



Universidade do Minho
Escola de Engenharia

José Pedro Cerqueira Pinto

Caracterização de um *Agile Coordination Office* para empresas TSI

Dissertação de Mestrado

Mestrado Integrado em Engenharia e Gestão de Sistemas de
Informação

Trabalho efetuado sob a orientação de:

Professor Doutor Pedro Miguel Gonzalez Abreu Ribeiro

Outubro de 2018

DECLARAÇÃO

Nome: José Pedro Cerqueira Pinto

Endereço eletrónico: a72344@alunos.uminho.pt Telefone: +351 914948780

Cartão do Cidadão: 14914366

Título da dissertação: Caracterização de um *Agile Coordination Office* para empresas TSI

Orientador:

Professor Doutor Pedro Miguel Gonzalez Abreu Ribeiro

Ano de conclusão: 2018

Mestrado Integrado em Engenharia e Gestão de Sistemas de Informação

É AUTORIZADA A REPRODUÇÃO INTEGRAL DESTA DISSERTAÇÃO APENAS PARA EFEITOS DE INVESTIGAÇÃO, MEDIANTE DECLARAÇÃO ESCRITA DO INTERESSADO, QUE A TAL SE COMPROMETE.

Universidade do Minho, 22/10/2018

Assinatura:

AGRADECIMENTOS

Finalizada esta etapa, quero deixar um agradecimento a todos os que me acompanharam e apoiaram ao longo destes anos.

Ao meu orientador, o Professor Doutor Pedro Ribeiro, pelo apoio incansável e pela partilha do seu conhecimento e experiência comigo. Também quero agradecer pelos desafios novos que me foi colocando, os quais me incentivaram a procurar aprender sempre mais.

A todos os entrevistados por compartilharem a sua expertise e por contribuírem para o produto final desta dissertação.

Aos meus amigos de curso Francisco, Rafael e Sérgio, com quem partilhei muitos e bons momentos. Espero que tenham muito sucesso ao longo da vossa vida.

À minha família, sobretudo aos meus pais e irmão, por me terem apoiado e motivado ao longo de todo este percurso, assim como me apoiaram sempre em todas as etapas da minha vida.

À minha namorada, Vânia, por todo o apoio, motivação, paciência e por ter vivenciado todas as minhas angústias e conquistas ao longo deste percurso. Obrigado por tudo o que fizeste e fazes por mim.

Aos meus amigos que sempre estiveram e estão ao meu lado. Obrigado pelos momentos de descontração tão necessários durante estes últimos anos.

Finalmente, gostaria de agradecer a todos os colegas e professores do MIEGSI que contribuíram para a minha formação académica. Agradecimento que se estende à Universidade do Minho por me ter acolhido e por ter contribuído para o meu desenvolvimento pessoal.

RESUMO

A crescente competitividade do mercado levou a que as organizações realizassem vários projetos em simultâneo, para conseguirem manter um portefólio de produtos atualizado e reduzir o tempo de colocação desses produtos no mercado. Este aumento de projetos fez com que a sua gestão e coordenação se tornasse ainda mais complexa. Como resposta a este aumento de complexidade surgiu o conceito de *Project Management Office* (PMO), cuja principal função é facilitar a gestão e coordenação dos projetos da organização, e a partilha de recursos, metodologias, ferramentas e técnicas entre eles.

Desde a formulação do manifesto ágil em 2001, novas práticas ágeis de desenvolvimento de *software* começaram a ser formalizadas em metodologias como o Scrum, o XP e o Kanban.

No entanto, o PMO teve origem numa altura em que predominavam as abordagens tradicionais, sobretudo o modelo em cascata. Isto fez com que os PMOs fossem concebidos para suportar projetos que seguissem estes tipos de metodologias. Mas como atualmente os projetos ágeis são cada vez maiores e em maior quantidade, surge a necessidade de um gabinete especializado no suporte deste tipo de projetos. Atualmente, a literatura com caracterizações detalhadas deste tipo de estruturas é escassa.

Assim sendo, a finalidade desta dissertação consistiu em caracterizar um gabinete de gestão de projetos ágeis, tendo como base o funcionamento dos PMOs, os valores e princípios ágeis, e alguns casos de estudo de gabinetes semelhantes que já existam.

A metodologia escolhida para o desenvolvimento desta dissertação foi a *Design Science Research* complementada com diretrizes específicas para a pesquisa na área de sistemas de informação. O resultado esperado da aplicação desta metodologia é um artefacto que contribua para um novo conhecimento sobre a área em estudo, o que se verificou.

O resultado principal desta dissertação foi um modelo, adequado para empresas de TSI, de um gabinete de coordenação ágil adaptável ao contexto organizacional, que facilita a sua implementação gradual e que suporta os vários níveis de gestão. Designado como *Agile Coordination Office*, este modelo foi avaliado e validado pelos experientes profissionais entrevistados, assim como pelos revisores do artigo, no qual foi exposto à comunidade o trabalho desenvolvido.

Palavras-Chave: Gestão de Projetos, Gabinete Ágil de Projetos, PMO Ágil, Abordagens Ágeis, Escalabilidade Ágil

ABSTRACT

The increasing market competitiveness has led organizations to carry out several projects simultaneously, in order to maintain an up-to-date portfolio of products and reduce the time-to-market of these products. This increase in the number of projects has made the management and coordination of these projects even more complex. In response to this increase in complexity was created the concept of Project Management Office (PMO) whose main function is to facilitate the management and coordination of the organization's projects, and the sharing of resources, methodologies, tools and techniques between them.

Since the formulation of the agile manifesto in 2001, new agile software development practices began to be formalized in methodologies such as Scrum, XP and Kanban.

However, the PMO originated at a time when traditional approaches predominated were used such as waterfall model. This made the PMOs specially designed to support projects that followed these kinds of methodologies. However, as agile projects are increasing in size and quantity, the need arises for a specialized office that supports that kind of projects. The literature about detailed characterizations of this type of structures is still very limited.

Therefore, the purpose of this dissertation was to characterize a coordination office for agile projects, based on the functions of the traditional PMOs, the agile values and principles, and some proposals of agile offices that already existed.

The methodology chosen for the development of this dissertation was Design Science Research complemented with specific guidelines for information systems research. The expected result of the application of this methodology is an artifact that contributes to build new knowledge about the studied area, which was verified.

The main result obtained in this dissertation was a model for IST companies of an Agile Coordination Office adaptable to the organizational context that facilitates its gradual implementation and supports the multiple levels of management. This model was evaluated and validated by the experienced professionals interviewed, as well as by reviewers of the full paper where the work developed was exposed to the community.

KEYWORDS: Project Management, Agile Coordination Office, Agile PMO, Agile Approaches, Scaling Agile

ÍNDICE

Agradecimentos.....	iii
Resumo.....	v
Abstract.....	vii
Lista de Figuras.....	xiii
Lista de Tabelas	xv
Lista de Abreviaturas, Siglas e Acrónimos	xvii
1. Introdução.....	1
1.1 Enquadramento	1
1.2 Finalidade e objetivos	3
1.3 Abordagem metodológica	3
1.3.1 Estratégia de pesquisa.....	8
1.3.2 Seleção, análise e utilização de artigos.....	9
1.4 Estrutura do documento	14
2. Revisão de literatura	17
2.1 Gestão de projetos de TSI.....	17
2.1.1 Definição de Projeto, Programa e Portefólio.....	17
2.1.2 Definição de Gestão de projetos, programas e portefólios	18
2.1.3 Tecnologias e Sistemas de Informação.....	19
2.2 Estruturas organizacionais.....	21
2.3 Project Management Office.....	24
2.3.1 Definição e contextualização	24
2.3.2 Tipologias e modelos	25
2.3.3 Processo de implementação	29
2.4 Gestão tradicional e ágil de projetos.....	30
2.4.1 O modelo em cascata	30
2.4.2 O manifesto ágil	32
2.4.3 Tradicional vs ágil	33
2.4.4 Migração para metodologias ágeis	37

2.5	Abordagens ágeis.....	38
2.5.1	Scrum.....	38
2.5.2	Extreme Programming.....	44
2.5.3	Kanban.....	50
2.6	Desenvolvimento ágil de larga escala.....	54
3.	Conceptualização do <i>Agile Coordination Office</i>	59
3.1	Análise de propostas semelhantes.....	59
3.1.1	Agile Office.....	59
3.1.2	Agile PMO.....	61
3.1.3	The Lean-Agile PMO.....	62
3.1.4	Agile Practice Guide.....	64
3.1.5	Succeeding with Agile.....	65
3.1.6	Análise crítica.....	66
3.2	Desenvolvimento do modelo.....	68
3.3	Caracterização inicial do modelo.....	71
3.3.1	ACO básico.....	73
3.3.2	ACO avançado.....	77
3.4	Implementação do modelo.....	79
4.	Avaliação do modelo.....	81
4.1	Entrevistas.....	81
4.1.1	Desenvolvimento das entrevistas.....	83
4.1.2	Procedimentos para as entrevistas.....	84
4.1.3	Perfil dos entrevistados.....	85
4.2	Comentários ao modelo inicial.....	87
4.2.1	Sugestões.....	87
4.2.2	Validações.....	89
4.2.3	Limitações.....	90
4.3	Caracterização final do modelo.....	90
5.	Conclusões e trabalhos futuros.....	95

5.1	Conclusões	95
5.2	Trabalho futuro	97
	Referências	99
	Anexo I – Guião da entrevista.....	107
	Anexo II – Artigo publicado	109

LISTA DE FIGURAS

Figura 1- Tipos de contribuição de conhecimento da DSR (adaptado de (Gregor & Hevner, 2013))	4
Figura 2- Modelo do processo da Design Science Research (adaptado de (Vaishnavi et al., 2004))	5
Figura 3- Organização funcional (retirado de (Project Management Institute, 2013a))	22
Figura 4- Organização orientada a projetos (adaptado de (Project Management Institute, 2013a))	23
Figura 5- Organização matricial equilibrada (adaptado de (Project Management Institute, 2013a))	23
Figura 6- Frequência dos modelos de PMO nas tipologias (retirado de (Monteiro et al., 2016))	28
Figura 7- Fases de desenvolvimento de software do modelo de cascata	30
Figura 8- Metodologias ágeis mais usadas em 2017 (retirado de (Versionone.com, 2018))	38
Figura 9- Framework do Scrum (retirado de (Scrum.org, 2018))	44
Figura 10- Story Card (retirado de (Beck, 1999b))	48
Figura 11- Ciclo de vida do processo do XP (retirado de (Abrahamsson et al., 2002))	49
Figura 12- Exemplo de um quadro kanban (retirado de (Anderson & Carmichael, 2016))	54
Figura 13- Estrutura organizacional do Lean-Agile PMO (retirado de (Augustine & Cuellar, 2006))	63
Figura 14- Estrutura inicial do ACO	72
Figura 15- Vista dos papéis do ACO	73
Figura 16- Estrutura refinada do ACO	91

