirements	Development	Environment	Test	Deployment	Release	Support																									
Initial	Practice Summary	Plan the Architecture	Review the Architecture	Assign Roles	Manage Risks	Plan Budget	Manage Change	Identify and Define Requirements	Identify User Requirements	Identify System Requirements	Develop Technical Vision	Implement Developer Tools	Plan Developer Tools	Integrate and Create Build	Design the Solution	Deploy the Process	Take the Process	Test the Process	Verify Tool Configuration and Installation	Install Tools	Implement Tools	Plan Tests	Develop Product Documentation	Develop User Documentation	Develop Support Documentation	Customer and User Training	Customer Support Training	Develop Training Materials	Customer Release Communications	Execute Backup Plan (if necessary)	Package the Release	Verify Successful Deployment	Develop Release Communications	Install and Activate Workflows	Test Deployment	Review and Confirm for Release Controls
Managed	Practice Summary	Plan the Architecture	Review the Architecture	Assign Roles	Manage Risks	Plan Budget	Manage Change	Identify and Define Requirements	Identify User Requirements	Identify System Requirements	Develop Technical Vision	Implement Developer Tools	Plan Developer Tools	Integrate and Create Build	Design the Solution	Deploy the Process	Take the Process	Test the Process	Verify Tool Configuration and Installation	Install Tools	Implement Tools	Plan Tests	Develop Product Documentation	Develop User Documentation	Develop Support Documentation	Customer and User Training	Customer Support Training	Develop Training Materials	Customer Release Communications	Execute Backup Plan (if necessary)	Package the Release	Verify Successful Deployment	Develop Release Communications	Install and Activate Workflows	Test Deployment	Review and Confirm for Release Controls
Defined	Practice Summary	Plan the Architecture	Review the Architecture	Assign Roles	Manage Risks	Plan Budget	Manage Change	Identify and Define Requirements	Identify User Requirements	Identify System Requirements	Develop Technical Vision	Implement Developer Tools	Plan Developer Tools	Integrate and Create Build	Design the Solution	Deploy the Process	Take the Process	Test the Process	Verify Tool Configuration and Installation	Install Tools	Implement Tools	Plan Tests	Develop Product Documentation	Develop User Documentation	Develop Support Documentation	Customer and User Training	Customer Support Training	Develop Training Materials	Customer Release Communications	Execute Backup Plan (if necessary)	Package the Release	Verify Successful Deployment	Develop Release Communications	Install and Activate Workflows	Test Deployment	Review and Confirm for Release Controls
Optimizing	Practice Summary	Plan the Architecture	Review the Architecture	Assign Roles	Manage Risks	Plan Budget	Manage Change	Identify and Define Requirements	Identify User Requirements	Identify System Requirements	Develop Technical Vision	Implement Developer Tools	Plan Developer Tools	Integrate and Create Build	Design the Solution	Deploy the Process	Take the Process	Test the Process	Verify Tool Configuration and Installation	Install Tools	Implement Tools	Plan Tests	Develop Product Documentation	Develop User Documentation	Develop Support Documentation	Customer and User Training	Customer Support Training	Develop Training Materials	Customer Release Communications	Execute Backup Plan (if necessary)	Package the Release	Verify Successful Deployment	Develop Release Communications	Install and Activate Workflows	Test Deployment	Review and Confirm for Release Controls
Advanced	Practice Summary	Plan the Architecture	Review the Architecture	Assign Roles	Manage Risks	Plan Budget	Manage Change	Identify and Define Requirements	Identify User Requirements	Identify System Requirements	Develop Technical Vision	Implement Developer Tools	Plan Developer Tools	Integrate and Create Build	Design the Solution	Deploy the Process	Take the Process	Test the Process	Verify Tool Configuration and Installation	Install Tools	Implement Tools	Plan Tests	Develop Product Documentation	Develop User Documentation	Develop Support Documentation	Customer and User Training	Customer Support Training	Develop Training Materials	Customer Release Communications	Execute Backup Plan (if necessary)	Package the Release	Verify Successful Deployment	Develop Release Communications	Install and Activate Workflows	Test Deployment	Review and Confirm for Release Controls
World Class	Practice Summary	Plan the Architecture	Review the Architecture	Assign Roles	Manage Risks	Plan Budget	Manage Change	Identify and Define Requirements	Identify User Requirements	Identify System Requirements	Develop Technical Vision	Implement Developer Tools	Plan Developer Tools	Integrate and Create Build	Design the Solution	Deploy the Process	Take the Process	Test the Process	Verify Tool Configuration and Installation	Install Tools	Implement Tools	Plan Tests	Develop Product Documentation	Develop User Documentation	Develop Support Documentation	Customer and User Training	Customer Support Training	Develop Training Materials	Customer Release Communications	Execute Backup Plan (if necessary)	Package the Release	Verify Successful Deployment	Develop Release Communications	Install and Activate Workflows	Test Deployment	Review and Confirm for Release Controls

Figura 36 – Tabela exemplo do cruzamento entre CMMI, OpenUp e ITEC

Para o preenchimento da tabela completa (CMMI, OpenUp e ITEC), de seguida é apresentada uma tabela com a descrição detalhada das *tarefas* e os detalhes das tarefas que foram levantadas tendo em consideração toda a informação trabalhada no Capítulo 3.

Lista das *tarefas* da ITEC

Para o preenchimento da Figura 36, é então criada uma lista com as *tarefas* desempenhadas pela ITEC, com as suas respetivas *Detailed tarefas*, onde estão divididas por áreas de atuação nos procedimentos diários da empresa. A informação contida na Figura 37, foi elaborada tendo em consideração todos os documentos fornecidos, reuniões e entrevistas estruturadas aos seus colaboradores.

TASK ITEC	DETAILED TASK
01 Emissão de ordem	011 Emite ordem de novo projeto
11 Recebe ordem	111 Recebe ordem de novo projeto
12 Planeia projeto	121 Documenta informação do projeto e caderno de encargos 122 Planeamento temporal do projeto
13 Transmissão de informação do projeto	131 Reunião kick off 132 Reunião ponto de situação e planeamento (semanal)
14 Escolha do programador	141 Planeamento tarefas e subdivisão
15 Orientação	151 Receção dos feedbacks de anomalia 152 Orientação para validar o SW 153 Orientação para testes elétricos e mecânicos
16 Validação	161 Verificação final 162 Aprovação para instalação no cliente
21 Recebe projeto	210 Pré ou pós Kick-off 211 Recebe novo projeto
22 Análise e feedback do projeto	2201 Reunião Kick-off 221 Análise das informações do projeto e caderno de encargos 222 Definição de parâmetros e objetivos funcionais 223 Análise da descrição de tarefas jira/trello 224 Reuniões de ponto de situação e planeamento (semanal)
23 Desenvolvimento	231 Esquematização do SW 232 Programação da máquina
24 Monitorização e desempenho	241 Feedback e levantamento de dificuldades 242 Atualização diária para o sistema de controlo de versões 243 Registo da taxa de realização 244 Registo de comentários dos motivos do atraso (se houver)
25 Validação	251 Validação do SW 252 Realização de teste elétricos e mecânicos 253 Retrabalho e revisão
26 Instalação no cliente final	261 Instalação do equipamento no cliente final
27 Monitorização e desempenho	271 Assistência técnica e suporte 272 Registo de reclamações 273 Solução de reclamações

Figura 37 - Lista de tarefas da ITEC

Deste modo, para haver uma melhor compreensão da importância das *tarefas* e *detailed tarefas*, são então criados níveis de cobertura, níveis de prioridade e nível de urgência do desempenho das mesmas, onde são explicados nos pontos seguintes.

Posto isto, em primeiro lugar são apresentados os níveis de cobertura da ITEC relativamente às tarefas levantadas.

Níveis de cobertura da ITEC

Para atribuir os níveis de cobertura do desempenho das *tarefas* desempenhadas pela ITEC, foram criados vários níveis, temos a vermelho com a letra W (*Without Coverage*) quando a atividade não é praticada por esta, a laranja e com a letra I quando a atividade é praticada mas não o satisfatoriamente (*Unsatisfactory Coverage*), a amarelo e com a letra S quando a atividade praticada é realizada satisfatoriamente (*Satisfactory Coverage*) e a verde e com a letra H quando a tarefa é desempenhada

com uma cobertura alta (*High Coverage*), dando assim origem à Tabela 5. Esta tabela, serve para podermos contextualizar o ponto em que a ITEC se encontra no momento da realização deste projeto.

Coverage	
<i>Not applicable</i>	N/A
<i>Without coverage</i>	W
<i>Unsatisfactory Coverage</i>	I
<i>Satisfactory Coverage</i>	S
<i>High coverage</i>	H

Tabela 5 - Níveis de cobertura das tarefas da ITEC

Níveis de Prioridade

De seguida são apresentados os níveis de prioridade com que as tarefas devem ser realizadas, em que consoante o nível de prioridade este vai ter impacto na qualidade do produto final.

De seguida, foi também criada a Tabela 6, com os níveis de prioridade onde que concluímos que seria o ideal para atribuir pesos às tarefas/atividades desempenhadas, tendo em consideração a qualidade do produto final. Posto isto, temos a vermelho e com as letras LP (*Low*) quando a atividade tem uma prioridade baixa, pois o produto pode ser entregue mesmo que a tarefa não seja realizada, a amarelo e com as letras MP (*Medium*) temos que a atividade tem uma prioridade média e neste caso quando a tarefa não é realizada traz impacto à qualidade do produto. Por último, temos a verde e com as letras HP (*High*) em que todas as atividades precisam de ser realizadas, pois são de prioridade alta e sem efetuar esta tarefa não conseguem que o produto esteja operacional para ser entregue ao cliente.

Priority	
<i>Low</i>	LP
<i>Medium</i>	MP
<i>High</i>	HP

Tabela 6 - Níveis de Prioridade tarefas da ITEC

Por último, surge então o nível de urgência da execução das tarefas, com origem na relação entre as duas anteriores e onde é explicada a sua importância.

Nível de urgência da execução da tarefa

Como já referido anteriormente, é utilizada a informação referente à Tabela 5 e à Tabela 6, cuja relação entre as duas deu origem ao nível de urgência da execução das tarefas, como podemos observar na Tabela 7. Relativamente às cores presentes na tabela temos a vermelho quando a tarefa tem de ser realizada com muita urgência, pois neste caso tem uma importância acrescida na entrega do produto final. No entanto, o nível mais baixo está representado com a verde, na qual não traz grande impacto para a entrega do produto final.

Extremely urgent				WHP
Urgent			WMP	IHP
Do ASAP		WLP	IMP	SHP
To Do		ILP	SMP	
Do if possible		SLP		
OK	HNP, SNP, INP, WNP	HLP	HMP	HHP

Tabela 7 - Níveis de urgência de realização das tarefas da ITEC

Em forma de exemplo, podemos observar na Figura 38, o preenchimento da parte direita da imagem, em que são utilizadas as informações apresentadas na lista de *tarefas* da ITEC e os níveis de cobertura, prioridade e de urgência. Todavia, o resultado desta relação é utilizado para o mapeamento em *Neo4j* de todas as interações realizadas e que são apresentadas de seguida.

Doing		CMMI/OpenUp						
Ensuring Quality (ENQ)								
Verification and Validation (VV)								
Level		Practice: Summary		Tasks ITEC	Detailed tasks	Coverage	Priority	Action is urgent?
ML1	PI 1.1 Assemble solutions and deliver to the customer.	26		261		H	HP	HHP
ML1	VV 1.2 Perform validation to ensure the solution will function as intended in its target environment and record and communicate results.	25		251, 252, 253		S	HP	SHP
ML2	VV 2.1 Select components and methods for verification and validation.	15, 22, 24		152, 153, 223, 241, 242, 244		S	MP	SMP
ML2	VV 2.2 Develop, keep updated, and use the environment needed to support verification and validation.	16, 22, 23, 24, 25		161, 162, 201, 221, 222, 223, 224, 231, 232, 242, 243, 244, 251, 252, 253		I	MP	IMP
ML2	VV 2.3 Develop, keep updated, and follow procedures for verification and validation.	15, 16, 22, 23, 24, 25, 26		152, 153, 162, 162, 220, 221, 222, 223, 224, 231, 242, 243, 244, 251, 252, 253, 261,		S	MP	SMP
								Notas

Figura 38 - Exemplo de preenchimento Tabela

Após apresentados todos os procedimentos e informações recolhidas, passamos agora então para a apresentação do mapeamento em que são apresentadas todas as informações que achamos que são importantes para o estudo.

Contudo, nesta fase do trabalho as *Categorias*, já apresentadas anteriormente, são explicadas e detalhadas em separado, de modo a facilitar a compreensão do mesmo. Desse modo, são apresentados os mapeamentos efetuados recorrendo ao *Neo4j* e às tabelas com as informações importantes sobre o assunto em estudo. Deste modo, começamos a demonstração do nosso estudo, com a *Category "Doing" do CMMI*, relacionando esta com as tarefas do OpenUp e da ITEC e de seguida o mesmo é feito para as restantes *Categorias* do CMMI.

Doing – Mapeamento das *Capability Areas (CMMI)*, *Tarefas Open Up* e *ITEC*

A *Category "Doing"* está dividida em quatro *Capability Areas*, conhecidas por "*Delivering & Managing Service (DMS)*", "*Engineering & Developing Products (EDP)*", "*Ensuring Quality (ENQ)*" e "*Selecting & Managing Suppliers (SMS)*", no entanto a *Capability Area SMS*, não é representada nesta demonstração de resultados, pois chegamos à conclusão não estar enquadrada com o pretendido no projeto em estudo. Assim sendo, passamos então para apresentação dos resultados obtidos na *Capability Area "Delivery & Managing Service (DMS)"*, tendo em consideração a informação recolhida.

Capability Area “Delivering & Managing Service (DMS)” da Category “Doing”

Relativamente à *Capability Area* “Delivering & Managing Service (DMS)”, está dividida em duas *Practice Areas*, a “Service Delivery Management (SDM)” e “Strategic Service Management (STSM)”. Deste modo, é utilizado então o *Neo4j* como ferramenta para o mapeamento dos dados, sendo que na Figura 39, podemos observar a castanho-escuro as áreas de atuação das disciplinas do *OpenUp*, a azul-escuro as *tarefas* realizadas, a castanho-claro temos as *Practice Summary* referentes ao CMMI, a cor de rosa temos os nomes das duas *Practice Areas* (SDM e STSM), a roxo temos a *Capability Area* em cima mencionada, amarelo temos a categoria “Doing”, a laranja temos os níveis de maturidade referentes ao CMMI e a verde-escuro temos o nodo referente ao CMMI.

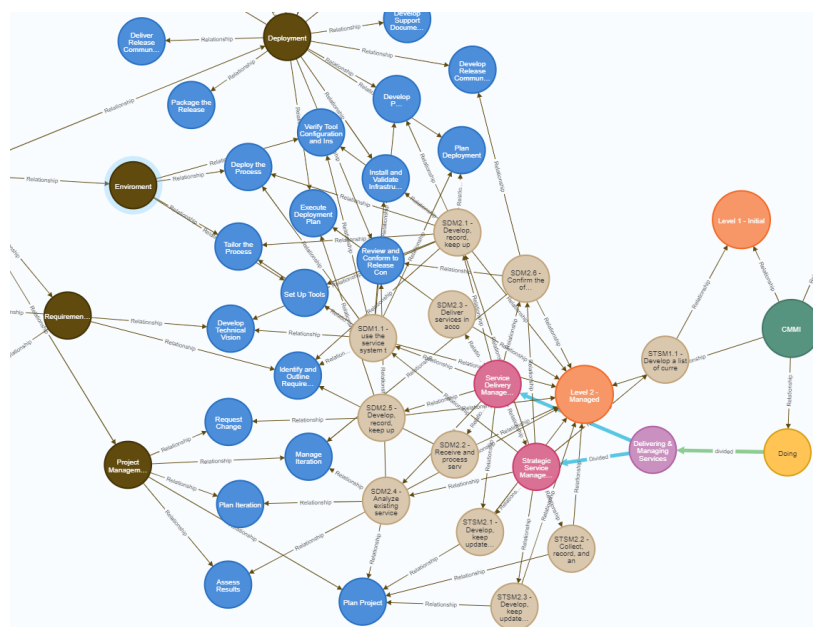


Figura 39 – Mapeamento da Capability Area DMS, Tarefas OpenUp e Tarefas da ITEC

Podemos observar na Tabela 8, o cruzamento realizado tendo em consideração a *Practice Area* “Service Delivery Management”, em que são apresentados quais os níveis de prioridade, com que cada atividade deve ser realizada tendo em consideração as *tarefas* do *OpenUp*, o mesmo acontece relativamente às relações entre *Practice Summary* (CMMI) e as *tarefas* da ITEC. Contudo, também é possível determinar qual o nível de cobertura da ITEC face às *Practice Summary* apresentadas, para além disso, é possível determinar o nível de prioridade que essa tarefa tem e qual o nível de urgência para que seja realizada, com objetivo que o produto seja entregue com a qualidade necessária e de acordo como solicitado na fase inicial.

Deste modo, o mesmo é realizado para a *Practice Area “Strategic Service Management”*, porém neste caso esta área de atuação não existe na ITEC, portanto não traz informação relevante para a realização deste projeto. Em forma de resumo, a *Practice Area “Service Delivery Management”*, tem como objetivo ter um sistema de prestação de serviços, onde esteja a informação sempre atualizada, ter os processos em conformidade com o que foi contratado, é necessário realizar uma recolha e análise dos dados para que haja uma boa estratégia.