LISTA DE TABELAS

Tabela 1- Matriz Conceitos X Referências	10
Tabela 2- Desenvolvimento de software: Tradicional vs Ágil (adaptado de (Nerur et al., 2005))	34
Tabela 3- Situações ideais das abordagens ágeis e tradicionais (adaptado de (Boehm & Turner, 2003))	35
Tabela 4- Descrição dos 5 valores do Scrum	40
Tabela 5- Papéis e responsabilidades do Scrum	40
Tabela 6- Descrição dos 5 valores do XP	45
Tabela 7- Descrição dos 5 princípios fundamentais do XP	46
Tabela 8- Papéis e responsabilidades do XP	48
Tabela 9- Descrição dos 6 princípios fundamentais do Kanban	52
Tabela 10- Agenda de pesquisa sugerida para o desenvolvimento ágil de larga escala (retirado de (Dingsøyr & Moe, 2013))	55
Tabela 11- Princípios e práticas do Lean-Agile PMO (retirado de (Augustine & Cuellar, 2006))	64
Tabela 12- Práticas recolhidas	68
Tabela 13- Modos de implementação do ACO	79
Tabela 14- Princípios de desenvolvimento ágil de larga escala refletidos no modelo	80
Tabela 15- Perfis dos entrevistados A,B e C	86
Tabela 16- Sugestões dos entrevistados e respectivas alterações	87

LISTA DE ABREVIATURAS, SIGLAS E ACRÓNIMOS

APM	Association for Project Management
DSR	Design Science Research
IST	Information Systems and Technology
PMBOK ®	Project Management Body of Knowledge ¹
PMI ®	Project Management Institute ¹
PMO	Project Management Office
SAFe ®	Scaled Agile Framework ²
SI	Sistema de Informação
TI	Tecnologias de informação
TSI	Tecnologias e Sistemas de Informação
WIP	Work-in-progress
XP	Extreme Programming

¹ “PMBOK” e “PMI” são marcas registadas pelo Project Management Institute, Inc.

² “SAFe” é uma marca registadas pelo Scale Agile, Inc.

1. INTRODUÇÃO

Neste capítulo é apresentado o enquadramento com as motivações para o desenvolvimento desta dissertação, definida a finalidade e os respetivos objetivos para a cumprir, detalhada a abordagem metodológica utilizada, apresentada a estratégia de pesquisa, apresentada a síntese da utilização dos artigos e obras e, finalmente, é apresentada a estrutura do presente documento.

1.1 Enquadramento

Um projeto de pequena dimensão e simples pode ser concluído sem grande esforço adicional para além do necessário para o desenvolvimento do resultado desejado. Nesta dissertação interessam-nos projetos da área das tecnologias e sistemas de informação, sendo um dos tipos mais populares os de desenvolvimento de *software* (Gonçalves, Cruz, & Varjão, 2008).

No entanto, projetos de maior complexidade necessitam que os recursos que tem disponíveis sejam geridos da melhor forma possível (Kerzner, 2009). Esse papel é atribuído à atividade de gestão de projetos cujo objetivo é garantir que os resultados do projeto cumprem com os respetivos requisitos, por intermédio da aplicação de conhecimentos, técnicas, ferramentas e competências (Project Management Institute, 2013a).

O aumento do número de projetos que simultaneamente são realizados, na mesma organização, fez com que a complexidade da sua gestão também aumentasse. Este alargamento do número de projetos deve-se sobretudo às pressões do mercado que se verificam, como a pressão para a rápida renovação do portefólio de produtos e a pressão para reduzir o tempo de colocação dos produtos no mercado (Aubry, Hobbs, & Thuillier, 2007). Como resposta a este aumento de complexidade surgiu o conceito de Project Management Office (PMO), cujo objetivo é a gestão e coordenação centralizada de todos os projetos sobre sua alçada.

O leque de PMOs é amplo, existindo vários tipos que se diferenciam sobretudo quanto ao seu nível de controlo e influência nos projetos (Project Management Institute, 2013a). Para além disso, variam quanto ao nível de gestão em que atuam, existindo modelos de PMOs para o nível operacional (gestão de projetos), tático (gestão de programas) e estratégico (gestão de portefólios) (Monteiro, Santos, & Varajão, 2016).

Contudo, estas estruturas foram idealizadas numa altura em que o desenvolvimento de *software* ainda não tinha sido introduzido à mentalidade ágil, tendo surgido posteriormente com a publicação do

manifesto ágil. Ou seja, grande parte dos PMOs surgiram para cobrir as necessidades de gestão e coordenação de projetos de desenvolvimento tradicional.

Atualmente, o número de abordagens ágeis disponíveis é bastante alargado, dificultando a escolha e exploração extensiva de cada uma por parte dos potenciais interessados. No entanto, verifica-se que o Scrum é a metodologia mais utilizada (Versionone.com, 2018). Esta metodologia é por vezes combinada com práticas de outras metodologias ou métodos, resultando em abordagens híbridas. As mais usuais para este cruzamento são a metodologia *Extreme Programming* (XP) e o método *Kanban* (Versionone.com, 2018). Cada uma destas abordagens ágeis tem o seu nível de prescrição de papéis, eventos, princípios e boas práticas. Contudo, atualmente os praticantes estão menos preocupados em como adotar o ágil e mais preocupados em como mantê-lo, sendo que esta sustentabilidade é ainda um tema subdesenvolvido (Gregory, Barroca, Sharp, Deshpande, & Taylor, 2016).

A documentação e literatura afetas às abordagens ágeis não apresentam qual o papel e/ou importância do PMO para a gestão de projetos ágeis (Cohn, 2010). Esta lacuna na literatura foi-se perpetuando até aos dias de hoje, evidenciando-se nas dificuldades e desafios identificados no estudo do desenvolvimento ágil de larga escala.

Além do mais, também se verificou que a literatura sobre casos de estudo de PMOs no contexto de projetos ágeis é muito limitada. Especialmente no que toca a soluções e caracterizações de PMOs ágeis, sendo que os casos de estudo existentes apresentam-se limitados quanto aos níveis de gestão em que atuam, à estrutura que sugerem para esse suporte e/ou ao contexto em que podem ser implementados. Assim sendo, verificou-se a existência de duas lacunas na literatura. A primeira, quanto à falta de perceção que as abordagens ágeis tem sobre a importância do PMO e, a segunda, quanto à escassez de casos de estudo de PMO ágeis.

Segundo um estudo do Standish Group (Hastie & Wojewoda, 2015), embora a taxa de sucesso dos projetos ágeis grandes (18%) tenha sido superior à taxa de sucesso dos projetos tradicionais grandes (3%), era consideravelmente inferior à taxa de sucesso dos projetos ágeis pequenos (58%), para os quais as abordagens ágeis foram inicialmente concebidas. Isto revela que o recurso às abordagens ágeis, por si só, pode melhorar a gestão dos projetos, mas isso não implica uma melhoria com o mesmo impacto da gestão feita a níveis superiores.

Com base nestas premissas, o propósito desta dissertação é contribuir com conhecimento para a área sob estudo, através do desenvolvimento de uma solução capaz de suportar os projetos ágeis, assim como os programas e portefólios dos quais fazem parte, para ser aplicado no contexto das empresas de TSI.

1.2 Finalidade e objetivos

A finalidade desta dissertação de mestrado é o estudo da seguinte questão de investigação: “Quais as características de um *Agile Coordination Office* capaz de apoiar a gestão de projetos ágeis em empresas de TSI?”.

Para alcançar com sucesso esta finalidade, isto é, para responder à questão de investigação, nesta dissertação visa-se desenvolver um modelo de um *Agile Coordination Office* (ACO) que seja capaz de suportar a gestão de projetos ágeis de empresas de TSI.

Para concretizar esta finalidade definiram-se os seguintes objetivos estruturantes:

- Revisão de literatura sobre os conceitos relevantes;
- Análise de propostas semelhantes;
- Conceção de uma proposta do ACO;
- Exposição e avaliação proposta;
- Caracterização da proposta refinada do ACO.

Para alcançar estes objetivos procedeu-se a um conjunto de etapas que se enquadram com as da abordagem metodológica selecionada e descrita de seguida.

1.3 Abordagem metodológica

A metodologia de investigação escolhida para o desenvolvimento desta dissertação foi a *Design Science Research* (ou abreviadamente DSR). Segundo Vaishnavi, Kuechler e Petter (2004) o conhecimento resultante desta metodologia manifesta-se na forma de artefactos, modelos, *frameworks*, arquiteturas, métodos, instanciações, teorias de *design* ou princípios de *design*. Os mesmos autores também afirmam que, conforme ilustra a Figura 1, esse conhecimento resultante pode ter vários tipos de contribuição, variando em relação a dois critérios: (1) Maturidade do domínio do problema e (2) maturidade do domínio das soluções.

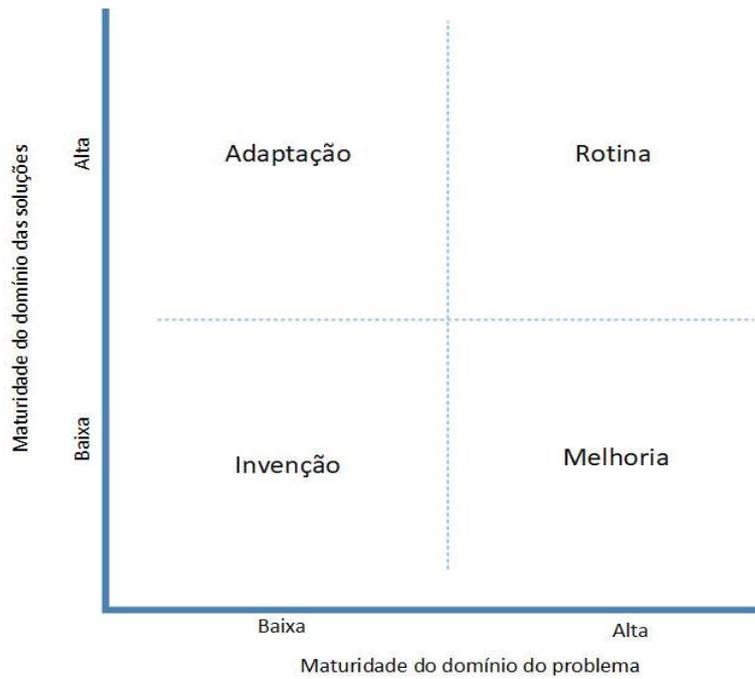


Figura 1- Tipos de contribuição de conhecimento da DSR (adaptado de (Gregor & Hevner, 2013))

Tendo em conta o enquadramento desta dissertação, a contribuição de conhecimento esperada caracteriza-se por uma maturidade do domínio do problema moderada, visto que se trata de um tema com alguma literatura relacionada, e por uma maturidade do domínio da solução baixa, visto que existem poucas soluções semelhantes. Assim sendo espera-se que o contributo desta dissertação seja do tipo invenção ou melhoria.