Doing	Delivering & Managing Services (DMS)	Service Delivery Management (SDM)	CMMI						
			ML1			ML2			
			SDM 1.1 Use the service system to deliver services.	SDM 2.1 Develop, record, keep updated, and follow service agreements.	SDM 2.2 Receive and process service requests in accordance with service agreements.	SDM 2.3 Deliver services in accordance with service agreements.	SDM 2.4 Analyze existing service agreements and service data to prepare for updated or new agreements.	SDM 2.5 Develop, record, keep updated, and follow the approach for operating and changing the service system.	SDM 2.6 Confirm the readiness of the service system to support the delivery of services.
Tasks OpenUp									
		Envision the Architecture	H					H	
		Assess Results						H	
		Manage Iteration						H	H
		Plan Iteration						H	
		Plan Project	M					H	
		Request Change							H
		Identify and Outline Requirements		H	H				
		Detail Use-Case Scenarios		H	H				
		Detail System-Wide Requirements		H	H				
		Develop Technical Vision		H					
		Implement Solution	H						
		Deploy the Process		H					
		Tailor the Process							
		Set Up Tools		M					
		Verify Tool Configuration and Installation		M					
		Execute Deployment Plan				H			H
		Package the Release							
		Verify Successful Deployment			H	H	H	H	H
		Develop Backout Plan		H					
		Develop Release Communications							H
		Install and Validate Infrastructure	H	H					
		Plan Deployment		H					
		Review and Conform to Release Controls					H		H
Doing	Delivering & Managing Services (DMS)	Service Delivery Management (SDM)	CMMI						
			ML1			ML2			
			SDM 1.1 Use the service system to deliver services.	SDM 2.1 Develop, record, keep updated, and follow service agreements.	SDM 2.2 Receive and process service requests in accordance with service agreements.	SDM 2.3 Deliver services in accordance with service agreements.	SDM 2.4 Analyze existing service agreements and service data to prepare for updated or new agreements.	SDM 2.5 Develop, record, keep updated, and follow the approach for operating and changing the service system.	SDM 2.6 Confirm the readiness of the service system to support the delivery of services.
Tasks ITEC									
		011 Emitte ordem de novo projeto	x	x	x				
		111 Recebe ordem de novo projeto	x	x	x				
		121 Documenta informação do projeto e caderno de encargos	x	x	x				
		122 Planeamento temporal do projeto	x	x	x				
		131 Reunião kick off	x		x				
		132 Reunião ponto de situação e planeamento (semanal)	x		x				
		141 Planeamento tarefas e subdivisão	x		x				
		151 Receção dos feedbacks de anomalia	x						
		152 Orientação para validar o SW	x			x			
		153 Orientação para testes elétricos e mecânicos	x			x			
		161 Verificação final	x			x			x
		162 Aprovação para instalação no cliente	x			x			x
		210 Pré ou pós Kick-off	x						
		211 Recebe novo projeto	x						
		2201 Reunião Kick-off	x		x				
		221 Análise das informações do projeto e caderno de encargos	x		x				
		222 Definição de parâmetros e objetivos funcionais	x		x				
		223 Análise da descrição de tarefas jira/trello	x		x				
		224 Reuniões de ponto de situação e planeamento (semanal)	x						
		231 Esquemáticação do Sw	x						
		232 Programação da máquina	x						
		241 Feedback e levantamento de dificuldades	x	x				x	
		242 Atualização diária para o sistema de controlo de versões	x	x				x	
		243 Registo da taxa de realização	x	x				x	
		244 Registo de comentários dos motivos do atraso (se houver)	x	x				x	
		251 Validação do SW	x			x			
		252 Realização de teste elétricos e mecânicos	x			x			
		253 Retrabalho e revisão	x			x			
		261 Instalação do equipamento no cliente final	x						
		271 Assistência técnica e suporte	x		x				
		272 Registo de reclamações	x		x				
		273 Solução de reclamações	x		x				
		Coverage	H	I	S	S	IN	I	I
		Priority	HP	HP	HP	HP	MP	MP	MP
		Action is urgent?	HHP	IHP	SHP	SHP	WNP	IMP	IMP

Tabela 8 - Cruzamento CA "SDM", Tarefas OpenUp e Tarefas da ITEC

Depois de realizada esta demonstração, passamos agora para apresentação dos resultados obtidos na *Capability Area “Engineering & Developing Products (EDP)”*, tendo em consideração a informação recolhida ao longo do projeto.

Capability Area “Engineering & Developing Products” da Category “Doing”

Ainda relativamente à *Category* “Doing”, temos a *Capability Area* “Engineering & Developing Products (EDP)”, em que está dividida em duas *Practice Areas*, a “*Product Integration (PI)*” e “*Technical Solution (TS)*”.

Desse modo, é realizado um mapeamento tendo em consideração toda a informação relativa ao *OpenUp*, CMMI e à ITEC, como podemos observar na Figura 40, a verde claro o nodo referente ao *OpenUp*, a castanho-escuro as áreas de atuação das *disciplinas* do *OpenUp*, a azul-escuro as *tarefas* realizadas, a castanho-claro temos as *Practice Summary* referentes ao CMMI, a cor de rosa temos os nomes das *Practice Areas (PI e TS)*, a roxo temos a *Capability Area* em cima mencionada, amarelo temos a categoria “*Doing*”, a laranja temos os níveis de maturidade referentes ao CMMI e a verde-escuro temos o nodo referente ao CMMI.

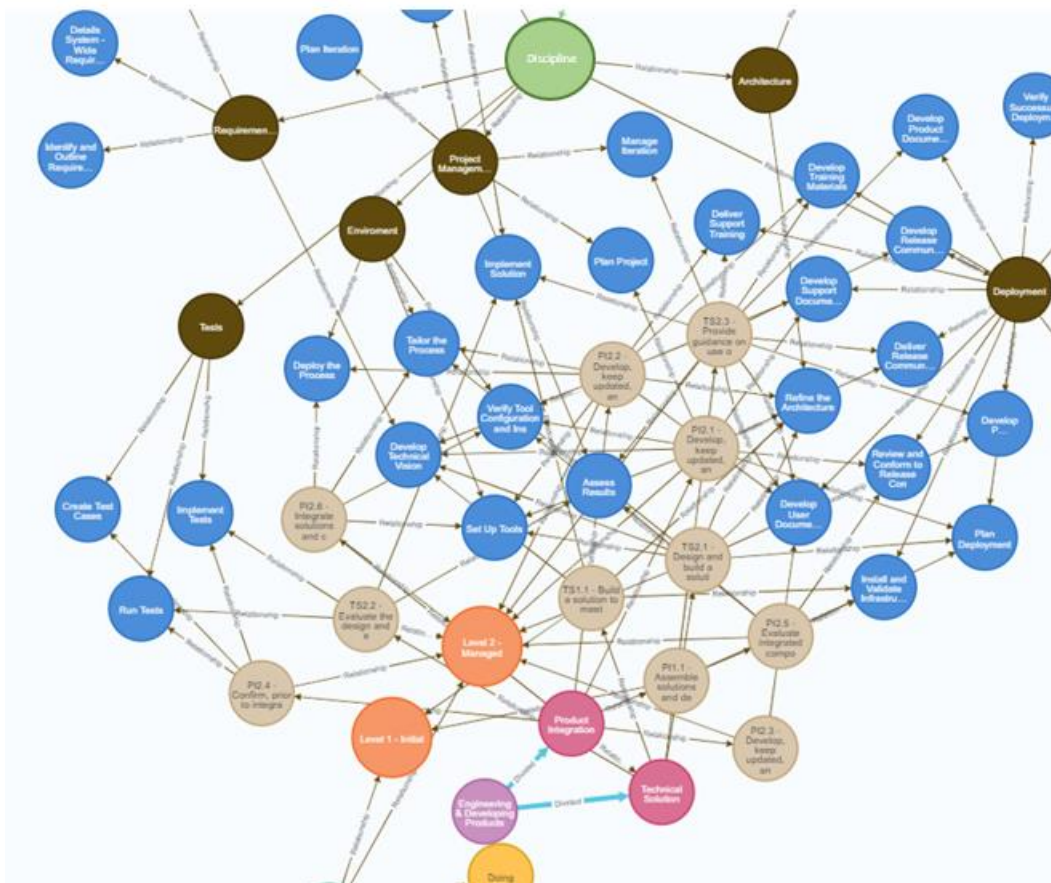


Figura 40 – Mapeamento da Capability Area EDP, Tarefas OpenUp e Tarefas da ITEC

Deste modo, podemos observar no Apêndice II, o cruzamento realizado tendo em consideração a *Practice Area* “*Product Integration*”, em que são apresentados quais os níveis de prioridade, que cada

atividade deve ser realizada tendo em consideração as *tarefas* do *OpenUp*, o mesmo acontece relativamente as relações entre *Practice Summary* (CMMI) e as *tarefas* da ITEC. Contudo, também é possível determinar qual o nível de cobertura da ITEC face às *Practice Summary* apresentadas, para além disso, é possível determinar o nível de prioridade que essa tarefa tem e qual o nível de urgência para que seja realizada, com objetivo que o produto seja entregue com a qualidade necessária e de acordo como solicitado na fase inicial. Porém, o mesmo é realizado para a *Practice Area* “*Technical Solution*”, como podemos observar no Apêndice III.

Estas duas *Practice Area* têm como objetivo entregar ao cliente o que ele necessita, em que vai ao encontro aos requisitos levantados no início do projeto. Deste modo, é necessário seguir um procedimento constante, atualizar os códigos e os seus comentários diariamente e encontrar soluções para as dificuldades encontradas, com o objetivo de entregar um produto final com a melhor qualidade possível.

Assim sendo, passamos agora para apresentação dos resultados obtidos na *Capability Area* “*Ensuring Quality (ENQ)*”, tendo em consideração a informação recolhida ao longo do projeto.

Capability Area “Ensuring Quality (ENQ)” da Category “Doing”

Relativamente à *Capability Area* “*Ensuring Quality (ENQ)*” esta dividida em quatro *Practice Area*, a “*Peer Review (PR)*”, “*Process Quality Assurance (PQA)*”, “*Requirements Development & Management (RDM)*” e “*Verification & Validation (VV)*”. Desse modo, é realizado um mapeamento tendo em consideração toda a informação relativa ao *OpenUp*, CMMI e à ITEC, como podemos observar na Figura 41, a verde claro o nodo referente ao *OpenUp*, a castanho-escuro as áreas de atuação das disciplinas do *OpenUp*, a azul-escuro as *tarefas* realizadas, a castanho-claro temos as *Practice Summary* referentes ao CMMI, a cor de rosa temos os nomes das *Practice Areas (PR, PQA, RDM e VV)*, a roxo temos a *Capability Area* em cima mencionada, amarelo temos a categoria “*Doing*”, a laranja temos os níveis de maturidade referentes ao CMMI.

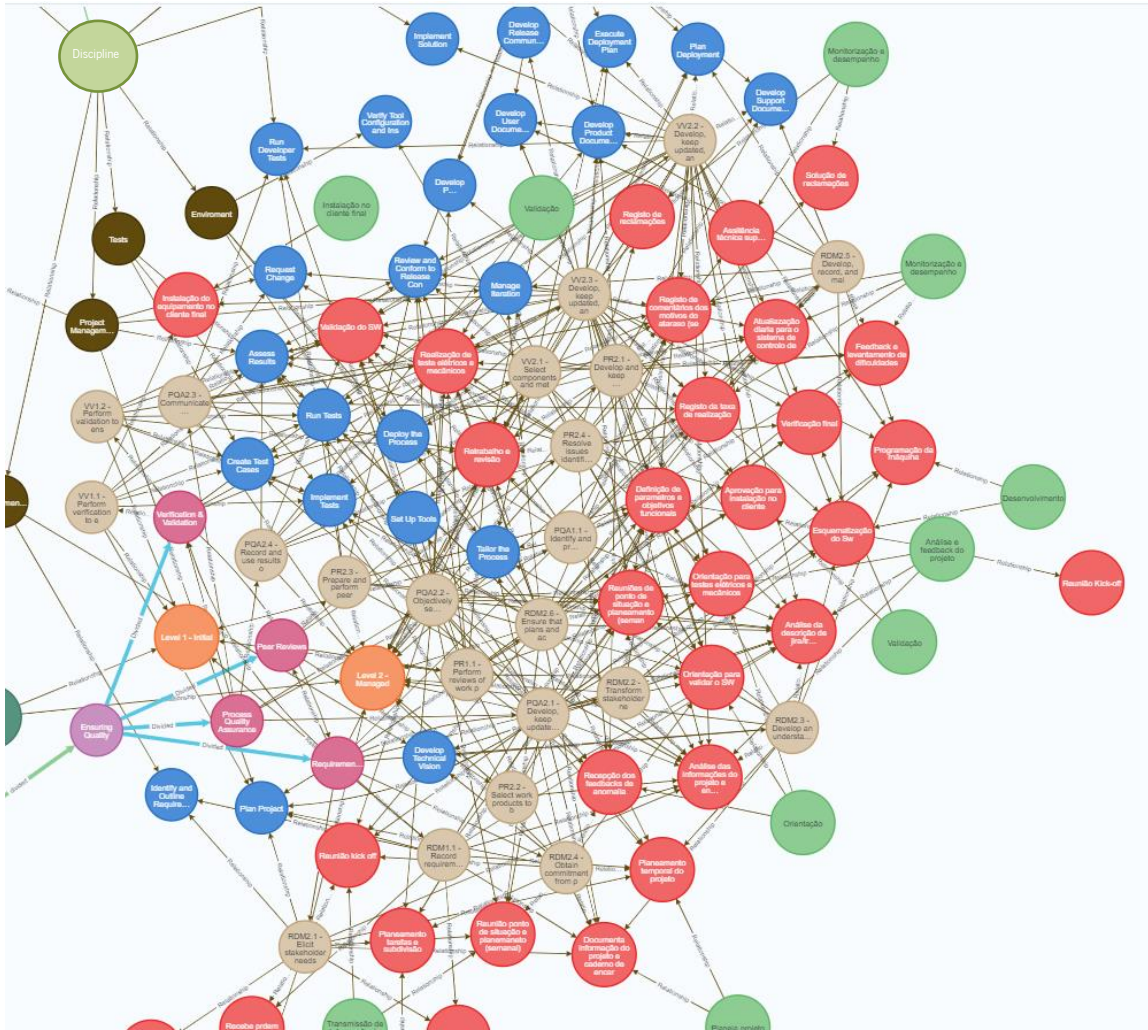


Figura 41 – Mapeamento da Capability Area ENQ, Tarefas OpenUp e Tarefas da ITEC

Desta forma, podemos observar no Apêndice IV, o cruzamento realizado tendo em consideração a *Practice Area* “Peer Review”, em que são apresentados quais os níveis de prioridade, que cada atividade deve ser realizada tendo em consideração as *tarefas* do *OpenUp*, o mesmo acontece relativamente as relações entre *Practice Summary* (CMMI) e as *tarefas* da ITEC. Contudo, também é possível determinar qual o nível de cobertura da ITEC face às *Practice Summary* apresentadas, para além disso, é possível determinar o nível de prioridade que essa tarefa tem e qual o nível de urgência para que seja realizada, com objetivo que o produto seja entregue com a qualidade necessária e de acordo como solicitado na fase inicial.

Contudo, o mesmo é realizado para as restantes *Practice Areas*, em que o Apêndice V é referente à *Practice Area* “Process Quality Assurance”, o Apêndice VI referente à “Requirements Development & Management” e o Apêndice VII relativo à “Verification & Validation”.

Estas quatro áreas de atuação têm como objetivo realizar levantamento dos requisitos, assegurar que o trabalho está a ser realizado de acordo com esses requisitos levantados, fazer uma avaliação da qualidade do produto, para facilitar na validação do mesmo, para que o produto seja entregue ao cliente com a máxima qualidade possível. Deste modo, terminamos a análise relativa à *Category “Doing”* e passamos para a análise e demonstração de resultados da *Category “Managing”*, que são apresentados no ponto seguinte.

Managing -Mapeamento das Capability Areas (CMMI), Tarefas Open Up e tarefas da empresa

A *Category “Managing”* está dividida em três *Capability Areas*, conhecidas por “*Managing Business Resilience (MBR)*”, a “*Managing the Workforce (MWF)*” e “*Planning & Managing Work (PMW)*”. Assim sendo, passamos então para apresentação dos resultados obtidos na *Capability Area “Managing Business Resilience (MBR)”*, tendo em consideração a informação recolhida ao longo de todo o projeto.

Capability Area “Managing Business Resilience” da Category “Managing”

Relativamente à *Capability Area “Managing Business Resilience (MBR)”* esta dividida em três *Practice Areas*, a “*Continuity (CONT)*”, “*Incident Resolution & Prevention (IRP)*” e “*Risk & Opportunity Management (RSK)*”. Desse modo, é realizado um mapeamento tendo em consideração toda a informação relativa ao *OpenUp*, *CMMI* e à *ITEC*, como podemos observar na Figura 42, a verde claro os referentes as tarefas da *ITEC*, a castanho-escuro as áreas de atuação das disciplinas do *OpenUp*, a azul-escuro as *tarefas* realizadas, a castanho-claro temos as *Practice Summary* referentes ao *CMMI*, a cor de rosa temos os nomes das *Practice Areas (CONT, IRP e RSK)*, a roxo temos a *Capability Area* em cima mencionada, amarelo temos a categoria “*Managing*”, a laranja temos os níveis de maturidade referentes ao *CMMI* e a verde-escuro temos o nodo referente ao *CMMI*.

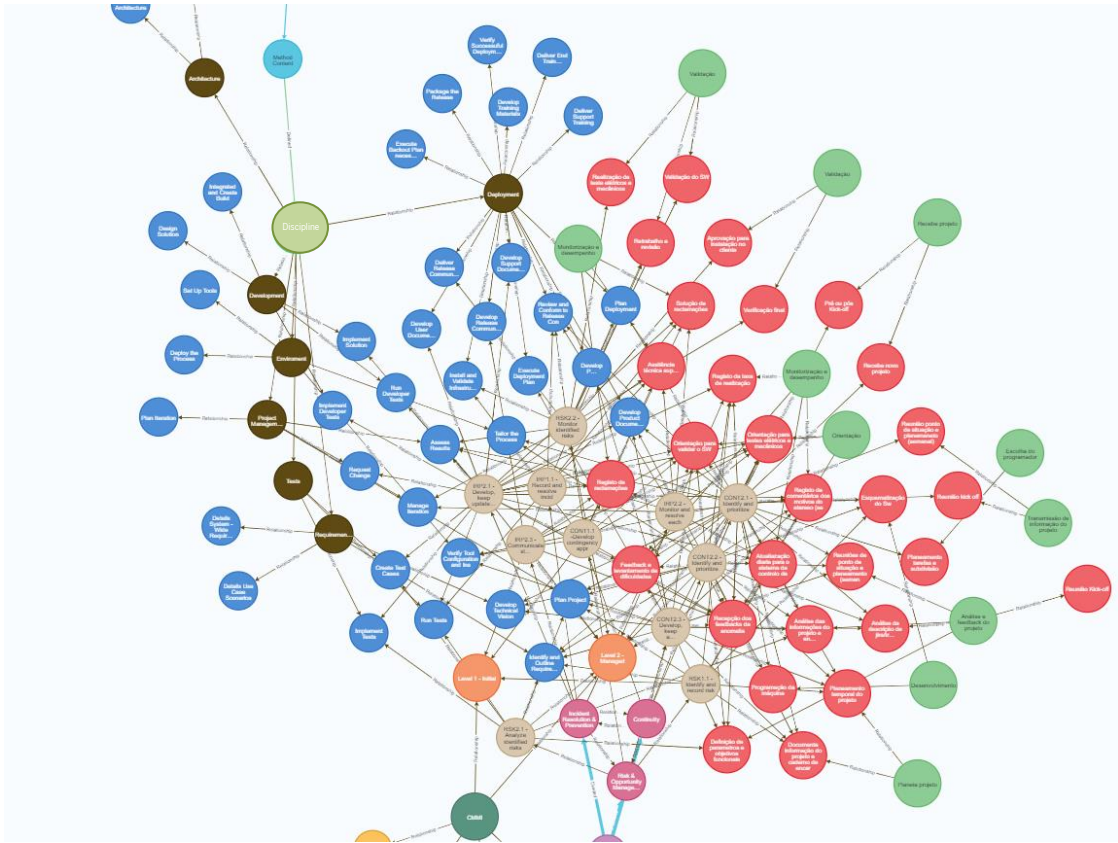


Figura 42 – Mapeamento da Capability Area MBR, Tarefas OpenUp e Tarefas da ITEC

Assim sendo, podemos observar no Apêndice VIII, o cruzamento realizado tendo em consideração a *Practice Area* “*Continualty*”, em que são apresentados quais os níveis de prioridade, que cada atividade deve ser realizada tendo em consideração as *tarefas* do OpenUp, o mesmo acontece relativamente as relações entre Practice Summary (CMMI) e as tarefas da ITEC. Contudo, também é possível determinar qual o nível de cobertura da ITEC face às *Practice Summary* apresentadas, para além disso, é possível determinar o nível de prioridade que essa tarefa tem e qual o nível de urgência para que seja realizada, com objetivo que o produto seja entregue com a qualidade necessária e de acordo como solicitado na fase inicial.

No entanto, o mesmo é feito para as ressaltantes *Practice Areas*, em que o Apêndice IX é referente à *Practice Area* “*Incident Resolution & Prevention*” e o Apêndice X referente à *Practice Area* “*Risk & Opportunity Management*”.

Estas três *Practice Areas* têm como objetivo, determinar os problemas detetados para dar continuidade aos procedimentos, identificar e dar prioridades às funções para haver essa mesma continuidade, tem também como finalidade registar todos os incidentes e comunicá-los, analisar os riscos

do projeto em que os clientes devem ser sempre informados desse mesmo risco. Assim sendo, passamos agora para apresentação dos resultados obtidos na *Capability Area* “*Managing the Workforce (MWF)*”, tendo em consideração a informação recolhida.

Capability Area “Managing the Workforce” da Category “Managing”

Relativamente à *Capability Area* “*Managing the Workforce (MWF)*” esta dividida em duas *Practice Areas*, a “*Enabling Virtual Solution Delivery (EVSD)*” e “*Organizational Training (OT)*”. Desse modo, é realizado um mapeamento tendo em consideração toda a informação relativa ao *OpenUp*, *CMMI* e à ITEC, como podemos observar na Figura 43, a castanho-escuro as áreas de atuação das *disciplinas* do *OpenUp*, a azul-escuro as *tarefas* realizadas, a castanho-claro temos as *Practice Summary* referentes ao *CMMI*, a cor de rosa temos os nomes das *Practice Areas (EVSD e OT)*, a roxo temos a *Capability Area* em cima mencionada, amarelo temos a categoria “*Managing*”, a laranja temos os níveis de maturidade referentes ao *CMMI* e a verde-escuro temos o nodo referente ao *CMMI*.

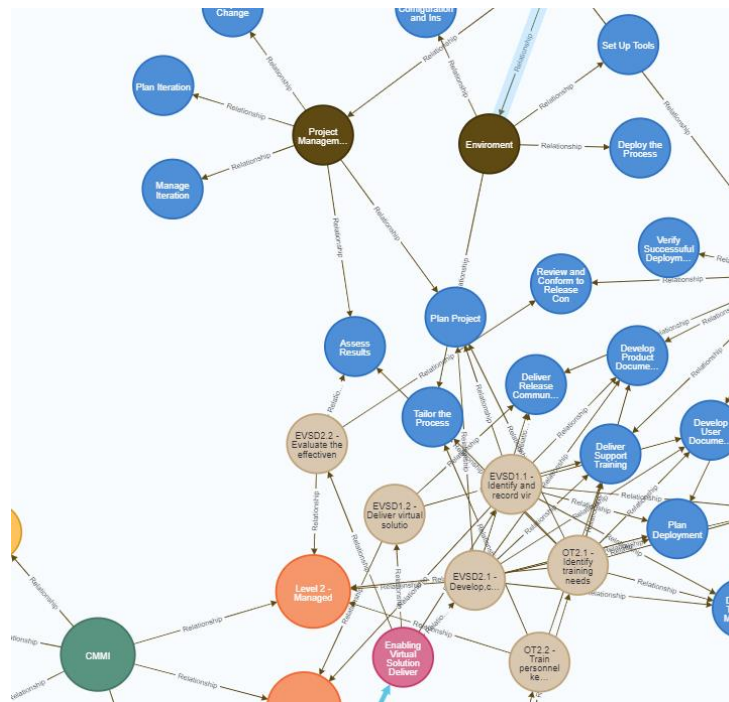


Figura 43 – Mapeamento das Capability Area MWF, Tarefas OpenUp e Tarefas da ITEC

Contudo, podemos observar no Apêndice XI, o cruzamento realizado tendo em consideração a *Practice Area* “*Enabling Virtual Solution Delivery*”, em que são apresentados quais os níveis de prioridade,

que cada atividade deve ser realizada tendo em consideração as *tarefas* do *OpenUp*, o mesmo acontece relativamente as relações entre *Practice Summary (CMMI)* e as *tarefas* da ITEC. Contudo, também é possível determinar qual o nível de cobertura da ITEC face às *Practice Summary* apresentadas, para além disso, é possível determinar o nível de prioridade que essa tarefa tem e qual o nível de urgência para que seja realizada, com objetivo que o produto seja entregue com a qualidade necessária e de acordo como solicitado na fase inicial. Da mesma forma, é realizado para a *Practice Area*, em que o Apêndice XII é referente à *Practice Area “Organizational Training”*.

Estas duas áreas de atuação têm como objetivo identificar e registar necessidades, fornecer soluções, desenvolver e comunicar as necessidades e as restrições encontradas, avaliar a eficácia do produto, treinar pessoas, identificar as necessidades de formação e manter um registo dessa mesma formação. Assim sendo, passamos agora para apresentação dos resultados obtidos na *Capability Area “Planning & Managing Work (PMW)”*, tendo em consideração a informação recolhida.

Capability Area “Planning & Managing Work (PMW)” da Category “Managing”

Relativamente à *Capability Area “Planning & Managing Work (PMW)”* esta dividida em três *Practice Areas*, a *“Estimating (EST)”*, *“Monitor & Control (MC)”* e *“Planning (PLAN)”*. Desse modo, é realizado um mapeamento tendo em consideração toda a informação relativa ao *OpenUp*, CMMI e à ITEC, como podemos observar na Figura 44, a verde claro o nodo referente ao *OpenUp*, a castanho-escuro as áreas de atuação das disciplinas do *OpenUp*, a azul-escuro as *tarefas* realizadas, a castanho-claro temos as *Practice Summary* referentes ao CMMI, a cor de rosa temos os nomes das *Practice Areas (EST, MC e PLAN)*, a roxo temos a *Capability Area* em cima mencionada, amarelo temos a categoria *“Managing”*, a laranja temos os níveis de maturidade referentes ao CMMI e a verde-escuro temos o nodo referente ao CMMI.

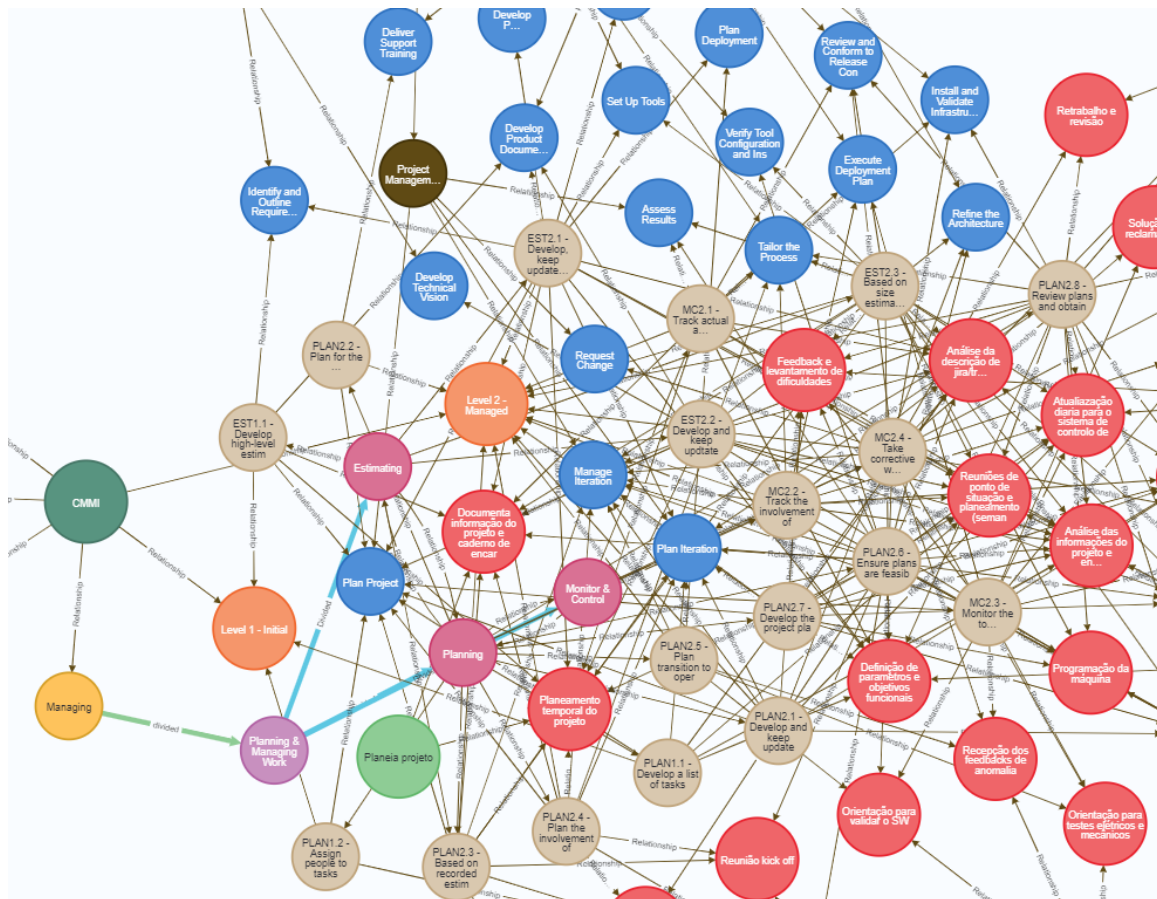


Figura 44 – Mapeamento das Capability Area PMW, Tarefas OpenUp e Tarefas da ITEC

No entanto, podemos observar no Apêndice XIII, o cruzamento realizado tendo em consideração a *Practice Area* “*Estimating*”, em que são apresentados quais os níveis de prioridade, que cada atividade deve ser realizada tendo em consideração as *tarefas* do *OpenUp*, o mesmo acontece relativamente as relações entre *Practice Summary* (CMMI) e as *tarefas* da ITEC. Contudo, também é possível determinar qual o nível de cobertura da ITEC face às *Practice Summary* apresentadas, para além disso, é possível determinar o nível de prioridade que essa tarefa tem e qual o nível de urgência para que seja realizada, com objetivo que o produto seja entregue com a qualidade necessária e de acordo como solicitado na fase inicial.

Contudo, o mesmo é realizado para as ressaltantes *Practice Areas*, em que o Apêndice XIV é referente à *Practice Area* “*Monitor & Control*” e o Apêndice XV referente à *Practice Area* “*Planning*”.

Estas três *Practice Areas*, têm como objetivo desenvolver estimativas para o desenvolvimento do trabalho, ter os documentos sempre atualizados, estimar custos e duração do projeto, fazer registos constantes do que esta a desenvolver, fazer um registo da conclusão da tarefa, identificar problemas,

fazer um acompanhamento constante do que esta a ser realizado, fazer um rastreio do envolvimento de todas as partes interessadas, elaborar uma lista de tarefas, atribuir essas tarefas a uma pessoa ou mais pessoas, fazer um planeamento do projeto e haver apoio constante. Deste modo, terminamos a análise relativa à *Category “Managing”* e passamos para a análise e demonstração de resultados da *Category “Enabling”*, que são apresentados no ponto seguinte.

Enabling - Mapeamento *Capability Areas* (CMMI), *Tarefas Open Up* e *tarefas da ITEC*

A *Category “Enabling”* está dividida em duas *Capability Areas*, conhecidas por *“Managing Security and Safety (MSS)”* e *“Supporting Implementation (SI)”*. Assim sendo, passamos então para apresentação dos resultados obtidos na *Capability Area “Delivery & Managing Service (DMS)”*, tendo em consideração a informação recolhida.

Capability Area “Managing Security and Safety (MSS)” da Category “Enabling”

Relativamente à *Capability Area “Managing Security and Safety (MSS)”* está dividida em três *Practice Areas*, a *“Enabling Safety (ESAF)”*, *“Enability Security (ESEC)”* e *“Managing Security Threats & Vulnerabilities (MST)”*. Assim sendo, é realizado um mapeamento tendo em consideração toda a informação relativa ao *OpenUp*, *CMMI* e à *ITEC*, como podemos observar na Figura 45, a verde claro o nodo referente ao *OpenUp*, a castanho-escuro as áreas de atuação das *tarefas* do *OpenUp*, a azul-escuro as *tarefas* realizadas, a castanho-claro temos as *Practice Summary* referentes ao *CMMI*, a cor de rosa temos os nomes das *Practice Areas (ESAF, ESEC e MST)*, a roxo temos a *Capability Area* em cima mencionada, amarelo temos a categoria *“Enabling”*, a laranja temos os níveis de maturidade referentes ao *CMMI* e a verde-escuro temos o nodo referente ao *CMMI*.

obtidos na *Capability Area* “*Supporting Implementation (SI)*”, tendo em consideração a informação recolhida.

Capability Area “Supporting Implementation (SI)” da Category “Enabling”

Relativamente à *Capability Area* “*Supporting Implementation*” está dividida em três *Practice Areas*, a “*Casual Analysis & Resolution (CAR)*”, “*Configuration Management (CM)*” e “*Decision Analysis & Resolution (DAR)*”. Desta forma, é realizado um mapeamento tendo em consideração toda a informação relativa ao *OpenUp*, *CMMI* e à *ITEC*, como podemos observar na Figura 46, a verde claro o nodo referente ao *OpenUp*, a castanho-escuro as áreas de atuação das *tarefas* do *OpenUp*, a azul-escuro as *tarefas* realizadas, a castanho-claro temos as *Practice Summary* referentes ao *CMMI*, a cor de rosa temos os nomes das *Practice Areas (CAR,CM e DAR)* , a roxo temos a *Capability Area* em cima mencionada, amarelo temos a categoria “*Enabling*”, a laranja temos os níveis de maturidade referentes ao *CMMI* e a verde-escuro temos o nodo referente ao *CMMI*.