Para complementar a metodologia escolhida, recorreu-se a um artigo sobre a DSR aplicada à investigação na disciplina de sistemas de informação, que recomenda um conjunto de diretrizes a ter em conta no momento de por em prática a metodologia (Hevner, March, Park, & Ram, 2004). De seguida encontram-se brevemente descritas cada uma das 7 diretrizes propostas:

1. **Design as an Artifact.** O resultado da aplicação da DSR deve ser um artefacto claro, permitindo que possa ser aplicado num contexto apropriado;
2. **Problem Relevance.** A DSR visa o desenvolvimento de soluções para os problemas relevantes de negócio. A relevância do problema de investigação deve ser exposta;
3. **Design Evaluation.** A qualidade e utilidade do artefacto concebido devem ser demonstradas através de rigorosos métodos de avaliação;
4. **Research Contributions.** O contributo que advém deste tipo de pesquisa deve ser irrefutavelmente claro e verificável;

5. **Research Rigor.** As atividades de construção e avaliação do artefacto devem ser realizadas com base em métodos rigorosos;
6. **Design as a Search Process:** *Design Science Research* é por natureza um processo iterativo e a sua procura pelo artefacto mais eficaz requer a utilização de todos os meios disponíveis para atingir os propósitos desejados;
7. **Communication of Research.** É preciso ter em conta que os resultados devem ser apresentados quer a audiências orientada para as tecnologias, quer a audiências orientada para a gestão.

Estas diretrizes foram aplicadas durante a dissertação em sintonia com a metodologia DSR cujo processo se divide em cinco etapas (Vaishnavi et al., 2004): a Percepção do Problema, a Sugestão, o Desenvolvimento, a Avaliação e a Conclusão (ver Figura 2).

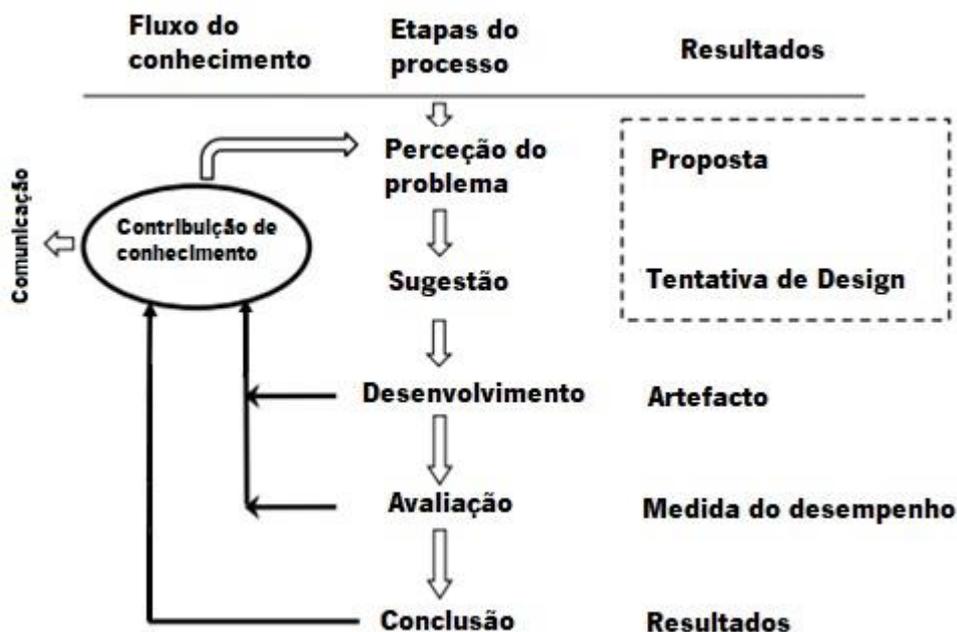


Figura 2- Modelo do processo da Design Science Research (adaptado de (Vaishnavi et al., 2004))

Segundo Vaishnavi, Kuechler e Petter (2004) estas etapas podem ser, resumidamente, descritas da seguinte forma:

- **Percepção do problema:** Nesta primeira fase é conduzida uma revisão de literatura, de fontes diversificadas, para ampliar o conhecimento na área a ser investigada promovendo a consciencialização do problema de pesquisa. Como resultado desta fase deve surgir uma proposta de um esforço de pesquisa com finalidade e objetivos definidos.

- **Sugestão:** Esta fase sucede à fase de Percepção do problema. Nela é feita a primeira Tentativa de *Design* para a proposta de esforço concebida na fase anterior. Espera-se um conhecimento alargado sobre o assunto a ser investigado para que a solução sugerida esteja de acordo com os objetivos de investigação. Se depois de colocado um esforço considerável no problema não surgir pelo menos uma ideia de uma solução desse problema, a proposta deve ser deixada de lado. Esta é a fase criativa do processo onde novas funcionalidades são projetadas com base numa nova configuração de elementos existentes ou de elementos novos e existentes.
- **Desenvolvimento:** Depois de concebido o protótipo na fase de Sugestão, nesta fase o protótipo é desenvolvido e implementado.
- **Avaliação:** Nesta fase, depois de já estar construído, o artefacto é avaliado de acordo com os critérios definidos na fase inicial. Deviações das expectativas devem tentar ser explicadas. De acordo com os resultados obtidos pode-se identificar trabalhos futuros.
- **Conclusão:** Nesta última fase, já com os resultados escritos e consolidados, retiram-se as conclusões relativamente ao conhecimento ganho nesta pesquisa, podendo estas ser categorizadas de duas formas: (1) o conhecimento é firme logo o artefacto pode ser aplicado fiável e repetidamente ou (2) o conhecimento apresenta anomalias comportamentais podendo servir como assunto para futuras pesquisas.

O motivo que levou à adoção desta metodologia foi o facto de os resultados esperados e o fluxo de processos serem bem definidos. A flexibilidade que o fluxo de processos providencia ao permitir voltar a fases anteriores também contribuiu para adoção desta metodologia.

Para além disso, foi decidido adotar um conjunto de diretrizes pois, nesta metodologia, o conhecimento e o entendimento de um problema e da sua solução é adquirido através da construção e aplicação de um artefacto. As diretrizes visam garantir que estas premissas da metodologia sejam cumpridas.

Para perceber melhor a aplicação da metodologia e das diretrizes, de seguida é apresentado um resumo de como foram utilizadas durante o desenvolvimento do trabalho.

Na primeira etapa desta dissertação, procedeu-se a uma revisão de literatura de conceitos relevantes, nomeadamente, acerca dos conceitos de gestão de projetos, de PMOs, da diferença entre abordagens tradicionais e ágeis de gerir projetos, do funcionamento das próprias abordagens e do estado atual do desenvolvimento ágil de larga escala. Esta etapa permitiu identificar as motivações para a investigação e, conseqüentemente definir a finalidade e objetivos. Esta etapa é mapeada com a primeira fase da DSR, a percepção do problema, tendo como resultados: (1) a formalização da questão de

investigação e respetivos objetivos estruturantes, detalhados no subcapítulo 1.2 - Finalidade e objetivos e (2) a descrição da revisão de literatura apresentada no capítulo 2 - Revisão de literatura

Com base nesta revisão de literatura e com o propósito de investigação devidamente fundamentado, iniciou-se a segunda etapa, o processo de conceptualização, do qual resultou um primeiro modelo do ACO. Este esboço inicial encontra-se descrito no subcapítulo 3.2 - Desenvolvimento do modelo. Esta etapa alinha-se com a fase de sugestão da DSR.

Essa sugestão inicial deu origem à etapa seguinte, a conceptualização e desenvolvimento do artefacto, nomeadamente, o modelo do ACO. Esta sugestão foi complementada com a análise dos casos de estudo relacionados. Esta atividade é mapeada com a fase de desenvolvimento da DSR e o resultado encontra-se descrito ao longo do capítulo 3 - Conceptualização do *Agile Coordination Office*.

A etapa seguinte foi a avaliação do artefacto desenvolvido. Para tal decidiu-se submeter o modelo, em formato de artigo, para uma conferência de referência na área de gestão de projetos visando obter o parecer de especialistas na área. O artigo foi revisto e aceite para publicação, portanto, foi comunicada uma contribuição de conhecimento intermediária, conforme sugere a metodologia DSR. Esta etapa mapeia-se com a fase de avaliação da DSR estando descrita no capítulo 4 - Avaliação do modelo. O resumo do artigo científico pode ser consultado no Anexo II – Artigo publicado (Pinto & Ribeiro, 2018).

No entanto decidiu-se efetuar uma segunda avaliação, desta vez com a perspetiva dos próprios praticantes. Já com o modelo refinado pela avaliação anterior, definiram-se os requisitos sobre o perfil do entrevistado desejável, o tipo de investigação adequado para esta etapa e avaliou-se outra vez o modelo. Depois analisaram-se os dados recolhidos, refinando e validando mais uma vez o modelo. O resultado desta etapa encontra-se descrito no capítulo 4 - Avaliação do modelo.

Finalmente, a última etapa consistiu na redação de conclusões do trabalho desenvolvido e na recomendação de trabalhos futuros. Esta etapa é mapeada com a fase de conclusão da DSR e o seu resultado encontra-se descrito no capítulo 5 - Conclusões e trabalhos futuros.

O resultado foi um artefacto bem definido (diretriz 1), o modelo do ACO, aplicado a um problema específico (diretriz 2), o suporte da gestão de projetos ágeis. O artefacto desenvolvido foi avaliado quanto à sua utilidade através de métodos bem definidos (diretriz 3), nomeadamente, através da avaliação dos revisores do artigo e através das entrevistas aos praticantes experientes. Tendo em conta que as poucas propostas semelhantes, o artefacto desenvolvido constitui uma solução inovadora e melhorada de resolver um problema já conhecido (diretriz 4), a caracterização um PMO ágil. A solução alcançada baseia-se em conhecimento bem fundamentado pela literatura estando rigorosamente descrita e representada na revisão de literatura e na conceptualização do ACO (diretriz 5). A necessidade de

organizar as práticas do ACO promoveu a procura de conhecimentos (diretriz 6), nomeadamente os níveis de gestão, a relação entre as práticas e as preocupações do desenvolvimento de larga escala.

Finalmente, os resultados da pesquisa foram comunicados adequadamente quer para audiências técnicas (pessoas interessadas em estender ou implementar o artefacto) quer para audiências de gestão (pessoas interessadas em estudar ou praticantes que ponderem utilizar o artefacto), através deste relatório de dissertação e previamente através do artigo publicado (diretriz 7).

1.3.1 Estratégia de pesquisa

Para realizar a revisão de literatura desta dissertação foram feitas várias pesquisas em vários motores de busca académicos. Num momento inicial, foi utilizada a plataforma académica do *Google*, o *Google Scholar*, para ter uma noção da quantidade e diversidade de potenciais artigos a serem analisados. Depois desta primeira, abordagem, aumentou-se o número de plataformas utilizadas passando-se a utilizar também o *ScienceDirect*, o *IEEEExplore*, o *ResearchGate* e o *RepositóriUM*. Alguns dos livros referidos nesta dissertação foram consultados fisicamente.

Devido ao número elevado de documentos encontrados foi necessário filtrar os que mais interessavam. O processo de seleção está explicado mais à frente neste subcapítulo. No entanto, no momento da pesquisa priorizaram-se os artigos conforme os jornais, revistas ou conferências em que foram publicados e o respetivo número de citações.

As pesquisas para adquirir um conhecimento abrangente do estado da arte foram feitas através de determinados termos e expressões compostas. De seguida encontram-se enumerados os termos, mais relevantes, utilizados durante estas pesquisas:

- “Gestão de projeto” e “*Project management*”;
- “Metodologias ágeis” e “*Agile methodologies*”;
- “Sistema de Informação” e “*Information system*”;
- “Gabinete ágil” e “*Agile office*”;
- “Manifesto ágil” e “*Agile Manifesto*”;
- “Tecnologia de Informação” e “*Information technology*”;
- “Estruturas organizacionais” e “*Organizational structures*”;
- “Gabinete de gestão de projetos” e “*Project Management Office*”;
- “Tipos de PMO’s” e “*Types of PMO’s*”;
- “Papéis do PMO” e “*PMO roles*”;
- “Coordenação de projetos” e “*Project coordination*”;

- “Coordenação de PMOs” e “*PMO Coordination*”;

Durante as pesquisas, as expressões compostas mais relevantes utilizadas foram as seguintes:

- “Gestão ágil vs tradicional” e “*Agile vs tradicional management*”;
- “Definição de Gestão de projetos” e “*Definition of project management*”;
- “Desenvolvimento de *software* ágil e tradicional” e “*Agile and traditional software development*”;
- “Escalabilidade ágil” e “*Scaling agile*”;
- “Caracterização de gabinetes ágeis” e “*Characterization of agile offices*”.

1.3.2 Seleção, análise e utilização de artigos

Foram utilizados três critérios para selecionar os artigos que mais se adequavam a esta investigação. O primeiro critério de seleção foi o título do artigo, permitindo uma filtragem mais grosseira permitindo identificar quais interessavam e quais podiam ser descartados.

O segundo critério foi o resumo dos artigos. Este critério funciona como um complemento para o primeiro pois permitiu, de forma sucinta, perceber o que cada artigo tratava. Da aplicação destes dois critérios resultou um conjunto de artigos para análise. No entanto, mesmo sendo identificado como um artigo interessante, haviam dúvidas quanto ao conteúdo pelo que se tinha de passar para o terceiro critério.

O terceiro critério consistia na leitura integral ou parcial do artigo, para retirar as ideias intrínsecas no próprio texto. Este critério foi utilizado sobretudo em livros, permitindo identificar os capítulos que mais interessavam no âmbito desta investigação.

Todos os artigos, livros, procedimentos de conferências e *websites* consultados e utilizados durante a realização desta dissertação estão devidamente identificados no capítulo das Referências.

A Tabela 1 foi desenvolvida com o intuito de facilitar a visualização de como estes foram utilizados e contribuíram para a dissertação. Nesta tabela são cruzados os conceitos abordados na dissertação com as referências utilizadas para os fundamentar. Contudo, importa realçar que o artigo decorrente do primeiro momento de avaliação, embora se encontre nas referências, não se encontra nesta tabela pois trata-se de um resultado intermédio desta dissertação e não de um fundamento para a mesma.

Capítulo 1- Introdução

Tabela 1- Matriz Conceitos X Referências

Referência	Conceitos							
	Gestão de projetos de TSI	Estrutura organizacional	PMO	Gestão de projetos tradicional e ágil	Abordagens ágeis	Agilidade de larga escala	Análise de propostas semelhantes	Metodologia de investigação
(Abrahamsson, Salo, Ronkainen, & Warsta, 2002)					X			
(Aghina, Lackey, Lurie, & Murarka, 2018)		X						
(Alqudah & Razali, 2016)						X		
(Alsadeq, 2011)			X					
(Alter, 1992)	X							
(Amaral, 1994)	X							
(Anderson & Carmichael, 2016)					X			
(Anderson, 2010)					X			
(APM, 2012)	X							
(Aubry et al., 2007)			X					
(Aubry, Müller, Hobbs, & Blomquist, 2010)			X					
(Augustine & Cuellar, 2006)							X	
(Beck & Andres, 2004)					X			
(Beck et al., 2001)				X				
(Beck, 1999a)					X			
(Beck, 1999b)					X			
(Boehm & Turner, 2003)				X				
(Boehm, 1988)				X				
(Boehm, 2002)				X				

Capítulo 1- Introdução

Conceitos								
Referência	Gestão de projetos de TSI	Estrutura organizacional	PMO	Gestão de projetos tradicional e ágil	Abordagens ágeis	Agilidade de larga escala	Análise de propostas semelhantes	Metodologia de investigação
(Buckingham, Hirschheim, Land, & Tully, 1987)	X							
(Carvalho, 2000)	X							
(Cohen & Crabtree, 2006)								X
(Cohn, 2010)		X					X	
(Crawford, 2010)			X					
(Desouza & Evaristo, 2006)			X					
(Dikert, Paasivaara, & Lassenius, 2016)						X		
(Dingsøy, Fægri, & Itkonen, 2014)						X		
(Dingsøy & Moe, 2013)						X		
(Dingsøy & Moe, 2014)						X		
(Dingsøy, Nerur, Balijepally, & Moe, 2012)				X				
(Dingsøy, Rolland, Moe, & Seim, 2017)						X		
(Falkenberg et al., 1998)	X							
(Fernandes & Machado, 2016)				X				
(Ferreira, Tereso, Ribeiro, Fernandes, & Loureiro, 2013)	X							
(Fontana & Frey, 2000)								X
(Freudenberg & Sharp, 2010)						X		

Capítulo 1- Introdução

Conceitos								
Referência	Gestão de projetos de TSI	Estrutura organizacional	PMO	Gestão de projetos tradicional e ágil	Abordagens ágeis	Agilidade de larga escala	Análise de propostas semelhantes	Metodologia de investigação
(Gonçalves et al., 2008)	X							
(Gregor & Hevner, 2013)								X
(Gregory et al., 2016)						X		
(Hastie & Wojewoda, 2015)				X				
(Hevner et al., 2004)								X
(Hill, 2008)			X					
(Hobbs & Aubry, 2007)			X					
(Hove & Anda, 2005)								X
(Jalal & Koosha, 2015)		X						
(Jones, 2013)		X						
(Hubbard & Bolles, 2015)			X					
(Kerzner, 2009)	X							
(Monteiro, Santos e Varajão, 2016)			X					
(Nerur, Mahapatra, & Mangalaaraj, 2005)				X				
(OGC, 2009)	X							
(Paasivaara, Lassenius, & Heikkilä, 2012)						X		
(Poppendieck & Poppendieck, 2003)					X			
(Power, 2011)							X	
(Project Management Institute, 2008)	X							

Capítulo 1- Introdução

Referência	Conceitos							
	Gestão de projetos de TSI	Estrutura organizacional	PMO	Gestão de projetos tradicional e ágil	Abordagens ágeis	Agilidade de larga escala	Análise de propostas semelhantes	Metodologia de investigação
(Project Management Institute, 2013a)	X	X	X					
(Project Management Institute, 2013b)			X					
(Project Management Institute, 2013c)	X							
(Project Management Institute, 2017)							X	
(Reifer, Maurer, & Erdogmus, 2003)						X		
(Royce, 1970)				X				
(Schwaber, 1994)					X			
(Schwaber & Beedle, 2002)					X			
(Scrum.org, 2018)					X			
(Serrador & Pinto, 2015)				X				
(Singh, 2009)			X					
(Sliger, 2007)							X	
(Soares & Amaral, 2014)	X							
(Soares, 1998)	X							
(Sommerville, 2010)				X				
(Stellman & Greene, 2014)					X			
(Sutherland & Schwaber, 2016)					X			
(Sutherland, 2016)					X			
(Takeuchi & Nonaka, 1986)					X			

Referência	Conceitos							
	Gestão de projetos de TSI	Estrutura organizacional	PMO	Gestão de projetos tradicional e ágil	Abordagens ágeis	Agilidade de larga escala	Análise de propostas semelhantes	Metodologia de investigação
(Tengshe & Noble, 2007)							X	
(Vaidya, 2014)			X					
(Vaishnavi et al., 2004)								X
(Varajão, Domingues, Ribeiro, & Paiva, 2014)				X				
(Versionone.com, 2018)					X			
(Ward, Griffiths, & Whitmore, 1990)	X							
(Williams & Cockburn, 2003)				X				

1.4 Estrutura do documento

Este relatório encontra-se dividido em 5 capítulos. O primeiro capítulo é a introdução desta dissertação, onde é feito o enquadramento relativo ao tema de investigação. Ou seja, é neste capítulo que se justifica a relevância da investigação e se definem a finalidade, objetivos e resultados esperados. Também é explicada a abordagem metodológica selecionada e apresentada a estrutura do presente documento.

No segundo capítulo faz-se o enquadramento conceptual e é onde se relatam os resultados da revisão de literatura, expondo os principais conceitos relevantes para esta dissertação, nomeadamente, a gestão de projetos de TSI, as estruturas organizacionais, o *Project Management Office* (PMO), a gestão tradicional e ágil de projetos, as abordagens ágeis e o desenvolvimento ágil de larga escala.