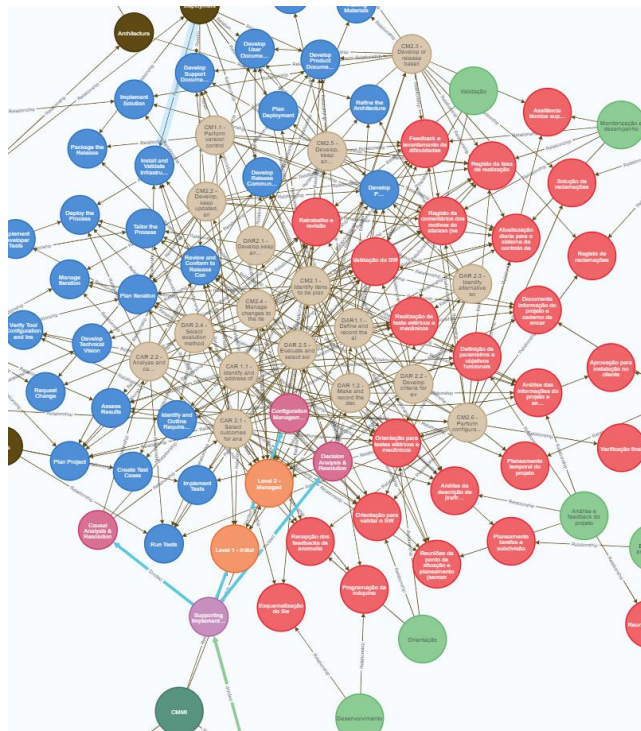


Figura 46 – Mapeamento das *Capability Area SI*, *Tarefas OpenUp* e *Tarefas da ITEC*

Contudo, podemos observar no Apêndice XVIII, o cruzamento realizado tendo em consideração a *Practice Area* “*Causal Analysis & Resolution*”, em que são apresentados quais os níveis de prioridade, que cada atividade deve ser realizada tendo em consideração as *tarefas* do *OpenUp*, o mesmo acontece

relativamente as relações entre *Practice Summary* (CMMI) e as tarefas da ITEC. Contudo, também é possível determinar qual o nível de cobertura da ITEC face às *Practice Summary* apresentadas, para além disso, é possível determinar o nível de prioridade que essa tarefa tem e qual o nível de urgência para que seja realizada, com objetivo que o produto seja entregue com a qualidade necessária e de acordo como solicitado na fase inicial. Assim sendo, o mesmo foi realizado para as ressaltantes *Practice Areas*, em que o Apêndice XIX é referente à *Practice Area “Configuration Management”* e o Apêndice XX referente à *Practice Area “Decision Analysis & Resolution”*.

Estas três áreas de atuação têm como objetivo, identificar e abordar os resultados obtidos, fazer uma seleção dos resultados para análise, efetuar um controlo de versões onde precisa de estar constantemente atualizado e para tal é necessário utilizar um software que responda a estas necessidades, fazer uma gestão do sistema utilizado para o controlo de versões, realizar auditorias, definir e registar alternativas, tomar e registar decisões, identificar soluções, selecionar métodos de avaliação e por fim avaliar e selecionar soluções utilizando critérios e métodos. Deste modo, terminamos a análise relativa à *Category “Enabling”* e passamos para a análise e demonstração de resultados da *Category “Improving”*, que são apresentados no ponto seguinte.

Improving - Mapeamento Capability Areas (CMMI), Tarefas Open Up e tarefas da ITEC

A *Category “Improving”* está dividida em duas *Capability Areas*, conhecidas por *“Improving Performance (IMP)”* e *“Sustaining Habit and Persistence (SHP)”*. Assim sendo, passamos então para apresentação dos resultados obtidos na *Capability Area “Improving Performance”*, tendo em consideração a informação recolhida.

Capability Area “Improving Performance” da Category “Improving”

Relativamente à *Capability Area “Improving Performance (IMP)”* está dividida em três *Practice Areas*, a *“Managing Performance & Measurement (MPM)”*, *“Process Asset Development (PAD)”* e *“Process Management (PCM)”*. Assim sendo, é realizado um mapeamento tendo em consideração toda a informação relativa ao OpenUp, CMMI e à ITEC, como podemos observar na Figura 47, a verde claro o nodo referente ao OpenUp, a castanho-escuro as áreas de atuação das *tarefas* do *OpenUp*, a azul-escuro as *tarefas* realizadas, a castanho-claro temos as *Practice Summary* referentes ao CMMI, a cor de

rosa temos os nomes das *Practice Areas* (MPM, PAD e PCM), a roxo temos a *Capability Area* em cima mencionada, amarelo temos a categoria “*Improving*”, a laranja temos os níveis de maturidade referentes ao CMMI e a verde-escuro temos o nodo referente ao CMMI.

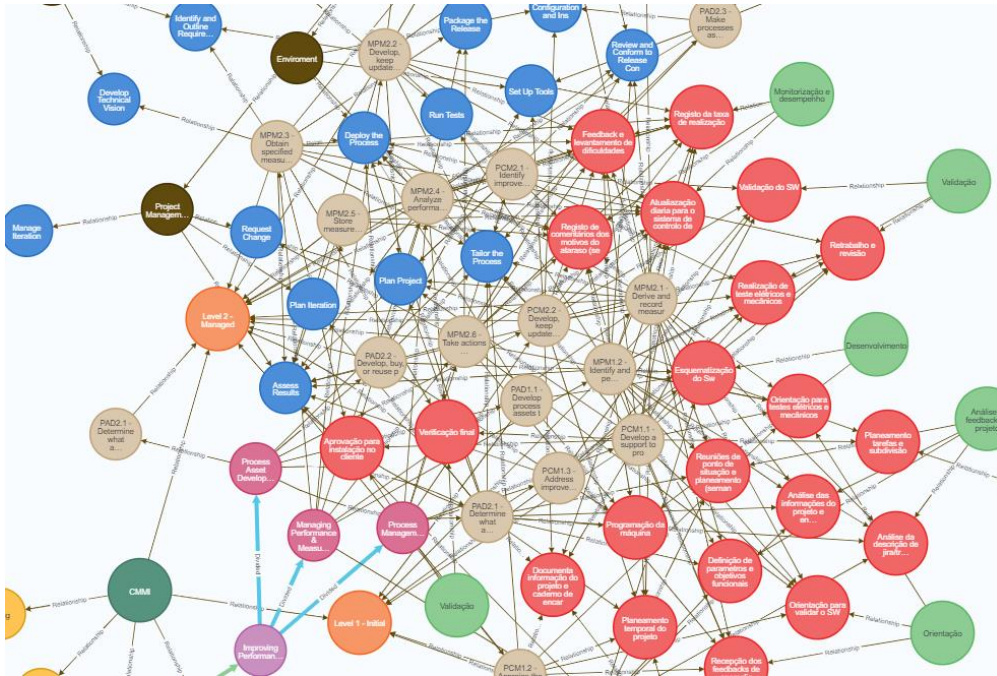


Figura 47 – Mapeamento das Capability Area IMP, Tarefas OpenUp e Tarefas da ITEC

Porém, podemos observar no Apêndice XXI, o cruzamento realizado tendo em consideração a *Practice Area* “*Managing Performance & Measurement*”, em que são apresentados quais os níveis de prioridade, que cada atividade deve ser realizada tendo em consideração as *tarefas* do *OpenUp*, o mesmo acontece relativamente as relações entre *Practice Summary* (CMMI) e as tarefas da ITEC. Contudo, também é possível determinar qual o nível de cobertura da ITEC face às *Practice Summary* apresentadas, para além disso, é possível determinar o nível de prioridade que essa tarefa tem e qual o nível de urgência para que seja realizada, com objetivo que o produto seja entregue com a qualidade necessária e de acordo como solicitado na fase inicial. No entanto, o mesmo é realizado para as ressaltantes *Practice Areas*, em que o Apêndice XXII é referente à *Practice Area* “*Process Asset Development*” e o Apêndice XXIII referente à *Practice Area* “*Process Management*”.

Estas três áreas de atuação têm como objetivo, registar o desempenho das pessoas envolvidas, abordar e identificar questões de desempenho, obter dados de desempenho, analisar os dados e

armazená-los, desenvolver meios de processo para realizar o trabalho, determinar que meios utilizar, disponibilizar processos e ativos, ter uma estrutura de apoio, identificar problemas e corrigi-los para haver uma melhoria constante , avaliar a implementação dos processos atuais e identificar pontos francos e fortes dos mesmos, abordar oportunidades de melhoria ou de problemas de processos e por fim estar constantemente atualizado para implementar melhorias dos processos selecionados. Assim sendo, passamos agora para apresentação dos resultados obtidos na *Capability Area “Sustaining Habit and Persistence”*, tendo em consideração a informação recolhida.

Capability Area “Sustaining Habit and Persistence (SHP)” da Category “Improving”

Relativamente a CA “*Sustaining Habit and Persistence (SHP)*” esta dividida em duas áreas de atuação, a “*Governance (GOV)*” e “*Implementation Infrastructure (II)*”. Assim sendo, é realizado um mapeamento tendo em consideração toda a informação relativa ao *OpenUp*, *CMMI* e à ITEC, como podemos observar na Figura 47, a verde claro o nodo referente ao *OpenUp*, a castanho-escuro as áreas de atuação das *tarefas* do *OpenUp*, a azul-escuro as *tarefas* realizadas, a castanho-claro temos as *Practice Summary* referentes ao *CMMI*, a cor de rosa temos os nomes das *Practice Areas (GOV e II)* , a roxo temos a *Capability Area* em cima mencionada, amarelo temos a categoria “*Improving*”, a laranja temos os níveis de maturidade referentes ao *CMMI* e a verde-escuro temos o nodo referente ao *CMMI*.

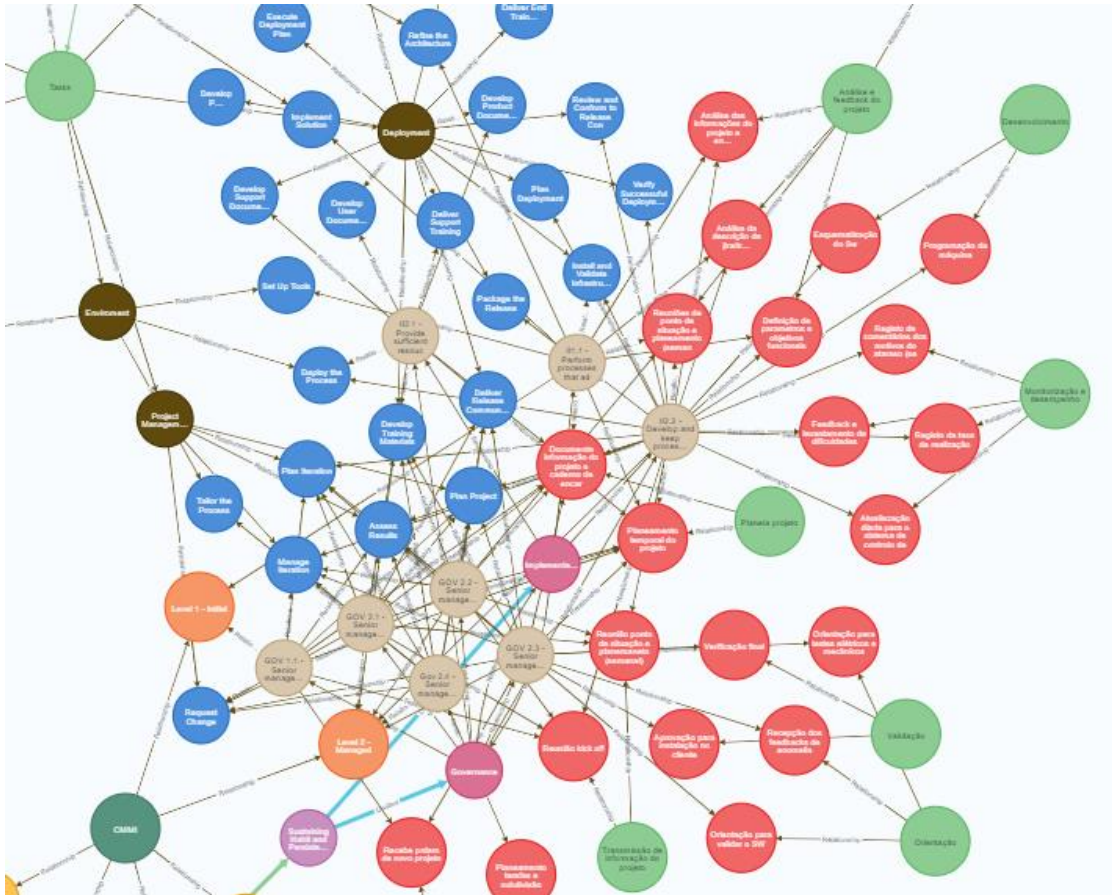


Figura 48 – Mapeamento das Capability Area SHP, Tarefas OpenUp e Tarefas da ITEC

Contudo, podemos observar no Apêndice XXIV, o cruzamento realizado tendo em consideração a *Practice Summary "Governance"*, em que são apresentados quais os níveis de prioridade, que cada atividade deve ser realizada tendo em consideração as *tarefas do OpenUp*, o mesmo acontece relativamente as relações entre *Practice Summary (CMMI)* e as *tarefas da ITEC*. Contudo, também é possível determinar qual o nível de cobertura da ITEC face às *Practice Summary* apresentadas, para além disso, é possível determinar o nível de prioridade que essa tarefa tem e qual o nível de urgência para que seja realizada, com objetivo que o produto seja entregue com a qualidade necessária e de acordo como solicitado na fase inicial.

No entanto, o mesmo é realizado para a outra *Practice Area* presente nesta *Capability Area*, em que o Apêndice XXV é referente à *Practice Area "Implementation Infrastructure"*. Estas duas áreas de atuação têm como objetivo fornecer recursos, financiamento e formação suficiente para o desenvolvimento e execução do processo, haver uma constante atualização dos processos e verificar se os mesmos estão a ser seguidos e implementar uma melhoria do desempenho dos processos.

4.5 Conclusão

O objetivo final deste mapeamento e do nível de cobertura das diversas tarefas desenvolvidas, é no final desenvolver um documento que servirá como guia de boas práticas.

Para a construção deste guia de boas práticas, iremos também utilizar os *templates* que estão disponíveis no *OpenUp*, em que o objetivo é criar um documento único que servirá como referência diária para os procedimentos da empresa, com o objetivo de melhorar a qualidade do desenvolvimento e da qualidade dos produtos/serviços fornecidos pela empresa em estudo.

5. CONCLUSÕES E TRABALHOS FUTUROS

Após várias reuniões e entrevistas estruturadas, conseguimos determinar quais as áreas presentes no departamento de desenvolvimento de software, onde são detalhadas todas as atividades e intervenientes, tendo bom base a informação recolhida. Depois de uma análise exaustiva, conseguimos apresentar algumas sugestões de melhoria que são apresentadas nos casos de uso refinados.

Sendo que uma das mais importantes é a criação da área de *Workshops*, em que achamos que é uma mais-valia para o contexto da ITEC, em que a ideia é a criação de uma “Escola/Universidade”, em que seja possível a troca de conhecimentos e aprendizagem contínua entre todos os colaboradores, onde no final de cada curso tem de realizar um exame e no final obteriam um certificado do mesmo.

Depois de analisados os casos de uso refinados, obtemos uma lista de quinze requisitos de processos, onde achamos que são fundamentais para a execução dos procedimentos diários da empresa. Para tal, em cada requisito de processos levantados elaboramos uma breve descrição, das pessoas envolvidas e da prioridade de cada um dos requisitos levantados, pois chegou-se a conclusão que seria importante que a empresa tivesse estes requisitos contidos nos seus procedimentos.

Posto isto, são estudados dois *frameworks* cujo objetivo é retirar o máximo de informação possível, para que seja possível mapeá-los e relacioná-los com o contexto da empresa, e para isso foram elaboradas tabelas e criada uma base de dados por grafos, em que foram introduzidas as informações dos *frameworks* e informação da empresa.

No entanto, conseguimos obter níveis de prioridade que cada *task* deverá ser realizada, conseguimos também obter qual o nível de cobertura de cada tarefa desempenhada pela ITEC e qual o nível de prioridade que esta tarefa tem para que o produto fique pronto e com a máxima qualidade possível, por fim, relacionando o nível de cobertura e prioridade da tarefa conseguimos obter qual o nível de urgência que cada tarefa tem que ser desempenhada. Em forma de conclusão, toda esta informação recolhida é importante para que seja possível construir um guia de boas práticas que vá de acordo com o pretendido pela empresa e que resolva o problema encontrado.

Por fim, sugere-se como trabalho futuro elaborar um estudo dos diversos *frameworks* existentes, de modo a serem utilizados como referência em empresas da área da robótica, com o objetivo de aplicar as melhores práticas nos procedimentos diários das mesmas.