O terceiro capítulo apresenta o modelo desenvolvido, começando por apresentar os casos de estudo semelhantes e uma análise crítica dos mesmos. Posteriormente apresenta-se o processo de desenvolvimento do ACO, descreve-se o modelo pormenorizadamente e, finalmente, explora-se os cenários e procedimentos para a sua implementação.

O quarto capítulo apresenta a avaliação feita ao modelo, descrevendo os métodos utilizados para tal efeito. Os resultados dessa avaliação também são analisados e discutidos, estando organizados em

sugestões, validações e limitações. Ainda neste capítulo são explicadas as transformações resultantes dessas avaliações.

O quinto capítulo apresenta as conclusões sobre o trabalho desenvolvido, tendo em conta os objetivos propostos inicialmente e os resultados obtidos. Também é descrito de que modo esta dissertação contribuiu para o avanço do conhecimento na área em que está inserida. Para além disso, são também apresentadas as limitações identificadas no decorrer da dissertação. Este capítulo é finalizado com a recomendação de trabalhos a serem desenvolvidos futuramente, de modo a complementarem e darem seguimento ao trabalho resultante da presente investigação.

Finalizados os cinco capítulos, surgem todas as referências bibliográficas que foram utilizadas durante esta dissertação.

Finalmente, como anexos deste relatório surgem o guião de entrevistas utilizado para a avaliação do modelo e o *abstract* do artigo científico publicado.

2. REVISÃO DE LITERATURA

Neste capítulo são explicados os conceitos relevantes para o desenvolvimento desta dissertação, tendo sido feito um detalhado levantamento do estado da arte.

Este capítulo encontra-se dividido em 6 subcapítulos que abordam os conceitos mais proeminentes em que se focou a revisão de literatura, nomeadamente, a gestão de projetos de TSI, as estruturas organizacionais, os *Project Management Offices* (PMOs), a comparação entre gestão tradicional e ágil de projetos, as abordagens ágeis mais utilizadas e o estado atual do desenvolvimento ágil de larga escala.

2.1 Gestão de projetos de TSI

2.1.1 Definição de Projeto, Programa e Portefólio

Projeto é definido como o esforço temporário e progressivo, com o objetivo de criar um produto, serviço ou resultado único (Project Management Institute, 2013a). Considera-se um evento temporário porque tem um início e um fim bem definidos, sendo portanto um evento finito.

Diz-se que é elaborado progressivamente porque, ao longo do tempo, todos os projetos são elaborados por um conjunto de fases. A este conjunto de fases dá-se o nome de ciclo de vida que, segundo o PMI (2013a), inclui cinco fases distintas, nomeadamente a iniciação, o planeamento, a execução, a monitorização e controlo, e o encerramento.

O produto ou serviço resultante diz-se único devido à natureza heterogénea dos projetos. O resultado é, de alguma forma, diferente de todos os outros produtos ou serviços semelhantes, devido à variedade de condições, contexto e âmbito que caracterizam cada projeto.

Quanto aos projetos na área dos sistemas de informação, Gonçalves, Cruz e Varajão (2008) consideram que existem três tipos de projetos: projetos de desenvolvimento de *software*, projetos de desenvolvimento do produto e projetos híbridos. Cada um destes tipos de projeto apresenta um ciclo de vida composto por quatro fases distintas, nomeadamente a conceção, o desenvolvimento, a implementação e o fechamento.

Para além disso, importa também entender os conceitos de programas e portefólios assim como a relação que tem entre si e com o conceito de projeto.

Um programa é composto por vários projetos relacionados que são iniciados durante o ciclo de vida do programa e são gerenciados de forma coordenada (Project Management Institute, 2013c).

Já um portfólio pode ser definido como uma coleção de projetos e/ou programas, e outros trabalhos que são agrupados para facilitar o gerenciamento eficaz desse trabalho, a fim de atender os objetivos estratégicos de negócios (Project Management Institute, 2008).

2.1.2 Definição de Gestão de projetos, programas e portfólios

A gestão de projetos tem vindo a desempenhar um papel central na gestão das organizações em quase todos os campos da atividade humana (Aubry et al., 2010). Como tal, vários grupos de trabalho tem vindo a ser formados para desenvolverem referenciais que contribuam para o amadurecimento desta área, como é o caso do referencial PMBOK desenvolvido pelo Project Management Institute (2013a).

Para o Project Management Institute (2013a), a gestão de projetos é definida como sendo “ a aplicação de conhecimento, habilidades, ferramentas e técnicas para garantir que as atividades dos projetos cumprem os requisitos do projeto”.

No mesmo guia é referido que a gestão de projetos é conseguida através da aplicação e integração dos 47 processos da gestão de projetos, de 10 áreas de conhecimento distintas. Dependendo do contexto de cada projeto, estas áreas de conhecimento podem ter maior ou menor relevância para a gestão desse mesmo projeto. Engloba atividades como a identificação de requisitos, a gestão do orçamento, do calendário, da qualidade, dos riscos, dos recursos, do âmbito e dos requisitos. Todas as atividades envolvem vários *stakeholders*, que possuem necessidades e interesses diferentes. Cabe à gestão de projetos garantir a comunicação entre esses *stakeholders*, tendo em conta as suas diferentes necessidades, preocupações e expectativas, gerindo-os de modo a que cumpram os requisitos do projeto.

Existem outras definições de gestão de projetos que são populares na literatura. Segundo a APM (2012), a gestão de projetos é a aplicação de processos, métodos, conhecimento, competências e experiência com o intuito de atingir todos os objetivos do projeto.

Outra definição é dada pelo referencial Prince2 (OGC, 2009) que diz que a gestão de projetos é o conjunto das atividades de planejar, delegar, monitorizar e controlar todos os aspetos de um projeto, incluindo a motivação de todas as partes envolvidas. Estas atividades visam alcançar os objetivos de um projeto dentro do tempo, custo qualidade, âmbito, benefícios e riscos esperados.

Como é possível constatar, existem vários *standards* nesta área pelo que as organizações devem escolher o que melhor se adequa aos seus projetos. Esta escolha das melhores práticas é influenciada

pelas características dos responsáveis da tomada de decisão, como a sua experiência de trabalho, a idade e a posição que atualmente ocupam na organização (Ferreira et al., 2013).

No entanto, assim como referido no subcapítulo anterior, o âmbito desta dissertação vai para além dos projetos e da gestão dos projetos. Isto porque o nosso foco é a gestão de projetos organizacional que, segundo o Project Management Institute (2013a), é uma abordagem da execução da estratégia que recorre a projetos, programas e portefólios, assim como a práticas organizacionais, para consistentemente e previsivelmente concretizar a estratégia organizacional, gerando melhor desempenho, melhores resultados e, conseqüentemente, vantagem competitiva. Portanto, é importante também definir gestão de programa e gestão de portefólio.

A gestão de programa é definida como a aplicação de conhecimento, habilidades, ferramentas e técnicas a um programa, a fim de atender os requisitos e obter benefícios e controlo não disponíveis através da gestão de projetos individuais (Project Management Institute, 2013a).

Quando uma organização se depara com a necessidade de gerir um conjunto de projetos e/ou de programas, em simultâneo, surge a necessidade de aplicar a gestão de portefólios. Segundo o Project Management Institute (2013a), a gestão de portefólios concentra-se na gestão centralizada de um ou mais portefólios para alcançar objetivos estratégicos. As atividades da gestão de portefólio focam-se em garantir que os projetos e programas são revistos para priorizar alocação de recursos, e em garantir que a gestão do portefólio é consistente e está alinhada com as estratégias organizacionais.

2.1.3 Tecnologias e Sistemas de Informação

O tema desta dissertação é focado em empresas de Tecnologias e Sistemas de Informação (TSI). Como tal, o entendimento dos conceitos de tecnologias de informação (TI), de sistema de informação (SI) e da sua relação é importante para ter uma perceção mais clara de empresas de TSI.

Não existe uma definição consensual para Sistema de Informação e tal falta de concordância deve-se, sobretudo, ao uso indistinto do termo para designar coisas diferentes (Carvalho, 2000), (Amaral, 1994). Mesmo em livros introdutórios às Tecnologias e Sistemas de Informação, frequentemente os autores escolhem a definição de sistemas de informação que lhes é mais útil, descrevem os seus componentes mas não dão nenhuma definição explícita para o termo (Carvalho, 2000). Assim sendo, nesta secção serão abordadas algumas definições que foram apresentadas ao longo dos anos e serão abordados alguns artigos relativos a esta temática.

Buckingham, Hirschheim, Land e Tully (1987) definem sistema de informação como sendo um sistema que junta, guarda, processa e entrega a informação relevante para uma organização, de tal

modo que a informação seja acessível e útil para aqueles que desejem utilizá-la. Acrescenta ainda que um sistema de informação é um sistema de atividade humana que não tem obrigatoriamente de envolver o uso de computadores.

Posteriormente, Alter (1992) definiu sistema de informação como sendo o conjunto de pessoas, procedimentos, informação e tecnologias de informação organizados para alcançar os objetivos da organização.

Outra definição é dada por Amaral (1994) que considera que um sistema de informação é uma abstração que resulta de uma observação a uma organização numa perspetiva de informação, assim como dos suportes humanos, organizacionais e tecnológicos envolvidos na recolha, armazenamento, processamento e entrega da informação.

No relatório FRISCO (Falkenberg et al., 1998), sistema de informação é definido como sendo um subsistema de um sistema organizacional, que abrange a conceção de como os aspetos da comunicação e da informação são compostos e a forma como estes são operados. Um sistema de informação descreve, portanto, as ações orientadas à comunicação, as ações de fornecimento de informações e os arranjos dentro da organização

De forma mais breve, Soares (1998) diz que o sistema de informação de uma organização é o conjunto das relações informacionais que existem nessa organização.

Um artigo publicado mais tarde por Carvalho (2000) aborda a falta de concordância relativamente ao que é um sistema de informação. O objetivo deste artigo não passa por dar outra definição de sistema de informação mas sim, por fazer uma revisão literária e comparar os vários significados que o termo assume. Desta revisão resultaram quatro objetos que, segundo o autor, podem ser considerados sistemas de informação, sendo cada um deles utilizado para diferentes atividades profissionais.

O primeiro (IS1) é o sistema de informação a nível organizacional. São as organizações cujo propósito é providenciar informação aos seus clientes.

O segundo (IS2), e olhando já para dentro das organizações, é o subsistema que suporta a comunicação entre o subsistema de gestão e o subsistema operacional. Este subsistema existe dentro de qualquer sistema capaz de se autogovernar.

O terceiro (IS3) é sobretudo utilizado para automatizar atividades que mexem com informação.