BIBLIOGRAFIA

- [1] M. Chinosi and A. Trombetta, "BPMN: An introduction to the standard," *Comput. Stand. Interfaces*, vol. 34, no. 1, pp. 124–134, 2012, doi: 10.1016/j.csi.2011.06.002.
- [2] T. Koc and E. Bozdog, "Measuring the degree of novelty of innovation based on Porter's value chain approach," *Eur. J. Oper. Res.*, vol. 257, no. 2, pp. 559–567, Mar. 2017, doi: 10.1016/J.EJOR.2016.07.049.
- [3] R. Clegg, Dai; Barker, "Case Method Fast-Track: A RAD Approach No Title," 1994.
- [4] M. M. Lashin, "Different Applications of Programmable Logic Controller (PLC)," *Int. J. Comput. Sci. Eng. Inf. Technol.*, vol. 4, no. 1, pp. 27–32, 2014, doi: 10.5121/ijcseit.2014.4103.
- [5] G. C. Burdea and S. Member, "Invited Review: The Synergy Between Virtual Reality and Robotics," vol. 15, no. 3, pp. 400–410, 1999.
- [6] E. R. Alphonsus and M. O. Abdullah, "A review on the applications of programmable logic controllers (PLCs)," *Renew. Sustain. Energy Rev.*, vol. 60, pp. 1185–1205, 2016, doi: 10.1016/j.rser.2016.01.025.
- [7] R. Galin and R. Meshcheryakov, "Automation and robotics in the context of Industry 4.0: The shift to collaborative robots," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 537, no. 3, pp. 2–7, 2019, doi: 10.1088/1757-899X/537/3/032073.
- [8] A. Dresch, A. Proença, J. Antonio, and V. Antunes, "Design Science Research : método de pesquisa para a engenharia de produção," pp. 741–761, 2013, [Online]. Available: <https://www.scielo.br/j/gp/a/3CZmL4JJxLmxCv6b3pnQ8pq/abstract/?lang=en>.
- [9] J. P. and S. R. Alan R. Hevner, Salvatore T. March, "Design Science in Information Systems Research," *MIS Q.*, vol. Vol. 28, N, pp. 75-105 (31 pages), 2004, doi: <https://doi.org/10.2307/25148625>.
- [10] B. Nuseibeh and S. Easterbrook, "Requirements engineering: A roadmap," *Proc. Conf. Futur. Softw. Eng. ICSE 2000*, vol. 1, pp. 35–46, 2000, doi: 10.1145/336512.336523.
- [11] C. Orozco, C. Pardo, S. Vásquez, H. Ordoñez, and E. Suescún, "An agile process to support software configuration management | Proceso ágil para soportar la gestión de la configuración de software," *RISTI - Rev. Iber. Sist. e Technol. Inf.*, vol. 2020, no. E32, pp. 345–358, 2020.
- [12] M. STOICA, M. MIRCEA, and B. GHILIC-MICU, "Software Development: Agile vs. Traditional," *Inform. Econ.*, vol. 17, no. 4/2013, pp. 64–76, 2013, doi:

- 10.12948/issn14531305/17.4.2013.06.
- [13] N. B. Ruparelia, "Software development lifecycle models," *ACM SIGSOFT Softw. Eng. Notes*, vol. 35, no. 3, p. 8, 2010, doi: 10.1145/1764810.1764814.
- [14] S. Balaji, "Waterfall vs v-model vs agile : A comparative study on SDLC," *WATEERFALL Vs V-MODEL Vs Agil. A Comp. STUDY SDLC*, vol. 2, no. 1, pp. 26–30, 2012, [Online]. Available: <chrome-extension://efaidnbmninnibpcajpcglclefindmkaj/https://mediaweb.saintleo.edu/Courses/COM430/M2Readings/WATEERFALLVs V-MODEL Vs AGILE A COMPARATIVE STUDY ON SDLC.pdf>.
- [15] I. (n. d. . Architctcs, "The Seven Phases of the System-Development Life Cycle. Retrieved August 9, 2018," 2018. .
- [16] D. Cohen, M. Lindvall, and P. Costa, "A State of the Art Report: Agile Software Development," *DACS SOAR Rep.*, 2003.
- [17] C. Harper and G. Virk, "Towards the Development of International Safety Standards for Human Robot Interaction," *Int. J. Soc. Robot.*, vol. 2, no. 3, pp. 229–234, 2010, doi: 10.1007/s12369-010-0051-1.
- [18] S. L. Pfleeger, N. Fenton, and S. Page, "Evaluating Software Engineering Standards," *Computer (Long. Beach. Calif.)*, vol. 27, no. 9, pp. 71–79, 1994, doi: 10.1109/2.312041.
- [19] Ahern Dennis, C. Aaron, and T. Richard, *CMMI@ Distilled: A Practical Introduction to Integrated Process Improvement, Third Edition*, vol. 39. 2008.
- [20] M. B. Chrissis, M. Konrad, and S. Shrum, *CMMI for Development: Guidelines for Process Integration and Product Improvement (SEI Series in Software Engineering) 3rd Edition*. 2011.
- [21] ISACA, *CMMI Model V2.0 - Overview*. 2022.
- [22] M. Ybanez, "Estendendo o OpenUP para Atender as Areas de Processo Relacionadas a Garantia da Qualidade e ~ o do CMMI-DEV N ´ Medic , a," 2017, [Online]. Available: <https://silo.tips/download/estendendo-o-openup-para-atender-as-areas-de-processo-relacionadas-a-garantia-da>.
- [23] M. Ybanez, "Estendendo o OpenUP para Atender as Areas de Processo Relacionadas a Garantia da Qualidade e ~ o do CMMI-DEV N ´ Medic , a," 2017.
- [24] I. Sommerville, "Engenharia de Software," *Pearson Bras.*, p. 544, 2011, [Online]. Available: http://tmv.edu.in/pdf/Diploma Syllabus/Computer/TY_fifth_sem/Fifth Semester Curriculum.pdf.
- [25] L. Diedrich, "Integração da metodologia ágil OpenUp nos processos de engenharia de software,"

- 2011.
- [26] S. Roongsangjan, "Using FCA Implication to Determine the Compliance of Model Practice Implementation for Software Process," 2017, [Online]. Available: <https://dl.acm.org/doi/abs/10.1145/3034950.3034998>.
 - [27] C. Vicknair, X. Nan, M. Macias, Y. Chen, Z. Zhao, and D. Wilkins, "A comparison of a graph database and a relational database: A data provenance perspective," *Proc. Annu. Southeast Conf.*, 2010, doi: 10.1145/1900008.1900067.
 - [28] P. T. Wood, "Query languages for graph databases," *SIGMOD Rec.*, vol. 41, no. 1, pp. 50–60, 2012, doi: 10.1145/2206869.2206879.
 - [29] P. Rodriguez, M. A., & Neubauer, *The Graph Traversal Pattern*. 2010.
 - [30] M. A. Rodriguez and P. Neubauer, "Constructions from dots and lines," *Bull. Am. Soc. Inf. Sci. Technol.*, vol. 36, no. 6, pp. 35–41, 2010, doi: 10.1002/bult.2010.1720360610.
 - [31] J. Miller, "Graph Database Applications and Concepts with Neo4," *Proc. 2013 South. Assoc.*, pp. 141–147, 2013.
 - [32] R. Angles, "A comparison of current graph database models," *Proc. - 2012 IEEE 28th Int. Conf. Data Eng. Work. ICDEW 2012*, pp. 171–177, 2012, doi: 10.1109/ICDEW.2012.31.
 - [33] D. Engines, "No TitleDB-Engines ranking of graph DBMS.," 2018.
 - [34] Neo4J, "What is a Graph Database?" .
 - [35] I. Bittner, K., & Spence, *Use Case Modeling*. 2002.
 - [36] Lukuisia kirjoittajia, *Software Business Community*. 2014.

APÊNDICE I

ID: RP2 - Definir objetivos atribuindo peso ao tempo
Descrição: A reunião de <i>Kickoff</i> , tem como propósito dar início ao projeto. Nesta reunião participam os chefes de equipa e gestores de projetos, que, em conjunto, definem os parâmetros e objetivos funcionais a obter ao longo do projeto. Estas decisões são tomadas com base no planeamento temporal, previamente definido pelo gestor de projetos e no caderno de encargos. Para a tomada de decisão, e ao longo da discussão, deveriam ser considerados e atribuídos pesos e tempos realistas nos vários parâmetros e objetivos.
Fonte do requisito: Chefe de Equipa
Interfaces: N/A
Dependência: <ul style="list-style-type: none">● Stakeholders<ul style="list-style-type: none">○ Gestor de Projetos○ Chefe de Equipa○ Programador
Problemas identificados: N/A
Prioridades: Must

ID: RP3 - Realizar reunião de planeamento por projeto
Descrição: Em alternativa à reunião global, já realizada pela empresa onde se juntam todos os projetos, deveria ser implementada a reunião de planeamento projeto a projeto, para se fazer um ponto de situação, adaptar prazos e objetivos de cada projeto individualmente.
Fonte de requisitos: Gestor de Projetos e Chefe de Equipa
Interfaces: N/A
Dependência: <ul style="list-style-type: none">● Stakeholders<ul style="list-style-type: none">○ Gestor de Projetos○ Chefe de Equipa
Problemas identificados: N/A
Prioridade: Must

ID: RP4 - Planear e alocar tarefas atribuindo peso à dificuldade
Descrição: Após a reunião de <i>Kickoff</i> o chefe de Equipa tem como principal responsabilidade planear as tarefas e escolha de programadores, assim como a escolha dos elementos da sua equipa de trabalho, considerando a dificuldades das várias tarefas.
Fonte de requisitos: Chefe de Equipa
Interfaces: N/A
Dependências: <ul style="list-style-type: none"> ● Stakeholders <ul style="list-style-type: none"> ○ Chefe de Equipa; ○ Programadores;
Problemas identificados: N/A
Prioridade: Must

ID: RP5 - Analisar o caderno de encargos de forma normalizada
Descrição: O programador deverá iniciar a análise do caderno de encargos de forma normalizada (em tempo, ou seja, impondo um período mínimo de análise).
Fonte de requisitos: Programador
Interfaces: N/A
Dependências: <ul style="list-style-type: none"> ● Stakeholders <ul style="list-style-type: none"> ○ Chefe de Equipa ○ Programador
Problemas identificados: N/A
Prioridades: Must

ID: RP6 - Esquematizar e planear o software de forma normalizada
Descrição: O programador deverá realizar um planeamento e esquematização do <i>software</i> de forma normalizada (em método e em tempo), ou seja, este só deveria passar para a fase de desenvolvimento (programação) depois de ter definido todos os processos e necessidades que irá ter ao longo do projeto, para assim reduzir tempo na realização do mesmo.
Fontes de requisitos: Programador
Interfaces: N/A
Dependências: <ul style="list-style-type: none"> • Stakeholders <ul style="list-style-type: none"> ○ Programador;
Problemas Identificados: N/A
Prioridade: Must

ID: RP8 - Planear e preparar workshops
Descrição: Com vista a promover o incentivo à aprendizagem contínua sob a forma de <i>workshops</i> , estes deveriam abordar tópicos técnicos ao nível de linguagens de programação, processos de gestão e metodologias Agile (<i>Lean, Scrum, Kaban</i>). O chefe de Equipa ficará responsável pela preparação e planeamento destes workshops sendo que os colaboradores que compõem a sua equipa irão participar nos <i>workshops</i> e obter eventuais certificados.
Fontes de Requisitos: Chefe de Equipa
Interfaces: N/A
Dependências: <ul style="list-style-type: none"> • Stakeholders <ul style="list-style-type: none"> ○ Chefe de Equipa ○ Programadores;
Problemas identificados: N/A
Prioridade: Must

ID: RP9 - Participar em <i>Workshops</i>
Descrição: O objetivo da participação nestes workshops, é que no futuro os participantes utilizem estes conhecimentos regularmente ao longo dos projetos em que estejam a trabalhar ou que sejam eles a dar os <i>workshops</i> a eventuais novos colaboradores ou mesmo a colegas que ainda não tenham tido oportunidade de os frequentar. Desta forma os seus conhecimentos serão aplicados nos projetos futuros promovendo assim melhorias significativas na qualidade do produto final.
Fontes de Requisitos: Chefe de Equipa
Interfaces: N/A
Dependência: <ul style="list-style-type: none"> ● Stakeholders <ul style="list-style-type: none"> ○ Gestor de projetos ○ Chefe de Equipa e Programadores;
Problemas identificados: N/A
Prioridade: Must

ID: RP10 - Obter certificado de frequência nos workshops
Descrição: Os colaboradores que já tenham frequentado workshops anteriores e que tenham recebido o certificado da conclusão do mesmo, poderão no futuro ser eles os responsáveis por dar os próximos <i>workshops</i> a colaboradores novos da empresa ou até mesmo a quem não teve oportunidade de participar nos anteriores. Com isto, consegue-se criar uma consolidação de toda a informação teórico-prática na empresa, garantir a passagem de conhecimento a novos participantes e testa-se assim as capacidades de ensino e aplicabilidade de quem já frequentou anteriormente.
Fontes de Requisitos: Chefe de Equipa e Programadores
Interfaces: N/A
Dependência: <ul style="list-style-type: none"> ● Stakeholders <ul style="list-style-type: none"> ○ Gestor de Projetos; ○ Chefe de Equipa; ○ Programador;
Problemas identificados: N/A
Prioridade: Should

ID: RP11 - Realizar reunião breve de <i>feedback</i> e dificuldades
Descrição: Foi estabelecida a mudança da reunião semanal de feedback e discussão de dificuldades para uma reunião breve diária, ou cíclica com maior frequência. Este cenário apareceu após a comprovação, via questionários e análise dos processos, onde constatamos que uma reunião semanal para este tipo de questões acaba por tornar o seu propósito obsoleto ou impraticável, visto que ninguém espera e guarda dúvidas especificamente para serem discutidas uma vez por semana num dia definido.
Fontes de Requisitos: Programadores
Dependência: <ul style="list-style-type: none"> ● Stakeholders <ul style="list-style-type: none"> ○ Gestor de Equipas; ○ Chefe de Equipa; ○ Programador;
Problemas identificados: N/A
Prioridade: Must!

ID: RP12 - Escrever de forma regular comentários no código
Descrição: Registo regular e contínuo de comentários perceptíveis e explicados à medida que se escreve o código. Esta tarefa tem como finalidade atingir o rigor e standard ao nível de boas práticas de programação, permitindo assim um fácil retorno aquando da leitura do código no futuro e também garantindo que outros programadores, que possam vir a trabalhar nesse mesmo código, possam facilmente entender as suas funcionalidades.
Fonte de Requisitos: Programador
Interfaces: N/A
Dependência: <ul style="list-style-type: none"> ● Stakeholders <ul style="list-style-type: none"> ○ Programador;
Problemas identificados: N/A
Prioridade: Must!

ID: RP13 - Definir e escrever o guia de boas práticas
Descrição: Haverá um documento oficial e padrão que explica as práticas normalizadas na programação, que tem como garantia que todas as equipas de desenvolvimento trabalhem da mesma forma. Esta tarefa em conjunto com a validação do código, que vai ser apresentado de seguida, tem como finalidade garantir um nível de <i>standard</i> relativamente ao controlo de qualidade. Esta tarefa deverá ser desempenhada por equipas externas aos próprios programadores, como por exemplo o Técnico de Qualidade.
Fonte de Requisitos: Programador e Técnico de Qualidade
Interfaces: N/A
Dependência: <ul style="list-style-type: none"> ● Stakeholders <ul style="list-style-type: none"> ○ Programador ○ Técnico de Qualidade ○ Chefe de Equipa
Problemas identificados: N/A
Prioridade: Must!

ID: RP14 - Validar código de forma contínua
Descrição: Haverá também controlo de qualidade regulares sobre o trabalho desenvolvido pelos programadores, com o propósito de garantir que as suas tarefas estão a ser respeitadas, como por exemplo a escrita regular de comentários na escrita de código. Irá ser igualmente controlada utilização efetiva do guia de boas práticas referido em F13.
Fonte de Requisitos: Técnico de Qualidade
Interfaces: N/A
Dependência: <ul style="list-style-type: none"> ● Stakeholders <ul style="list-style-type: none"> ○ Técnico de Qualidade ○ Programador e Chefe de Equipa
Problemas identificados: N/A
Prioridade: Must!

ID: RP15 - Obter feedback dos clientes

Descrição: Esta tarefa tem como propósito criar-se uma relação mais próxima, saudável e normalizada após a instalação dos produtos finais no cliente, com o objetivo de procurar a opinião do cliente, seja por questionários ou por contacto direto, com isto estamos a dar oportunidade ao cliente de dar a sua opinião acerca do produto em causa.

Fontes de Requisitos: Cliente

Interfaces: N/A

Dependências:

- **Stakeholders**
 - Gestor de Projetos
 - Chefe de Equipa
 - Programador
 - Cliente

Problemas identificados: N/A

Priority: Must!

APÊNDICE II

Doing	Engineering & developing products (EDP)	Product Integration (PI)	CMMI							
			ML1			ML2				
			PI 1.1 Assemble solutions and deliver to the customer.	PI 2.1 Develop, keep updated, and follow an integration strategy.	PI 2.2 Develop, keep updated, and use the integration environment.	PI 2.3 Develop, keep updated, and follow procedures and criteria for integrating solutions and components.	PI 2.4 Confirm, prior to integration, that each component has been properly identified and operates according to its requirements and design.	PI 2.5 Evaluate integrated components to ensure conformance to the solution's requirements and design.	PI 2.6 Integrate solutions and components according to the integration strategy.	
		Tasks OpenUp								
		Refine the Architecture		H	H					
		Assess Results						H		
		Plan Project		H						
		Develop Technical Vision		H	H					
		Integrate and Create Build			M	H				
		Design the Solution		H	M	H				
		Deploy the Process			M				H	
		Tailor the Process			M				H	
		Set Up Tools		M	H				H	
		Verify Tool Configuration and Installation		M	M			H	H	
		Create Test Cases					M			
		Implement Tests					M			
		Run Tests					H			
		Develop User Documentation			H	H				
		Develop Support Documentation		H	H					
		Deliver end user Training			H					
		Deliver Support Training			H					
		Develop Training Materials			H					
		Deliver Release Communications		M						
		Install and Validate Infrastructure	H					H		
		Plan Deployment		H				H		
		Review and Conform to Release Controls		H				H		
Doing	Engineering & developing products (EDP)	Product Integration (PI)	CMMI							
			ML1			ML2				
			PI 1.1 Assemble solutions and deliver to the customer.	PI 2.1 Develop, keep updated, and follow an integration strategy.	PI 2.2 Develop, keep updated, and use the integration environment.	PI 2.3 Develop, keep updated, and follow procedures and criteria for integrating solutions and components.	PI 2.4 Confirm, prior to integration, that each component has been properly identified and operates according to its requirements and design.	PI 2.5 Evaluate integrated components to ensure conformance to the solution's requirements and design.	PI 2.6 Integrate solutions and components according to the integration strategy.	
		Tasks ITEC								
		121 Documenta informação do projeto e caderno de encargos		x		x				
		122 Planeamento temporal do projeto		x		x				
		131 Reunião kick off		x						
		132 Reunião ponto de situação e planeamento (semanal)		x						
		141 Planeamento tarefas e subdivisão		x						
		151 Receção dos feedbacks de anomalia		x	x		x	x		
		152 Orientação para validar o SW		x	x	x	x			
		153 Orientação para testes elétricos e mecânicos		x	x	x	x	x		
		161 Verificação final		x	x	x		x	x	
		162 Aprovação para instalação no cliente		x				x	x	
		2201 Reunião Kick-off		x						
		221 Análise das informações do projeto e caderno de encargos		x						
		222 Definição de parâmetros e objetivos funcionais		x						
		223 Análise da descrição de tarefas jira/trello		x						
		224 Reuniões de ponto de situação e planeamento (semanal)		x						
		232 Programação da máquina	x							
		241 Feedback e levantamento de dificuldades		x	x	x	x			
		242 Atualização diária para o sistema de controlo de versões		x	x	x	x			
		243 Registo da taxa de realização		x	x	x	x			
		244 Registo de comentários dos motivos do ataraso (se houver)		x	x	x	x			
		251 Validação do SW		x	x	x	x			
		252 Realização de teste elétricos e mecânicos		x	x	x	x	x		
		253 Retrabalho e revisão		x	x	x	x	x		
		261 Instalação do equipamento no cliente final	x							
		271 Assistência técnica e suporte						x	x	
		272 Registo de reclamações					x	x	x	
		273 Solução de reclamações						x		
		Coverage	H	I	S	S	S	S	I	
		Priority	HP	MP	MP	MP	HP	MP	HP	
		Action is urgent?	HHP	IMP	SMP	SMP	SHP	SMP	IHP	

APÊNDICE III

Doing	Engineering & developing products (EDP)	Technical Solution (TS)	CMMI			
			ML1	ML2		
			TS 1.1 Build a solution to meet requirements.	TS 2.1 Design and build a solution to meet requirements.	TS 2.2 Evaluate the design and address identified issues.	TS 2.3 Provide guidance on use of the solution.
		Tasks OpenUp				
		Refine the Architecture	H	H		
		Envision the Architecture		H		
		Assess Results			M	H
		Manage Iteration				H
		Develop Technical Vision	H	H+		
		Implement Developer Tests				
		Implement Solution	H		M	H
		Design the Solution		H		
		Tailor the Process		M		
		Set Up Tools		M		
		Implement Tests			M	
		Run Tests			M	
		Develop Product Documentation				H
		Develop User Documentation				H
		Develop Support Documentation				H
		Deliver end user Training				H
		Deliver Support Training				H
		Develop Training Materials				H
		Deliver Release Communications				M
		Execute Backout Plan (if necessary)		L	L	
		Verify Successful Deployment			M	
		Develop Backout Plan		M		L
		Develop Release Communications				M
		Install and Validate Infrastructure	H			
		Plan Deployment		M		
Doing	Engineering & developing products (EDP)	Technical Solution (TS)	CMMI			
			ML1	ML2		
			TS 1.1 Build a solution to meet requirements.	TS 2.1 Design and build a solution to meet requirements.	TS 2.2 Evaluate the design and address identified issues.	TS 2.3 Provide guidance on use of the solution.
		Tasks ITEC				
		121 Documenta informação do projeto e caderno de encargos	x	x		
		122 Planeamento temporal do projeto	x	x		
		151 Recepção dos feedbacks de anomalia				x
		152 Orientação para validar o SW				x
		153 Orientação para testes elétricos e mecânicos				x
		221 Análise das informações do projeto e caderno de encargos	x	x	x	x
		222 Definição de parametros e objetivos funcionais	x	x	x	x
		223 Análise da descrição de tarefas jira/trello	x	x	x	x
		224 Reuniões de ponto de situação e planeamento (semanal)			x	x
		231 Esquematisação do Sw	x	x		
		241 Feedback e levantamento de dificuldades	x	x		
		242 Atualização diária para o sistema de controlo de versões	x	x		x
		243 Registo da taxa de realização	x	x		x
		244 Registo de comentários dos motivos do ataraso (se houver)				x
		Coverage	S	I	I	I
		Priority	HP	HP	MP	MP
		Action is urgent?	SHP	IHP	IMP	IMP

APÊNDICE IV

Doing	Ensuring Quality (ENQ)	Peer Reviews (PR)	CMMI				
			ML1	ML2			
		Tasks OpenUp	PR 1.1 Perform reviews of work products and record issues.	PR 2.1 Develop and keep updated procedures and supporting materials used to prepare for and perform peer reviews.	PR 2.2 Select work products to be peer reviewed.	PR 2.3 Prepare and perform peer reviews on selected work products using established procedures.	PR 2.4 Resolve issues identified in peer reviews.
		Assess Results					M
		Manage Iteration					M
		Request Change					H
		Run Developer Tests				M	
		Create Test Cases		H			
		Implement Tests			H		
		Run Tests				H	
		Develop Product Documentation		M			
		Develop User Documentation		M			
		Develop Support Documentation		M			
		Review and Conform to Release Controls	H			H	
Doing	Ensuring Quality (ENQ)	Peer Reviews (PR)	CMMI				
			ML1	ML2			
		Tasks ITEC	PR 1.1 Perform reviews of work products and record issues.	PR 2.1 Develop and keep updated procedures and supporting materials used to prepare for and perform peer reviews.	PR 2.2 Select work products to be peer reviewed.	PR 2.3 Prepare and perform peer reviews on selected work products using established procedures.	PR 2.4 Resolve issues identified in peer reviews.
		151 Recepção dos feedbacks de anomalia	x	x			x
		152 Orientação para validar o SW	x	x			x
		153 Orientação para testes elétricos e mecânicos	x	x			x
		161 Verificação final	x	x			x
		162 Aprovação para instalação no cliente	x	x			x
		221 Análise das informações do projeto e caderno de encargos			x	x	
		222 Definição de parametros e objetivos funcionais			x	x	
		223 Análise da descrição de tarefas jira/trello			x	x	
		224 Reuniões de ponto de situação e planeamento (semanal)			x	x	
		241 Feedback e levantamento de dificuldades		x			
		242 Atualização diária para o sistema de controlo de versões		x			
		243 Registo da taxa de realização		x			
		244 Registo de comentários dos motivos do ataraso (se houver)		x			
		251 Validação do SW	x	x			x
		252 Realização de teste elétricos e mecânicos	x	x			x
		253 Retrabalho e revisão	x	x			x
		271 Assistência técnica e suporte					x
		272 Registo de reclamações					x
		273 Solução de reclamações					x
		Coverage	S	S	S	J	J
		Priority	HP	MP	MP	MP	HP
		Action is urgent?	SHP	SMP	SMP	IMP	IHP

APÊNDICE V

Doing	Ensuring Quality (ENQ)	Process Quality Assurance (PQA)	CMMI							
			ML1	ML2						
			PQA 1.1 Identify and address process and work product issues.	PQA 2.1 Develop, keep updated, and follow a quality assurance approach and plan based on historical quality data.	PQA 2.2 Objectively evaluate selected performed processes and work products against the recorded process and applicable standards.	PQA 2.3 Communicate quality and non-compliance issues and ensure their resolution.	PQA 2.4 Record and use results of quality assurance activities.			
		Tasks OpenUp								
		Envision the Architecture				M				
		Assess Results		H	H	H	H			
		Plan Project		H						
		Request Change					H			
		Implement Developer Tests		M						
		Run Developer Tests			M		H			
		Design the Solution					H			
		Deploy the Process	H							
		Tailor the Process	H							
		Set Up Tools		H						
		Verify Tool Configuration and Installation								
		Create Test Cases		H						
		Implement Tests			H					
		Run Tests			H					
		Develop Product Documentation							H	
		Execute Backout Plan (if necessary)			H					
		Develop Backout Plan			H					
		Review and Conform to Release Controls			H					
Doing	Ensuring Quality (ENQ)	Process Quality Assurance (PQA)	CMMI							
			ML1	ML2						
			PQA 1.1 Identify and address process and work product issues.	PQA 2.1 Develop, keep updated, and follow a quality assurance approach and plan based on historical quality data.	PQA 2.2 Objectively evaluate selected performed processes and work products against the recorded process and applicable standards.	PQA 2.3 Communicate quality and non-compliance issues and ensure their resolution.	PQA 2.4 Record and use results of quality assurance activities.			
		Tasks ITEC								
		111 Recebe prdem de novo projeto								
		121 Documenta informação do projeto e caderno de encargos		x	x					
		122 Planeamento temporal do projeto		x	x					
		131 Reunião kick off		x	x					
		132 Reunião ponto de situação e planemaneto (semanal)		x	x					
		141 Planeamento tarefas e subdivisão		x	x					
		151 Recepção dos feedbacks de anomalia		x	x					
		152 Orientação para validar o SW		x	x					
		153 Orientação para testes elétricos e mecânicos		x	x					
		221 Análise das informações do projeto e caderno de encargos	x	x	x					
		222 Definição de parametros e objetivos funcionais	x	x	x					
		223 Análise da descrição de tarefas jira/trello	x	x	x					
		224 Reuniões de ponto de situação e planeamento (semanal)	x	x	x					
		231 Esquematização do Sw		x						
		232 Programação da máquina		x						
		241 Feedback e levantamento de dificuldades	x							
		242 Atualização diária para o sistema de controlo de versões	x							
		243 Registo da taxa de realização	x					x		x
		244 Registo de comentários dos motivos do ataraso (se houver)	x					x		x
		251 Validação do SW	x		x			x		x
		252 Realização de teste elétricos e mecânicos	x		x			x		x
		253 Retrabalho e revisão	x		x			x		x
		Coverage	S	I	I			S		I
		Priority	MP	MP	MP			HP		MP
		Action is urgent?	SMP	IMP	IMP			SHP		IMP

APÊNDICE VII

Doing	Ensuring Quality (ENQ)	Verification and Validation (VV)	CMMI				
			ML1		ML2		
			PI 1.1 Assemble solutions and deliver to the customer.	VV 1.2 Perform validation to ensure the solution will function as intended in its target environment and record and communicate results.	VV 2.1 Select components and methods for verification and validation.	VV 2.2 Develop, keep updated, and use the environment needed to support verification and validation.	VV 2.3 Develop, keep updated, and follow procedures for verification and validation.
Tasks OpenUp							
		Assess Results	M		H	H	H
		Plan Project	M				
		Request Change					H
		Implement Developer Tests				M	
		Implement Solution				M	
		Run Developer Tests				H	
		Deploy the Process			M		
		Tailor the Process			M		
		Set Up Tools			H		
		Verify Tool Configuration and Installation			M		
		Create Test Cases	M	H			H
		Implement Tests	H	H+			H
		Run Tests	H	H+		H	H
		Develop Product Documentation				H	H
		Develop User Documentation				H	H
		Develop Support Documentation				H	H
		Execute Deployment Plan			H	H	H
		Verify Successful Deployment	H	H	H+	H+	H+
		Develop Backout Plan					H
		Develop Release Communications					H
		Plan Deployment			H	H	H+
		Review and Conform to Release Controls	M	M	H	H	H+
Doing	Ensuring Quality (ENQ)	Verification and Validation (VV)	CMMI				
			ML1		ML2		
			PI 1.1 Assemble solutions and deliver to the customer.	VV 1.2 Perform validation to ensure the solution will function as intended in its target environment and record and communicate results.	VV 2.1 Select components and methods for verification and validation.	VV 2.2 Develop, keep updated, and use the environment needed to support verification and validation.	VV 2.3 Develop, keep updated, and follow procedures for verification and validation.
Tasks ITEC							
		152 Orientação para validar o SW			x		x
		153 Orientação para testes elétricos e mecânicos			x		x
		161 Verificação final				x	x
		162 Aprovação para instalação no cliente				x	x
		2201 Reunião Kick-off				x	x
		221 Análise das informações do projeto e caderno de encargos				x	x
		222 Definição de parâmetros e objetivos funcionais				x	x
		223 Análise da descrição de tarefas jira/trello			x	x	x
		224 Reuniões de ponto de situação e planeamento (semanal)				x	x
		231 Esquemática do Sw				x	x
		232 Programação da máquina				x	x
		241 Feedback e levantamento de dificuldades			x		
		242 Atualização diária para o sistema de controlo de versões			x	x	x
		243 Registo da taxa de realização				x	x
		244 Registo de comentários dos motivos do atraso (se houver)			x	x	x
		251 Validação do SW		x		x	x
		252 Realização de teste elétricos e mecânicos		x		x	x
		253 Retrabalho e revisão		x		x	x
		261 Instalação do equipamento no cliente final	x				x
		Coverage	H	S	S	I	S
		Priority	HP	HP	MP	MP	MP
		Action is urgent?	HHP	SHP	SMP	IMP	SMP

APÊNDICE VIII

Managing Business Resilience (MBR)	Continguity (CONT)	Tasks OpenUp	CMMI			
			ML1	ML2		
			CONT 1.1 Develop contingency approaches for managing significant disruptions to	CONT 2.1 Identify and prioritize functions essential for continuity.	CONT 2.2 Identify and prioritize resources essential for continuity.	CONT 2.3 Develop, keep updated, and follow continuity plans to resume performing essential
		Assess Results		H		
		Plan Project		H	H	H
		Identify and Outline Requirements		M	M	H
		Develop Technical Vision		M	M	H
		Tailor the Process		H	H	
		Verify Tool Configuration and Installation	H	H		H
		Develop Product Documentation		M	M	M
		Execute Backout Plan (if necessary)				H
		Execute Deployment Plan				H
		Verify Successful Deployment		H	H	H
		Develop Backout Plan				H
		Install and Validate Infrastructure	H			
		Plan Deployment		H		
		Review and Conform to Release Controls		H		
Managing Business Resilience (MBR)	Continguity (CONT)	Tasks ITEC	CMMI			
			ML1	ML2		
			CONT 1.1 Develop contingency approaches for managing significant disruptions to operations.	CONT 2.1 Identify and prioritize functions essential for continuity.	CONT 2.2 Identify and prioritize resources essential for continuity.	CONT 2.3 Develop, keep updated, and follow continuity plans to resume performing essential functions.
		121 Documenta informação do projeto e caderno de encargos		x	x	x
		122 Planeamento temporal do projeto		x	x	x
		131 Reunião kick off		x		
		132 Reunião ponto de situação e planeamento (semanal)		x		
		141 Planeamento tarefas e subdivisão		x		
		151 Recepção dos feedbacks de anomalia	x	x	x	x
		152 Orientação para validar o SW	x	x	x	x
		153 Orientação para testes elétricos e mecânicos	x	x	x	x
		210 Pré ou pós Kick-off		x		
		211 Recebe novo projeto		x		
		2201 Reunião Kick-off		x		
		221 Análise das informações do projeto e caderno de encargos		x	x	x
		222 Definição de parâmetros e objetivos funcionais		x	x	x
		223 Análise da descrição de tarefas jira/trello		x	x	x
		224 Reuniões de ponto de situação e planeamento (semanal)		x	x	x
		231 Esquematização do Sw		x	x	
		232 Programação da máquina		x	x	
		241 Feedback e levantamento de dificuldades		x	x	x
		242 Atualização diária para o sistema de controlo de versões		x	x	
		243 Registo da taxa de realização		x	x	
		244 Registo de comentários dos motivos do ataraso (se houver)		x	x	x
		271 Assistência técnica e suporte	x			
		272 Registo de reclamações	x			
		273 Solução de reclamações	x			
		Coverage	I	S	S	S
		Priority	HP	MP	MP	MP
		Action is urgent?	WHP	WMP	WMP	WMP