Por fim, o quarto (IS4) é um subconjunto de uma organização que engloba todas as atividades que manipulam informação. De realçar o facto de se a organização lidar apenas com a informação, então este tipo de sistema de informação inclui todas as atividades organizacionais. Neste contexto, o IS4 é muito semelhante ao IS1, sendo apenas diferente deste porque o IS4 não faz referência ao seu propósito.

O autor do artigo, depois de caracterizar estes quatro tipos de sistemas de informação, analisou as definições encontradas na sua revisão de literatura. Como resultado, verificou que as definições raramente eram utilizadas para precisar os sistemas de informação de acordo com os quatro sistemas de informação propostos. Qualquer definição encontrada enquadrava-se com vários dos objetos propostos.

Mais recentemente, Soares e Amaral (2014) analisaram várias definições para sistemas de informação e salientaram que existem várias perspectivas quanto à definição de SI, sendo uma perspectiva comum considerar um sistema de informação como um sistema sociotécnico. Aliás, tal perspectiva pode ser verificada nas definições de Amaral e Alter, já abordadas acima nesta secção. Desta perspectiva resultam três ideias nucleares acerca do que é um sistema de informação: (1) é uma abstração da organização, ou seja, é algo inerente à organização. Se existe organização, existe sistema de informação; (2) é um sistema de atividades sociais e humanas e (3) é, cada vez mais, um sistema suportado tecnologicamente, ou seja, cada vez mais os sistemas de informação são suportados por tecnologias de informação (TI).

Quanto às tecnologias de informação, estas podem ser definidas como o conjunto de equipamentos e suportes lógicos que suportam as atividades de aquisição, transmissão, armazenamento, recuperação e disseminação de dados de uma organização (Alter, 1992). Também podem ser definidas como o conjunto de recursos de *hardware* e *software* utilizado para automatizar os serviços de informação de uma organização (Ward et al., 1990). Ambas as definições vai ao encontro das ideias abordadas no parágrafo anterior, mais especificamente, de que as TI assumem um papel preponderante no suporte dos sistemas de informação.

2.2 Estruturas organizacionais

Uma estrutura organizacional é definida como sendo o sistema formal das relações de tarefas e autoridade, que controlam tanto como as pessoas coordenam as suas ações, como a forma como usam os recursos disponíveis para atingir as metas da organização (Jones, 2013).

O seu estudo é fundamental nesta dissertação pois a estrutura organizacional influencia a posição que um PMO assume na hierarquia da organização e, conseqüentemente define a proporção de projetos sob a sua alçada (Jalal & Koosha, 2015).

A estrutura organizacional é considerada um fator ambiental da empresa que pode afetar a disponibilidade de recursos e influenciar a forma como os projetos são conduzidos (Project Management Institute, 2013a, 2017).

Evolui à medida que a organização cresce e se diferencia, respondendo às contingências envolvendo fatores como o ambiente, a tecnologia e os recursos humanos (Jones, 2013). Podem variar desde organizações estritamente funcionais a organizações orientadas a projetos, sendo que no espectro entre esses dois polos surgem as organizações matriciais (Project Management Institute, 2013a).

Nas organizações funcionais os colaboradores são agrupados por especialidade, sendo que cada especialidade pode ser subdividida em unidades funcionais cada vez mais específicas. Cada um destes departamentos funcionais faz o seu trabalho independentemente dos restantes (Project Management Institute, 2013a). Conforme ilustrado na Figura 3, a gestão do projeto é feita pelos gestores desses departamentos funcionais. Nestes casos a autoridade de um gestor de projeto, quando existe, é muito limitada.

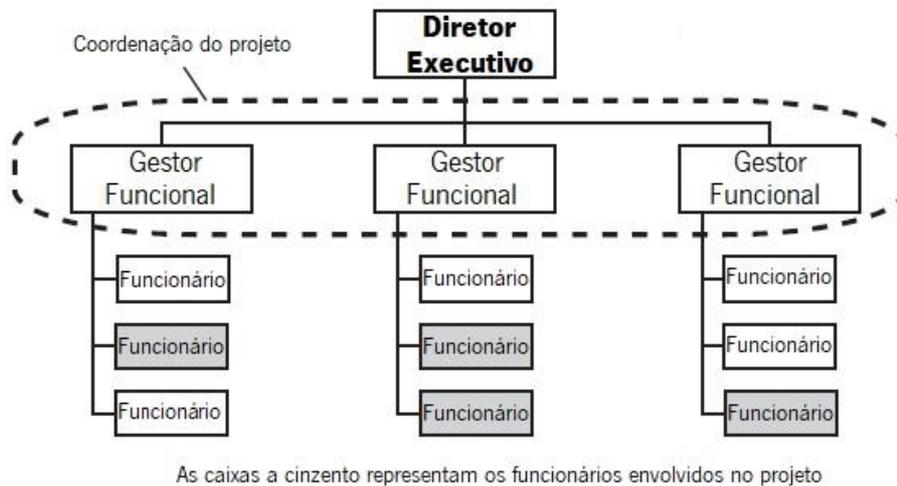


Figura 3- Organização funcional (retirado de (Project Management Institute, 2013a))

No outro extremo do espectro estão as organizações orientadas a projetos. Neste caso, maior parte dos recursos da organização estão envolvidos no trabalho de projeto, tendo o gestor de projeto um grande nível de autoridade, sendo também responsável pelo orçamento (Project Management Institute, 2013a). Enquanto a gestão de projetos em estruturas organizacionais funcionais é feita pelos gestores funcionais, neste tipo de organização é feita pelo gestor do projeto, conforme ilustrado na Figura 4.

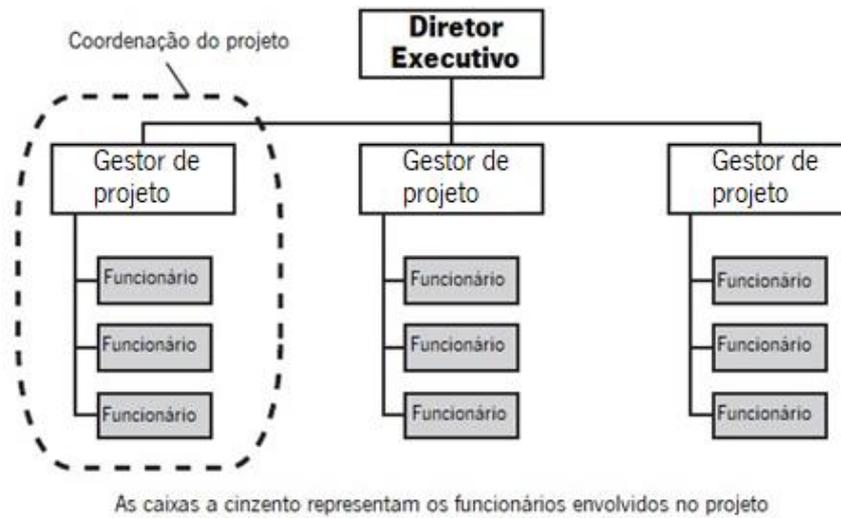


Figura 4- Organização orientada a projetos (adaptado de (Project Management Institute, 2013a))

No meio do espectro encontram-se as organizações matriciais. São uma mistura das características das estruturas dos polos do espectro, podendo ser caracterizadas como organizações matriciais fracas, equilibradas ou fortes, dependendo do nível relativo de poder entre gestores funcionais e gestores de projeto (Project Management Institute, 2013a). Organizações matriciais fracas têm características mais parecidas às organizações funcionais e as organizações matriciais fortes têm características mais parecidas com as organizações orientadas a projetos.

Para entender o meio-termo desse espectro importa compreender como se caracterizam as organizações matriciais equilibradas (ver Figura 5). Neste tipo de estrutura organizacional é reconhecida a necessidade de um gestor de projeto, no entanto, o seu nível de autoridade sobre o projeto é limitado partilhando responsabilidades com os gestores funcionais (Project Management Institute, 2013a).

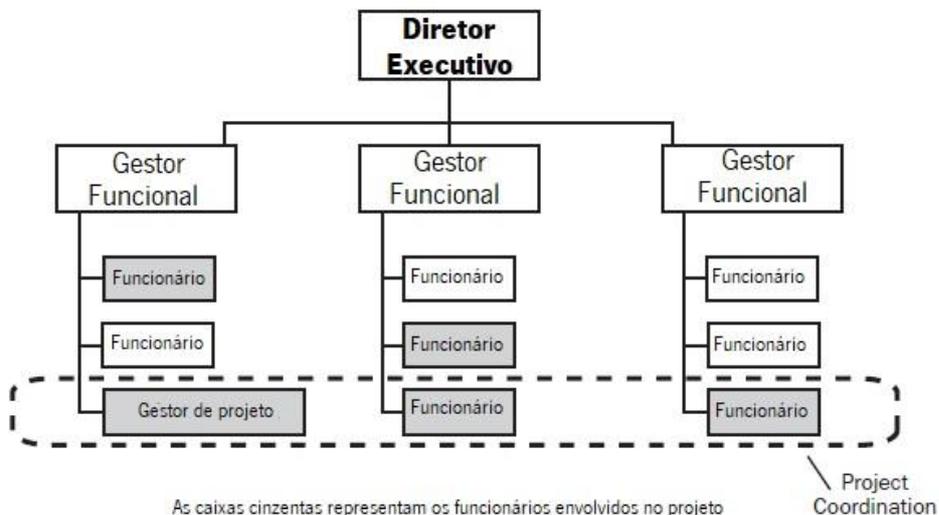


Figura 5- Organização matricial equilibrada (adaptado de (Project Management Institute, 2013a))

Todas estas estruturas organizacionais não são incompatíveis entre si, tanto que muitas organizações usam-nas em diferentes níveis, simultaneamente, no que se designa como uma organização composta (Project Management Institute, 2013a).

Visto que o resultado desta investigação é um modelo para um gabinete de suporte a projetos ágeis, importa entender de que forma a estrutura da organização influencia a sua integração.

Segundo um estudo da McKinsey, de 2011, 57% das organizações inquiridas estavam a iniciar um processo de reformulação organizacional a cada dois anos e a duração média de cada uma dessas reformulações durava aproximadamente 18 meses, o que significa que quando estavam a terminar uma reformulação já tinham de planear e iniciar outra para responder às condições do mercado (Aghina et al., 2018).

Este aumento da complexidade e volatilidade dos mercados promoveu a adoção da mentalidade ágil ao nível organizacional, exigindo mudanças na cultura da organização, estilo de gestão, estrutura organizacional e modelo operativo como um todo (Aghina et al., 2018).