APÊNDICE IX

Managing Business Resilience (MBR)	Managing Business Resilience (MBR)	Incident Resolution and Prevention (IRP)	CMMI			
			ML1	ML2		
			IRP 1.1 Record and resolve incidents.	IRP 2.1 Develop, keep updated, and follow an approach for incident resolution and prevention.	IRP 2.2 Monitor and resolve each incident to closure.	IRP 2.3 Communicate incident status.
		Tasks OpenUp				
		Assess Results		H	M	
		Manage Iteration			H	
		Request Change		H		
		Implement Developer Tests		H		
		Implement Solution		H		
		Run Developer Tests		H		
		Create Test Cases		H		
		Implement Tests		H		
		Run Tests	H	H		
		Develop Product Documentation		H		
		Develop User Documentation		L		
		Develop Support Documentation		L		
		Verify Successful Deployment		H		
		Develop Release Communications				H
		Review and Conform to Release Controls	H		H	
Managing Business Resilience (MBR)	Managing Business Resilience (MBR)	Incident Resolution and Prevention (IRP)	CMMI			
			ML1	ML2		
			IRP 1.1 Record and resolve incidents.	IRP 2.1 Develop, keep updated, and follow an approach for incident resolution and prevention.	IRP 2.2 Monitor and resolve each incident to closure.	IRP 2.3 Communicate incident status.
		Tasks ITEC				
		151 Recepção dos feedbacks de anomalia		x	x	x
		152 Orientação para validar o SW		x	x	x
		153 Orientação para testes elétricos e mecânicos		x	x	x
		241 Feedback e levantamento de dificuldades		x	x	
		242 Atualização diária para o sistema de controlo de versões		x	x	
		243 Registo da taxa de realização		x	x	
		244 Registo de comentários dos motivos do ataraso (se houver)		x	x	
		253 Retrabalho e revisão	x			
		271 Assistência técnica e suporte	x	x	x	
		272 Registo de reclamações	x	x	x	x
		273 Solução de reclamações	x	x	x	
		Coverage	S	I	I	I
		Priority	HP	MP	HP	LP
		Action is urgent?	SHP	IMP	IHP	ILP

APÊNDICE X

Managing	Managing Business Resilience (MBR)	Risk and Opportunity Management (RSK)	CMMI		
			ML1	ML2	
		Tasks OpenUp	RSK 1.1 Identify and record risks or opportunities and keep them updated.	RSK 2.1 Analyze identified risks or opportunities.	RSK 2.2 Monitor identified risks or opportunities and communicate status to affected stakeholders.
		Assess Results			M
		Manage Iteration			H
		Plan Project	M	H	
		Identify and Outline Requirements	M	H	
		Create Test Cases		M	
		Implement Tests		M	
		Run Tests		M	
		Deliver Release Communications			H
		Install and Validate Infrastructure			H
		Plan Deployment			H
		Review and Conform to Release Controls			M
Managing	Managing Business Resilience (MBR)	Risk and Opportunity Management (RSK)	CMMI		
			ML1	ML2	
		Tasks ITEC	RSK 1.1 Identify and record risks or opportunities and keep them updated.	RSK 2.1 Analyze identified risks or opportunities.	RSK 2.2 Monitor identified risks or opportunities and communicate status to affected stakeholders.
		121 Documenta informação do projeto e caderno de encargos	x		
		122 Planeamento temporal do projeto	x		
		161 Verificação final			x
		162 Aprovação para instalação no cliente			x
		221 Análise das informações do projeto e caderno de encargos		x	
		222 Definição de parametros e objetivos funcionais		x	
		231 Esquematisação do Sw	x		
		242 Atualização diária para o sistema de controlo de versões	x		
		243 Registo da taxa de realização	x		
		244 Registo de comentários dos motivos do ataraso (se houver)	x		
		251 Validação do SW			x
		252 Realização de teste elétricos e mecânicos			x
		253 Retrabalho e revisão			x
		Coverage	S	I	I
		Priority	HP	MP	MP
		Action is urgent?	SHP	IMP	IMP

APÊNDICE XI

		CMMI							
		ML1		ML2					
Managing Managing the Workforce (MWF)	Managing Virtual Solution Delivery (EVSD)	Tasks OpenUp	EVSD 1.1 Identify and record virtual solution delivery needs and constraints.	EVSD 1.2 Deliver virtual solutions.	EVSD 2.1 Develop, communicate, and follow an approach to address virtual solution delivery needs and constraints; and keep it updated.	EVSD 2.2 Evaluate the effectiveness and fidelity of the virtual solution delivery approach and take action to improve performance.			
			Assess Results			H			
			Plan Project	H		H			
			Develop Product Documentation	M		M			
			Develop User Documentation	M		M			
			Develop Support Documentation	M		M			
			Deliver end user Training		M	M			
			Deliver Support Training		M	M			
			Develop Training Materials	H		H			
			Deliver Release Communications	H	H	H			
			Plan Deployment	H		H			
			Review and Conform to Release Controls				H		
			Managing Managing the Workforce (MWF)	Managing Virtual Solution Delivery (EVSD)	Tasks ITEC	EVSD 1.1 Identify and record virtual solution delivery needs and constraints.	EVSD 1.2 Deliver virtual solutions.	EVSD 2.1 Develop, communicate, and follow an approach to address virtual solution delivery needs and constraints; and keep it updated.	EVSD 2.2 Evaluate the effectiveness and fidelity of the virtual solution delivery approach and take action to improve performance.
						243 Registo da taxa de realização	x	-	-
244 Registo de comentários dos motivos do ataraso (se houver)	x	-				-	-		
Coverage	S	-				-	-		
Priority	MP	-				-	-		
Action is urgent?	SMP	-				-	-		

APÊNDICE XII

Managing	Managing the Workforce (MWF)	Organizational Training (OT)	CMMI			
			Tasks OpenUp	ML1	ML2	
				OT 1.1 Train people.	OT 2.1 Identify training needs.	OT 2.2 Train personnel and keep records.
			Assess Results	M		
			Plan Project	M		
			Detail Use-Case Scenarios	M		
			Tailor the Process	M	M	
			Develop Product Documentation	H		
			Develop User Documentation	H		
			Develop Support Documentation	H		
			Deliver end user Training	H	H	
			Deliver Support Training	H	H	
			Develop Training Materials	H		
			Plan Deployment	M		
Managing	Managing the Workforce (MWF)	Organizational Training (OT)	CMMI			
			Tasks ITEC	ML1	ML2	
				OT 1.1 Train people.	OT 2.1 Identify training needs.	OT 2.2 Train personnel and keep records.
			152 Orientação para validar o SW	x		
			153 Orientação para testes elétricos e mecânicos	x		
			Coverage	I	W	W
			Priority	HP	HP	HP
			Action is urgent?	IHP	WHP	WHP
Este é um dos pontos que nós ao elaborar achamos que seria uma mais valia implementar na empresa... que seria uma partilha de conhecimento, com isto demos como sugestao a implementação de uma especie de "Workshops"						

APÊNDICE XIII

		CMMI				
		ML1	ML2	EST 2.2 Develop and keep updated estimates for the size of the solution.	EST 2.3 Based on size estimates, develop and record effort, duration, and cost estimates and their rationale for the solution.	
Managing Planning and Managing Work (PMW)	Estimating (EST)	Tasks OpenUp	EST 1.1 Develop high-level estimates to perform the work.	EST 2.1 Develop, keep updated, and use the scope of what is being estimated.	EST 2.2 Develop and keep updated estimates for the size of the solution.	EST 2.3 Based on size estimates, develop and record effort, duration, and cost estimates and their rationale for the solution.
		Refine the Architecture			M	
		Envision the Architecture			M	
		Assess Results				H
		Manage Iteration				H
		Plan Iteration		H		
		Plan Project	H	H	H	
		Identify and Outline Requirements	H	H		
		Detail Use-Case Scenarios		H		
		Detail System-Wide Requirements			H	
		Develop Technical Vision			H	
		Design the Solution			H	H
		Tailor the Process				H
		Set Up Tools		M		H
		Verify Tool Configuration and Installation				H
		Develop Product Documentation	H	H	H	
		Develop Training Materials		M		
		Develop Backout Plan		L		
		Plan Deployment		H		
		Review and Conform to Release Controls				H
Managing Planning and Managing Work (PMW)	Estimating (EST)	Tasks ITEC	EST 1.1 Develop high-level estimates to perform the work.	EST 2.1 Develop, keep updated, and use the scope of what is being estimated.	EST 2.2 Develop and keep updated estimates for the size of the solution.	EST 2.3 Based on size estimates, develop and record effort, duration, and cost estimates and their rationale for the solution.
		121 Documenta informação do projeto e caderno de encargos	x	x	x	x
		122 Planeamento temporal do projeto	x	x	x	x
		221 Análise das informações do projeto e caderno de encargos		x	x	x
		222 Definição de parâmetros e objetivos funcionais		x	x	x
		223 Análise da descrição de tarefas jira/trello		x	x	x
		224 Reuniões de ponto de situação e planeamento (semanal)		x	x	x
		231 Esquemática do Sw			x	x
		232 Programação da máquina			x	x
		241 Feedback e levantamento de dificuldades				x
		242 Atualização diária para o sistema de controlo de versões				x
		243 Registo da taxa de realização				x
		244 Registo de comentários dos motivos do ataraso (se houver)				x
		Coverage	I	S	S	S
		Priority	MP	LP	LP	MP
		Action is urgent?	JMP	SLP	SLP	SMP

APÊNDICE XIV

Managing		CMMI						
		ML1		ML2				
		MC 1.1 Record task completions.	MC 1.2 Identify and resolve issues.	MC 2.1 Track actual results against estimates for size, effort, schedule, resources, knowledge and skills, and budget.	MC 2.2 Track the involvement of identified stakeholders and commitments.	MC 2.3 Monitor the transition to operations and support.	MC 2.4 Take corrective actions when actual results differ significantly from planned results and manage to closure.	
Managing	Planning and Managing Work (PMW) Monitor and Control (MC)	Tasks OpenUp						
		Assess Results	M	M	H			
		Manage Iteration				H	H	M
		Request Change		H				H
		Tailor the Process			M	M		
		Execute Deployment Plan			H	M	H	
		Verify Successful Deployment			H	H	H	
		Install and Validate Infrastructure			M			H
		Review and Conform to Release Controls		M	H			M
		Managing	Planning and Managing Work (PMW) Monitor and Control (MC)	Tasks ITEC				
121 Documenta informação do projeto e caderno de encargos					x			
122 Planeamento temporal do projeto					x			
151 Recepção dos feedbacks de anomalia							x	x
152 Orientação para validar o SW							x	x
153 Orientação para testes elétricos e mecânicos							x	x
221 Análise das informações do projeto e caderno de encargos					x	x		
222 Definição de parametros e objetivos funcionais					x			
223 Análise da descrição de tarefas jira/trello					x	x		
224 Reuniões de ponto de situação e planeamento (semanal)					x	x		
241 Feedback e levantamento de dificuldades	x					x	x	
242 Atualização diária para o sistema de controlo de versões	x					x	x	
243 Registo da taxa de realização	x					x	x	
244 Registo de comentários dos motivos do ataraz (se houver)	x					x	x	
251 Validação do SW				x				x
252 Realização de teste elétricos e mecânicos				x				x
253 Retrabalho e revisão				x				x
271 Assistência técnica e suporte							x	x
272 Registo de reclamações							x	x
273 Solução de reclamações							x	x
Coverage	S			S	I	I	I	S
Priority	MP			HP	MP	LP	MP	HP
Action is urgent?	SMP			SHP	IMP	ILP	IMP	SHP

APÊNDICE XV

Managing	Planning and Managing Work (PMW)	CMMI											
		ML1		ML2									
		PR 1.1 Develop a list of tasks.	PLAN 1.2 Assign people to tasks.	PLAN 2.1 Develop and keep updated the approach for accomplishing the work.	PLAN 2.2 Plan for the knowledge and skills needed to perform the work.	PLAN 2.3 Based on recorded estimates, develop and keep the budget and schedule updated.	PLAN 2.4 Plan the involvement of identified stakeholders.	PLAN 2.5 Plan transition to operations and support.	PLAN 2.6 Ensure plans are feasible by reconciling available and estimated resources.	PLAN 2.7 Develop the project plan, ensure consistency among its elements, and keep it updated.	PLAN 2.8 Review plans and obtain commitments from affected stakeholders.		
Managing	Planning and Managing Work (PMW)	Tasks OpenUp											
		Refine the Architecture			M					M			
		Envision the Architecture			M								
		Assess Results								M		H	
		Manage Iteration			H		M	M	M	M	H	H	
		Plan Iteration	H		H		H	H	H	M	H		
		Plan Project	H	H	H	H	H	H			H		
		Tailor the Process						H				H	
		Verify Tool Configuration and Installation								M			
		Deliver end user Training				H							
		Deliver Support Training				H							
		Install and Validate Infrastructure										H	
		Plan Deployment							H				
		Review and Conform to Release Controls										H	
		Managing	Planning and Managing Work (PMW)	Tasks ITEC									
				121 Documenta informação do projeto e caderno de encargos					x	x		x	
				122 Planeamento temporal do projeto					x	x		x	
131 Reunião kick off							x	x		x			
132 Reunião ponto de situação e planeamento (semanal)							x	x		x			
141 Planeamento tarefas e subdivisão				x						x			
151 Recepção dos feedbacks de anomalia									x				
152 Orientação para validar o SW									x				
153 Orientação para testes elétricos e mecânicos									x				
210 Pré ou pós kick-off							x						
211 Recebe novo projeto							x						
221 Análise das informações do projeto e caderno de encargos	x									x	x	x	
222 Definição de parâmetros e objetivos funcionais	x									x	x	x	
223 Análise da descrição de tarefas jira/trello	x									x	x	x	
224 Reuniões de ponto de situação e planeamento (semanal)	x									x	x	x	
231 Esquematização do SW										x	x	x	
232 Programação da máquina										x	x	x	
241 Feedback e levantamento de dificuldades					x					x	x	x	
242 Atualização diária para o sistema de controlo de versões					x					x	x	x	
243 Registo taxa de realização					x					x	x	x	
244 Registo de comentários dos motivos do atraso (se houver)					x					x	x	x	
251 Validação do SW												x	
252 Realização de teste elétricos e mecânicos												x	
253 Retrabalho e revisão												x	
Coverage	I			S	I	W	I	S	I	S	S	S	
Priority	MP			HP	MP	MP	MP	LP	MP	HP	MP	MP	
Action is urgent?	IMP			SHP	IMP	WMP	IMP	SLP	IMP	SHP	SMP	SMP	

APÊNDICE XVI

Enabling	Managing Security and Safety (MSS)	Enabling Security (ESEC)	CMMI					
			ML1			ML2		
			ESEC 1.1 Identify and record security needs and issues.	ESEC 1.2 Address prioritized security needs and issues.	ESEC 2.1 Identify security needs, keep them updated, and use to develop a security approach and objectives.	ESEC 2.3 Develop, keep updated, and follow an approach to address mission, personnel, and process-related security needs.	ESEC 2.4 Develop, keep updated, and follow an approach to address cybersecurity needs.	
Tasks OpenUp								
			M	H	M	H	H	
			M	H	M	H	H	
					H	H	M	
			L	L	M			
				H		H	H	
				M			L	
								M
			M	M			H	
							H	
							H	
				L	L	H	L	
								M
								M
								M
								M
								H
								M
								H
								M
								M
								M
Enabling	Managing Security and Safety (MSS)	Enabling Security (ESEC)	CMMI					
			ML1			ML2		
			ESEC 1.1 Identify and record security needs and issues.	ESEC 1.2 Address prioritized security needs and issues.	ESEC 2.1 Identify security needs, keep them updated, and use to develop a security approach and objectives.	ESEC 2.2 Develop, keep updated, and follow an approach to address physical security needs.	ESEC 2.3 Develop, keep updated, and follow an approach to address mission, personnel, and process-related security needs.	ESEC 2.4 Develop, keep updated, and follow an approach to address cybersecurity needs.
Tasks ITEC								
				x	x	x	x	x
				x	x	x	x	x
				x	x	x	x	x
			x					
			x	x	x	x	x	x
			x	x	x	x	x	x
			x	x	x	x	x	x
			S	S	S	S	S	S
			HP	HP	MP	MP	MP	MP
			SHP	SHP	SMP	SMP	SMP	SMP

APÊNDICE XVII

		CMMI							
		ML1		ML2					
		MST 1.1 Identify and record security threats and vulnerabilities.	MST 1.2 Take actions to address security threats and vulnerabilities.	MST 2.1 Develop, keep updated, and follow an approach for handling security threats and vulnerabilities.	MST 2.2 Develop and keep updated criteria to evaluate security threats and vulnerabilities.	MST 2.3 Use recorded criteria to prioritize, monitor, and address the most critical security threats and vulnerabilities that arise during operations.	MST 2.4 Evaluate the effectiveness of the approach and actions taken to address critical security threats and vulnerabilities to the solution.		
Enabling	Managing Security and Safety (MSS) Managing Security Threats and Vulnerabilities (MST)	Tasks OpenUp							
		Refine the Architecture	L	H					
		Envision the Architecture	M	H					
		Identify and Outline Requirements	L	M					
		Run Developer Tests					M		
		Integrate and Create Build					M		
		Deploy the Process					H		
		Verify Tool Configuration and Installation					M		
		Create Test Cases	H			H			
		Implement Tests					H		
		Run Tests		H			H		
		Develop Product Documentation			H	H			
		Develop User Documentation			H	H			
		Develop Support Documentation			H	H			
		Deliver end user Training			H		H		
		Deliver Support Training			H		H		
		Develop Training Materials			H		H		
		Deliver Release Communications			H		H		
		Execute Backout Plan (if necessary)			H		H		
		Execute Deployment Plan			H		H		
		Develop Backout Plan	M			H			
		Plan Deployment				H			
		Review and Conform to Release Controls					M	L	
		Enabling	Managing Security and Safety (MSS) Managing Security Threats and Vulnerabilities (MST)	Tasks ITEC					
				151 Recepção dos feedbacks de anomalia	x	x		x	
				152 Orientação para validar o SW	x	x		x	
				153 Orientação para testes elétricos e mecânicos	x	x			
161 Verificação final				x			x		
162 Aprovação para instalação no cliente				x			x		
231 Esquemáticação do Sw					x				
232 Programação da máquina					x				
241 Feedback e levantamento de dificuldades					x				
242 Atualização diaria para o sistema de controlo de versões					x				
243 Registo da taxa de realização					x				
244 Registo de comentários dos motivos do ataraso (se houver)					x				
251 Validação do SW	x			x	x				
252 Realização de teste elétricos e mecânicos	x			x	x				
253 Retrabalho e revisão	x			x	x				
Coverage	I			I	S	S	W	I	
Priority	MP			HP	MP	MP	MP	LP	
Action is urgent?	IMP			IHP	SMP	SMP	WMP	ILP	