Embora não seja consensual qual a estrutura organizacional mais adequada para o contexto ágil, para permitir que as organizações se possam adaptar às mudanças do mercado, as suas hierarquias devem ser o mais planas possíveis, isto é, deixar de ser rigidamente hierarquizadas e organizadas em silos funcionais, para passarem a hierarquias mais planas, focadas em redes de equipas empoderadas (Aghina et al., 2018; Cohn, 2010; Nerur et al., 2005).

2.3 Project Management Office

2.3.1 Definição e contextualização

Project Management Office (PMO) pode ser definido como uma estrutura de gestão que padroniza os processos de governação relacionados com os projetos debaixo do seu domínio e facilita a partilha de recursos, metodologias, ferramentas e técnicas entre eles (Project Management Institute, 2013a). A responsabilidade desta entidade pode variar desde funções de suporte à gestão do projeto até funções em que realmente gere diretamente um ou mais projetos (Project Management Institute, 2013a). Este espectro de responsabilidade é retratado mais á frente no subcapítulo 2.3.2.

A importância destas estruturas deve-se ao aumento do número de projetos que são realizados, simultaneamente, na mesma organização, o que conseqüentemente aumenta a complexidade de os gerir e coordenar. Este aumento do número de projetos é ditado por pressões do mercado, como a

necessidade de manter um portfólio de produtos atualizado e a necessidade de reduzir o tempo que o produto leva do desenvolvimento até ser colocado no mercado (Aubry et al., 2007). O aumento consequente da complexidade levou à criação dos PMOs.

No entanto, o que se verifica é que a designação de *Project Management Office* é utilizada para se referir a diferentes entidades. Existem, portanto, várias definições que diferem em variados aspetos, nomeadamente quanto ao nome, à estrutura do PMO, aos papéis assumidos e quanto ao seu valor percebido (Hobbs & Aubry, 2007).

O objetivo nuclear de qualquer PMO é suportar e melhorar a gestão dos projetos, através da adoção de metodologias para conseguir níveis altos de eficiência e eficácia (Monteiro et al., 2016). Porém, não há um modelo universal que permita que este objetivo seja atingido em todas as organizações. Cada organização deve procurar o modelo que melhor satisfaça as suas necessidades.

2.3.2 Tipologias e modelos

Vários modelos de PMO são propostos pelos mesmos autores e em muitos casos a posição hierárquica de cada PMO na organização estabelece o seu nível de autoridade, aceitação, adoção e autonomia. Estas descrições de PMOs são usualmente resumidas e agrupadas em tipologias (Monteiro et al., 2016).

Os autores apresentam vários modelos de PMOs que se diferenciam sobretudo quanto à sua estrutura, funções, papéis assumidos e no valor percebido. De notar que o mesmo autor pode propor mais do que um modelo de PMO, tendo em conta o contexto em que será utilizado. Essas descrições são muitas vezes agrupadas em tipologias (Monteiro et al., 2016).

Segundo o PMBOK (Project Management Institute, 2013a) existem três tipos diferentes de PMO. Estes diferenciam-se entre si quanto aos graus de controlo e influência que têm nos projetos pelos quais estão encarregues. Segundo o PMI, os PMOs podem ser caracterizados, da seguinte forma:

- **Supportive:** tipo de PMO com o nível de controlo mais baixo. Funciona como consultor dos projetos, providencia *templates*, boas práticas, treino, acesso a informação e é responsável por transmitir lições aprendidas de outros projetos.
- **Controlling:** tipo de PMO com um nível moderado de controlo. É responsável por fornecer apoio e exigir conformidade através de diversas formas.
- **Directive:** o tipo de PMO com o nível de controlo mais elevado. Assume o controlo dos projetos, gerindo-os diretamente.

Para além disso, o seu papel também difere relativamente ao nível de gestão em que opera, apresentando as seguintes funções para cada nível (Desouza & Evaristo, 2006):

- **Nível operacional:** fornece suporte básico centralizado para os projetos individuais e dissemina boas práticas e princípios de gestão de projetos;
- **Nível tático:** fornece coordenação entre vários projetos e faz a gestão das dependências entre projetos. Pode envolver a integração de recursos entre projetos;
- **Nível estratégico:** Para além dos serviços dos outros dois níveis, também é responsável por priorizar projetos tendo em conta a estratégia a empresa, assim como a seleção dos projetos mais viáveis.

Para entender mais aprofundadamente a variedade de modelos de PMO existentes, foram estudadas algumas tipologias de autores diferentes. A literatura relativa a este tema é bastante extensa e, como tal, optou-se por analisar apenas cinco tipologias, todas de anos e autores diferentes.

Uma das tipologias é proposta por Desouza e Evaristo (2006) que propõem quatro modelos de PMOs. O *Supporter* que serve uma função principalmente administrativa providenciando o estado do projeto, identificando riscos e potenciais problemas, e mantendo os arquivos dos projetos. Informa sobre os projetos mas não os influencia. O *Information Manager* tem como função seguir e relatar o progresso dos projetos com o foco de servir como fonte de informação de projetos e de atualizações consolidadas dos seus estados. Outro modelo proposto é o *Knowledge Manager* que funciona como um repositório de boas práticas, fornecendo expertise sobre projetos, aconselhamento e treino. No entanto não tem nenhuma responsabilidade administrativa. Por fim, o *Coach* enfatiza o melhoramento, a excelência e a responsabilidade para reforçar a gestão de projetos da organização.

Outra tipologia foi proposta por Hill (2008), sendo ela constituída por cinco modelos de PMOs. O *Strategic Office* fornece a capacidade de assegurar profissionalismo e excelência através da aplicação de princípios amplamente aceites e práticas de gestão adequadas ao esforço de cada projeto. O modelo seguinte é o *Basic PMO* cujo nível lida com múltiplas supervisões de projetos e permite o controlo de vários projetos tendo em conta o desempenho dos gestores de projeto. O *Standard PMO* permite que a supervisão e o controlo possam ser centralizados. Também suporta o ambiente de gestão do projeto. Outro PMO proposto nesta tipologia é o *Advanced PMO* cujo principal objetivo é integrar os objetivos e interesses organizacionais na gestão dos projetos. Por fim, propõe o *Centre of Excellence* que é focado nos interesses estratégicos do negócio ao longo da organização, tendo acesso direto ao diretor executivo e a capacidade de influenciar as operações da gestão de projetos da empresa.

Crawford (2010) propõe três modelos de PMOs diferentes. Um desses modelos é o *Project Control Office* que lida com projetos grandes e complexos e é o seu foco é normalmente restrito a um só projeto de grandes dimensões. Podem ser necessários vários calendários o que poderá criar a necessidade de os juntar num calendário geral do programa. O *Business Unit PMO* gere múltiplos projetos de vários tamanhos. Também providencia uma eficiência maior na gestão de recursos e identificação de prioridades entre projetos. Por último propõe o *Strategic PMO* que se posiciona a nível corporativo e que apoia a gestão de topo no processo de priorização dos projetos de acordo com os objetivos estratégicos da empresa.

Outra das tipologias é proposta pelo *Project Management Institute* que na apresenta cinco modelos de PMOs diferentes (Project Management Institute, 2013b). Um mais relacionado com o projeto em si, cuja principal função é providenciar apoio e suporte a projetos ou programas, que se designa por *Project Specific*. Um segundo que providencia serviços relacionados com os projetos para apoiar uma unidade de negócio, que se designa por *Business Unit PMO*. Outro modelo proposto pelo PMI é o *Project Support Office* que visa suportar administrativamente a realização e conclusão de um projeto. A um nível mais abrangente surge o modelo *Enterprise PMO* que é responsável pelo alinhamento entre os projetos/programas com a estratégia da organização. Por fim, existe o modelo que se designa por Centre of Excellence e a sua principal função é apoiar o trabalho realizado durante o projeto, através do estabelecimentos de padrões e metodologias a seguir.

Mais recentemente, foi proposta uma tipologia por Hubbard e Bolles (2015) com sete modelos de PMO. O *Enterprise PMO* é responsável pela gestão do negócio em toda a empresa, existindo apenas um por empresa. Este PMO reporta diretamente ao diretor executivo da empresa e é responsável pelo planeamento geral estratégico e tático. Para além disso, também seleciona e prioriza projetos e supervisiona portefólios e programas. O *Division PMO* é de natureza tática e providencia gestão de negócios do projeto para uma divisão inteira e reporta diretamente a um gestor de divisão ou ao *Enterprise PMO*. O seu foco é gerir portefólios e programas de projetos, e supervisionar *Project PMOs* e, por vezes, *Business PMOs*. O *Business Unit PMO* tem um foco mais operacional e o seu foco é a gestão de programas e portefólios de projetos. Reportam diretamente ao *Division PMO*. O *Project PMO* e o *Project Office* também tem um foco operacional mas enquanto o *Project PMO* tem como foco o suporte de um projeto grande e complexo e o *Project Office* tem como foco fornecer suporte direto a um único projeto simples. Ambos são responsáveis por todas as fases do projeto, nomeadamente, iniciação, planeamento, execução, monitoramento, controlo e fecho. O *Project Support Organization* cujo foco é providenciar suporte administrativo a um ou mais projetos simples em todas as suas fases. Também

reporta o progresso e estado do projeto e este tipo de PMO normalmente é temporário. Por fim o *Project Management Center of Excellence* estabelece e implementa padrões de gestão de negócios de projetos, as metodologias, as práticas, a educação, o treino e a competência de gestão de projeto em toda a empresa. Devido à similaridade entre o *Project Office* e o *Project PMO* e entre o *Division PMO* e o *Business Unit PMO* estes são agrupados em *Project Office/PMO* e *Division/Business Unit PMO*, respetivamente, por Monteiro *et al* (2016) perfazendo assim um total de apenas cinco modelos.

Da análise das tipologias pode-se verificar que, de facto, os modelos de PMO apresentados por cada uma se diferenciam nas suas funções básicas e no seu nível de controlo e de influência, conforme sugerido pelo Project Management Institute (2013a). Os modelos são tipificados de acordo com as suas funções mais de suporte, de controlo ou de direção.

Um estudo mais intensivo sobre as tipologias foi feito por Monteiro et al.(2016), onde analisaram doze tipologias diferentes de modo a encontrar uma melhor caracterização do termo PMO. Depois de identificados todos os modelos de PMO de cada tipologia analisada, foram comparados os respetivos nomes. Neste estudo concluiu-se que os modelos mais comuns são o *Enterprise PMO*, o *Project Management Centre of Excellence*, o *Project Office* e o *Project Support Office*, como é possível verificar na Figura 6.