APÊNDICE XVIII

Enabling	Managing Business Resilience (MBR)	Causal Analysis and Resolution (CAR)	Tasks OpenUp	CMMI		
				ML1	ML2	
				CAR 1.1 Identify and address causes of selected outcomes.	CAR 2.1 Select outcomes for analysis.	CAR 2.2 Analyze and address causes of outcomes.
			Assess Results	H	H	H
			Manage Iteration			H
			Plan Iteration		M	
			Plan Project		M	
			Integrate and Create Build			M
			Design the Solution		M	
			Deploy the Process			H
			Verify Tool Configuration and Installation	H	H	H
			Run Tests	H	H	H
			Verify Successful Deployment		H	H
			Develop Release Communications			H
			Install and Validate Infrastructure	H	H	H
			Review and Conform to Release Controls	H	H	H
Enabling	Managing Business Resilience (MBR)	Causal Analysis and Resolution (CAR)	Tasks ITEC	CMMI		
				ML1	ML2	
				CAR 1.1 Identify and address causes of selected outcomes.	CAR 2.1 Select outcomes for analysis.	CAR 2.2 Analyze and address causes of outcomes.
			151 Recepção dos feedbacks de anomalia	X	X	X
			152 Orientação para validar o SW	X	X	X
			153 Orientação para testes elétricos e mecânicos	X	X	X
			221 Análise das informações do projeto e caderno de encargos	X	X	
			222 Definição de parametros e objetivos funcionais	X	X	
			223 Análise da descrição de tarefas jira/trello	X	X	
			224 Reuniões de ponto de situação e planeamento (semanal)	X	X	
			231 Esquematização do Sw		X	
			232 Programação da máquina		X	
			251 Validação do SW	X	X	X
			252 Realização de teste elétricos e mecânicos	X	X	X
			253 Retrabalho e revisão	X	X	X
			Coverage	I	I	I
			Priority	MP	MP	MP
			Action is urgent?	IMP	IMP	IMP

APÊNDICE XX

Enabling	Supporting Implementation	Decision Analysis and Resolution (DAR)	CMMI						
			ML1		DAR 2.1 Develop, keep updated, and use rules to determine when to follow a recorded process for criteria-based decisions.	ML2			
			DAR 1.1 Define and record the alternatives.	DAR 1.2 Make and record the decision.		DAR 2.2 Develop criteria for evaluating alternatives.	DAR 2.3 Identify alternative solutions.	DAR 2.4 Select evaluation methods.	DAR 2.5 Evaluate and select solutions using criteria and methods.
Tasks OpenUp									
	Refine the Architecture		M		H		M		
	Envision the Architecture		M	H	H		M		
	Assess Results							H	
	Manage Iteration							H	
	Plan Iteration						M		
	Plan Project			H	H		M		
	Request Change			H					
	Identify and Outline Requirements		H	H	H		M	L	
	Detail Use-Case Scenarios				H				
	Detail System-Wide Requirements				H				
	Develop Technical Vision				H				
	Implement Developer Tests							H	
	Integrate and Create Build			M	H				
	Deploy the Process							M	
	Tailor the Process		H	H	H			H	
	Create Test Cases							H	
	Implement Tests							H	
	Run Tests							H	
	Develop Product Documentation			H	H				
	Develop User Documentation				H				
	Develop Support Documentation				H				
	Develop Training Materials				H				
	Develop Backout Plan		M	M		H	H	M	
	Develop Release Communications		H	H	H				
	Install and Validate Infrastructure								
	Plan Deployment						M		
	Review and Conform to Release Controls							H	
Enabling	Supporting Implementation	Decision Analysis and Resolution (DAR)	CMMI						
			ML1		DAR 2.1 Develop, keep updated, and use rules to determine when to follow a recorded process for criteria-based decisions.	ML2			
			DAR 1.1 Define and record the alternatives.	DAR 1.2 Make and record the decision.		DAR 2.2 Develop criteria for evaluating alternatives.	DAR 2.3 Identify alternative solutions.	DAR 2.4 Select evaluation methods.	DAR 2.5 Evaluate and select solutions using criteria and methods.
Tasks ITEC									
	111 Recebe prdem de novo projeto								
	121 Documenta informação do projeto e caderno de encargos		x	x	x				
	122 Planeamento temporal do projeto		x	x					
	141 Planeamento tarefas e subdivisão			x					
	152 Orientação para validar o SW						x	x	
	153 Orientação para testes elétricos e mecânicos						x	x	
	221 Análise das informações do projeto e caderno de encargos		x	x	x				
	222 Definição de parâmetros e objetivos funcionais		x	x					
	223 Análise da descrição de tarefas jira/trello		x	x					
	224 Reuniões de ponto de situação e planeamento (semanal)		x	x					
	241 Feedback e levantamento de dificuldades		x					x	
	242 Atualização diária para o sistema de controlo de versões		x						
	243 Registo da taxa de realização		x					x	
	244 Registo de comentários dos motivos do atraso (se houver)							x	
	251 Validação do SW					x	x	x	
	252 Realização de teste elétricos e mecânicos					x	x	x	
	253 Retrabalho e revisão					x	x	x	
	Coverage		S	S	S	I	I	I	
	Priority		MP	HP	HP	MP	MP	MP	
	Action is urgent?		SMP	SHP	SHP	IMP	IMP	IMP	

APÊNDICE XXI

		CMMI										
		ML1		ML2								
Improving	Improving Performance (IMP)	Managing Performance and Measurement (MPM)	Tasks OpenUp	MPM 1.1 Collect measures and record performance.	MPM 1.2 Identify and address performance issues.	MPM 2.1 Derive and record measurement and performance objectives from selected business needs and objectives and keep them updated.	MPM 2.2 Develop, keep updated, and use operational definitions for measures.	MPM 2.3 Obtain specified measurement data according to the operational definitions.	MPM 2.4 Analyze performance and measurement data according to the operational definitions.	MPM 2.5 Store measurement data, measurement specifications, and analysis results according to the operational definitions.	MPM 2.6 Take actions to address identified issues with meeting measurement and performance objectives.	
							H	H	H	H	H	H
			Assess Results									
			Plan Project									
			Request Change								H	
			Identify and Outline Requirements				H	H				
			Detail System-Wide Requirements					H				
			Develop Technical Vision					H				
			Deploy the Process			H		H	H			
			Tailor the Process			H	H	H				
			Run Tests		H							
			Develop Product Documentation			H	H					
			Develop User Documentation					H				
			Develop Support Documentation				H					
			Execute Deployment Plan			H						
			Package the Release					H	H		H	
			Verify Successful Deployment						H			
			Plan Deployment					H				
			Review and Conform to Release Controls		H	H			H		H	
Improving		Improving Performance (IMP)	Managing Performance and Measurement (MPM)	Tasks ITEC	MPM 1.1 Collect measures and record performance.	MPM 1.2 Identify and address performance issues.	MPM 2.1 Derive and record measurement and performance objectives from selected business needs and objectives and keep them updated.	MPM 2.2 Develop, keep updated, and use operational definitions for measures.	MPM 2.3 Obtain specified measurement data according to the operational definitions.	MPM 2.4 Analyze performance and measurement data according to the operational definitions.	MPM 2.5 Store measurement data, measurement specifications, and analysis results according to the operational definitions.	MPM 2.6 Take actions to address identified issues with meeting measurement and performance objectives.
			121 Documenta informação do projeto e caderno de encargos				x					
			122 Planeamento temporal do projeto				x					
			151 Receção dos feedbacks de anomalia		x		x					x
			152 Orientação para validar o SW		x		x					x
			153 Orientação para testes elétricos e mecânicos		x		x					x
			161 Verificação final						x	x		
			162 Aprovação para instalação no cliente						x	x		
			221 Análise das informações do projeto e caderno de encargos	x	x		x					
			222 Definição de parametros e objetivos funcionais	x	x		x					
			223 Análise da descrição de tarefas jira/trello	x	x		x					
			224 Reuniões de ponto de situação e planeamento (semanal)	x	x		x					
			231 Esquemática do SW	x	x							
			232 Programação da máquina		x							
			241 Feedback e levantamento de dificuldades		x		x	x	x	x	x	x
			242 Atualização diária para o sistema de controlo de versões		x		x	x	x	x	x	x
			243 Registo da taxa de realização		x		x	x	x	x	x	x
			244 Registo de comentários dos motivos do ataraso (se houver)		x		x	x	x	x	x	x
			251 Validação do SW		x							
			252 Realização de teste elétricos e mecânicos		x							
			253 Retrabalho e revisão		x							
			Coverage	I	I	I	S	S	I	I	I	I
			Priority	IMP	IMP	LP	MP	MP	MP	MP	IMP	MP
			Action is urgent?	IMP	IMP	ILP	SMP	SMP	IMP	IMP	IMP	IMP

APÊNDICE XXII

Improving	Improving Performance (IMP)	Process Asset Development (PAD)	CMMI				
			Tasks OpenUp	ML1	ML2		
				PAD 1.1 Develop process assets to perform the work.	PAD 2.1 Determine what process assets will be needed to perform the work.	PAD 2.2 Develop, buy, or reuse process assets.	PAD 2.3 Make processes and assets available.
				H			
				H			
				H			
			M	M	H		
				M	H	H	
					H	H	
						H	
Improving	Improving Performance (IMP)	Process Asset Development (PAD)	CMMI				
			Tasks ITEC	ML1	ML2		
				PAD 1.1 Develop process assets to perform the work.	PAD 2.1 Determine what process assets will be needed to perform the work.	PAD 2.2 Develop, buy, or reuse process assets.	PAD 2.3 Make processes and assets available.
				x			
				x			
				x			
				x			
				x			
				x			
			x	x	x		
			x	x	x		
						x	
						x	
						x	
			S	S	H	H	
			HP	MP	MP	HP	
			SHP	SMP	HMP	HHP	

APÊNDICE XXIII

		CMMI					
		ML1			ML2		
Improving Performance (IMP)	Process Management (PCM)	Tasks OpenUp	PCM 1.1 Develop a support structure to provide process guidance, identify and fix process problems, and continuously improve processes.	PCM 1.2 Appraise the current process implementation and identify strengths and weaknesses.	PCM 1.3 Address improvement opportunities or process issues.	PCM 2.1 Identify improvements to the processes and process assets.	PCM 2.2 Develop, keep updated, and follow plans for implementing selected process improvements.
		Assess Results		H	H	H	
		Plan Project	H			H	H
		Design the Solution				M	M
		Deploy the Process				H	H
		Tailor the Process	H	H		H	H
		Set Up Tools					H
		Verify Tool Configuration and Installation					H
		Review and Conform to Release Controls				M	
				CMMI			
		ML1			ML2		
Improving Performance (IMP)	Process Management (PCM)	Tasks ITEC	PCM 1.1 Develop a support structure to provide process guidance, identify and fix process problems, and continuously improve processes.	PCM 1.2 Appraise the current process implementation and identify strengths and weaknesses.	PCM 1.3 Address improvement opportunities or process issues.	PCM 2.1 Identify improvements to the processes and process assets.	PCM 2.2 Develop, keep updated, and follow plans for implementing selected process improvements.
		121 Documenta informação do projeto e caderno de encargos	x	x			x
		122 Planeamento temporal do projeto	x	x			x
		141 Planeamento tarefas e subdivisão	x				x
		151 Receção dos feedbacks de anomalia	x	x			
		152 Orientação para validar o SW	x	x			
		153 Orientação para testes elétricos e mecânicos	x	x			
		161 Verificação final	x				
		162 Aprovação para instalação no cliente	x				
		241 Feedback e levantamento de dificuldades	x				x
		242 Atualização diária para o sistema de controlo de versões	x				x
		243 Registo da taxa de realização	x				x
		244 Registo de comentários dos motivos do ataraso (se houver)	x				x
		251 Validação do SW	x		x	x	x
		252 Realização de teste elétricos e mecânicos	x		x	x	x
		253 Retrabalho e revisão	x		x	x	x
		Coverage	I	I	S	S	I
		Priority	MP	MP	MP	LP	LP
		Action is urgent?	IMP	IMP	SMP	SLP	ILP

APÊNDICE XXIV

		CMMI					
		ML1	ML2	ML2	ML2	ML2	
Improving Sustaining Habit and Persistence (SHP) Governance (GOV)	Tasks OpenUp	GOV 1.1 Senior management identifies what is important for doing the work and defines the approach needed to accomplish the objectives of the organization.	GOV 2.1 Senior management defines, keeps updated, and communicates organizational directives for process implementation and performance improvement based on organization needs and objectives.	GOV 2.2 Senior management ensures resources and training are provided for developing, supporting, performing, improving, and evaluating adherence to expected processes.	GOV 2.3 Senior management identifies their information needs and uses the collected information to provide governance and oversight of effective process implementation and improvement.	GOV 2.4 Senior management assigns authority and holds people accountable for adhering to organization directives and achieving process implementation and performance improvement objectives.	
		Assess Results	H	H+	H+	H+	H+
		Manage Iteration	H	H+	H+	H+	H+
		Plan Iteration	H	H+	H+	H+	H+
		Plan Project	H	H+	H+	H+	H+
		Request Change	H	H+	H+	H+	H+
		Tailor the Process					H
		Develop Training Materials		H	H	H	H
		Deliver Release Communications		H	H	H	H
Improving Sustaining Habit and Persistence (SHP) Governance (GOV)	Tasks ITEC	GOV 1.1 Senior management identifies what is important for doing the work and defines the approach needed to accomplish the objectives of the organization.	GOV 2.1 Senior management defines, keeps updated, and communicates organizational directives for process implementation and performance improvement based on organization needs and objectives.	GOV 2.2 Senior management ensures resources and training are provided for developing, supporting, performing, improving, and evaluating adherence to expected processes.	GOV 2.3 Senior management identifies their information needs and uses the collected information to provide governance and oversight of effective process implementation and improvement.	GOV 2.4 Senior management assigns authority and holds people accountable for adhering to organization directives and achieving process implementation and performance improvement objectives.	
		111 Recebe prdem de novo projeto	x			x	
		121 Documenta informação do projeto e caderno de encargos	x	x	x	x	
		122 Planeamento temporal do projeto	x	x	x	x	
		131 Reunião kick off	x	x	x		
		132 Reunião ponto de situação e planeamento (semanal)		x	x		
		141 Planeamento tarefas e subdivisão					x
		151 Recepção dos feedbacks de anomalia				x	
		152 Orientação para validar o SW				x	
		153 Orientação para testes elétricos e mecânicos				x	
		161 Verificação final				x	
		162 Aprovação para instalação no cliente				x	
		224 Reuniões de ponto de situação e planeamento (semanal)			x		
		Coverage	S	I	I	S	S
		Priority	HP	MP	HP	MP	MP
		Action is urgent?	SMP	IMP	IHP	SMP	SMP

APÊNDICE XXV

Improving	Sustaining Habit and Persistence (SHP)	Implementation Infrastructure (II)	CMMI		
			ML1	ML2	
			II 1.1 Perform processes that address the intent of the Level 1 practices.	II 2.1 Provide sufficient resources, funding, and training for developing and performing processes.	II 2.2 Develop and keep processes updated, and verify they are being followed.
		Tasks OpenUp			
		Refine the Architecture	M		
		Assess Results			H
		Manage Iteration			H
		Plan Iteration			H
		Plan Project			H
		Implement Solution	H		
		Deploy the Process		M	H
		Set Up Tools	H		
		Develop Product Documentation		M	
		Develop User Documentation		M	
		Develop Support Documentation		M	
		Deliver end user Training		H	
		Deliver Support Training		M	
		Develop Training Materials		H	
		Package the Release			H
		Verify Successful Deployment			H
		Install and Validate Infrastructure	H		H
		Plan Deployment			M
		Review and Conform to Release Controls			H
Improving	Sustaining Habit and Persistence (SHP)	Implementation Infrastructure (II)	CMMI		
			ML1	ML2	
			II 1.1 Perform processes that address the intent of the Level 1 practices.	II 2.1 Provide sufficient resources, funding, and training for developing and performing processes.	II 2.2 Develop and keep processes updated, and verify they are being followed.
		Tasks ITEC			
		121 Documenta informação do projeto e caderno de encargos	x	x	x
		122 Planeamento temporal do projeto	x	x	x
		131 Reunião kick off			x
		132 Reunião ponto de situação e planemaneto (semanal)			x
		221 Análise das informações do projeto e caderno de encargos	x		x
		222 Definição de parametros e objetivos funcionais	x		x
		223 Análise da descrição de tarefas jira/trello	x		x
		224 Reuniões de ponto de situação e planeamento (semanal)	x		x
		231 Esquematização do Sw			x
		232 Programação da máquina			x
		241 Feedback e levantamento de dificuldades			x
		242 Atualização diária para o sistema de controlo de versões			x
		243 Registo da taxa de realização			x
		244 Registo de comentários dos motivos do ataraso (se houver)			x
		Coverage	S	I	S
		Priority	HP	MP	MP
		Action is urgent?	SHP	IMP	SMP