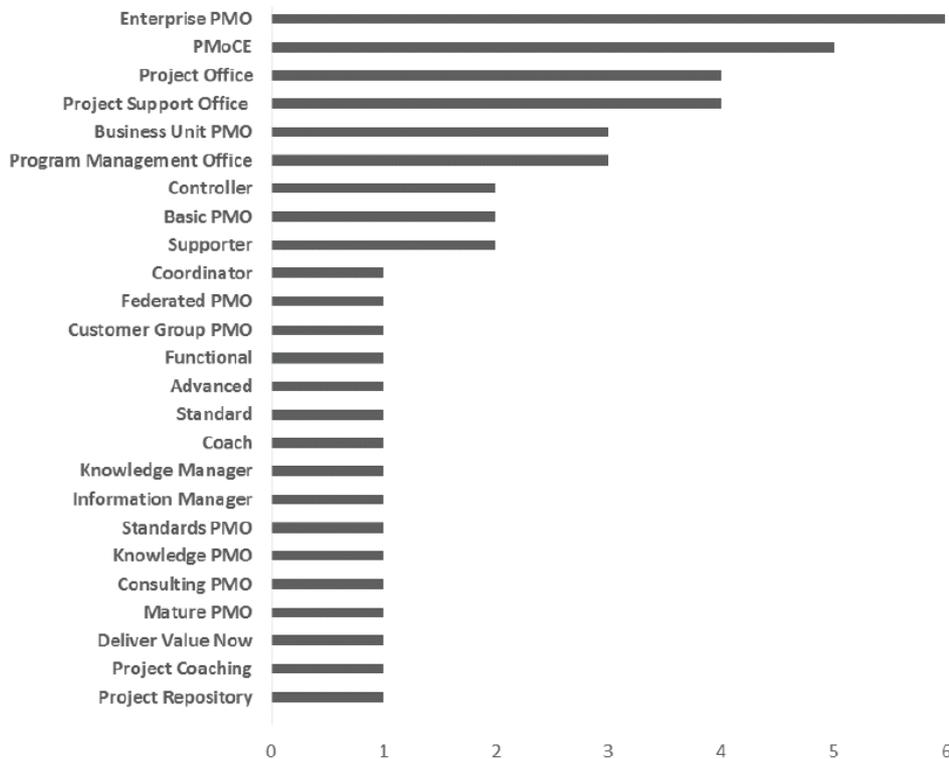


Figura 6- Frequência dos modelos de PMO nas tipologias (retirado de (Monteiro et al., 2016))

Esta variedade pode dificultar o processo de implementação de um PMO pois em muitas empresas os responsáveis são incentivados a implementar um sem qualquer conhecimento sobre o que isso pode implicar, nem sobre qual o PMO que mais se adequa à sua situação (Aubry et al., 2010).

O tipo de estrutura organizacional tem um papel fundamental nesta escolha pois enquanto para uma organização centralizada e rigidamente hierarquizada seria mais apropriado um PMO orientado aos processos que execute os projetos diretamente, numa organização descentralizada em que o direito de decisão está disponível a indivíduos de todos os níveis, seria mais apropriado um PMO que resultasse da colaboração voluntária entre os gestores de projetos (Desouza & Evaristo, 2006).

2.3.3 Processo de implementação

Quando se toma a decisão de estabelecer um PMO é necessário entender como este irá encaixar na cultura da empresa, perceber quais as motivações que despoletaram a sua utilização e qual a abordagem de implementação mais adequada (Desouza & Evaristo, 2006).

Segundo Singh (2009), por vezes, para implementar um PMO é necessária a mudança de mentalidade e uma mudança para uma organização mais orientada a projetos. O mesmo autor identificou vários desafios, sendo os seguintes os três mais proeminentes: (1) cultura empresarial rígida e falha na gestão da resistência à mudança organizacional, (2) falta de experiência dos gestores de projetos e falta de liderança do PMO e (3) falta de uma estratégia adequada para gerir a mudança.

Estes desafios são corroborados pelos fatores críticos de sucesso que Desouza e Evaristo(2006) identificam como fundamentais para implementar um PMO com sucesso, sendo eles (1) construir uma fundação forte, (2) estabelecer as motivações da implementação, (3) escolher o projeto certo para o gestor certo, (4) identificar claramente as linhas de comunicação, (5) estabelecer métricas para avaliar o PMO e (6) substanciar a sua credibilidade em documentos.

Em consonância com o âmbito desta dissertação e tendo em conta as dificuldades dos processos tradicionais, Alsadeq (2011) sugere um processo de implementação ágil do PMO. Os principais fatores motivadores para assumir esta abordagem são, (1) os fatores culturais, (2) falta de informação no início da implementação, (3) o PMO ser um conceito novo para o cliente e, (4) o PMO ser uma ferramenta de gestão da mudança. Nesta implementação o PMO é desenvolvido com o auxílio de uma organização externa de consultoria.

Esta abordagem parte do princípio que nem toda a gente vai concordar e aceitar todas as mudanças inerentes à implementação do PMO e portanto é importante ir conquistando vitórias rápidas para demonstrar o seu valor. Alsadeq (2011) sugere um processo incremental que começa com um

protótipo da solução final que será refinado com a colaboração do cliente assim este veja a concretização dos benefícios preconizados. O que sugere é que as pessoas sejam treinadas desde o primeiro dia, para compreenderem a fundo as potencialidades do PMO e que se apliquem mudanças ao longo do percurso, iterativamente até chegar ao estado desejado. Isto requer uma colaboração íntima com o cliente, envolvendo-o na seleção dos passos que acha mais prioritários implementar. Consequentemente, este envolvimento também permite que o cliente aceite com mais facilidade o processo de implementação.

Um PMO bem definido, alinhado com a cultura e objetivos da organização pode potencializar a capacidade deste alcançar um maior nível de sucesso (Desouza & Evaristo, 2006).

2.4 Gestão tradicional e ágil de projetos

2.4.1 O modelo em cascata

Frequentemente designado como a abordagem tradicional de desenvolvimento de *software*, o modelo em cascata foi formalmente descrito pela primeira vez, por Winston W. Royce (1970).

Este modelo caracteriza-se pelo seu ciclo de vida composto por fases sequenciais, conforme ilustrado na Figura 7. Esta linearidade torna a abordagem pouca flexível pois progride unidireccionalmente para as fases seguintes (razão pela qual se utiliza o termo “cascata”), não permitindo efetuar mudanças significativas (como por exemplo, mudança dos requisitos) no decorrer do projeto.

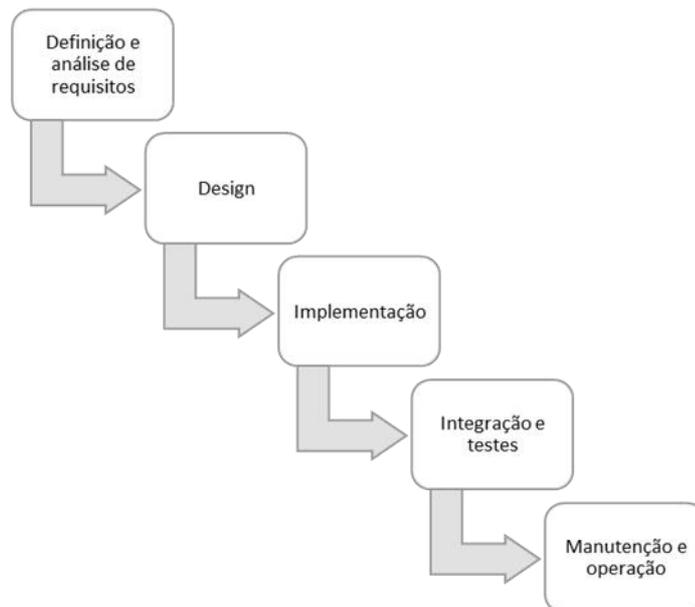


Figura 7- Fases de desenvolvimento de software do modelo de cascata

Para se mudar algo relativo a uma fase anterior é necessário descartar o trabalho feito até então, voltar a essa fase anterior e refazer todas as etapas que lhe sucedem. Como se trata de uma metodologia baseada no plano, todas as fases do projeto são planeadas antes de se iniciar qualquer atividade de desenvolvimento. De cada uma das fases da Figura 7 resulta pelo menos um documento que precisa de aprovação antes de se iniciar a próxima (Sommerville, 2010). Normalmente existem cinco fases distintas, que podem ser brevemente descritas da seguinte forma (Sommerville, 2010):

- **Definição e análise de requisitos:** Esta primeira fase é onde se estabelece os serviços, limitações e objetivos do sistema a ser desenvolvido. Para tal consulta-se os utilizadores do sistema. Depois de recolhidos os serviços, limitações e objetivos são definidos os requisitos com maior detalhe e utilizados como uma especificação do sistema.
- **Conceção:** O processo de conceção do sistema aloca os requisitos para os sistemas de *hardware* ou de *software* através do estabelecimento de uma arquitetura geral do sistema. Nesta fase concebe-se o *software* com base nos requisitos definidos na fase anterior.
- **Implementação:** O desenvolvimento do *software* é realizado originando um conjunto de programas. Nesta fase também se dá início aos testes, nomeadamente aos testes das unidades realizadas, em que se verifica se cumprem com a sua especificação.
- **Integração e testes:** Nesta fase dá-se início à integração dos vários programas realizados na etapa anterior. De seguida são testados já como um sistema completo para assegurar que os requisitos do *software* foram cumpridos. Depois de concluir estes testes, o *software* é entregue ao cliente.
- **Manutenção e operação:** Nesta fase o sistema é instalado e operado. A sua manutenção consiste sobretudo na correção de defeitos que não foram detetados durante as anteriores fases do ciclo de vida.

De realçar que o modelo de Royce foi baseado na experiência profissional do autor, que à data desenvolvia pacotes de *software* para missões aeroespaciais. O elevado nível de disciplina característico desta área é claramente refletido na estruturação do modelo inicial que propôs. No entanto, e citando o autor relativamente a este modelo, ele “acreditava neste conceito mas a sua implementação é arriscada e convidativa a falhas” (Royce, 1970).

Outros processos de desenvolvimento de *software* tradicionais surgiram inspirados no modelo em cascata, como é o caso do modelo em V e do modelo Incremental e Iterativo. Outras abordagens tradicionais, como o modelo espiral formalizado por Boehm (1988), foram propostas como alternativas ao modelo em cascata (Fernandes & Machado, 2016).

Na década de 1990, vários profissionais de desenvolvimento de *software* começaram a aperceber-se que durante o decorrer dos projetos, o ambiente tecnológico e do negócio ia mudando e, conseqüentemente, os requisitos e planos do projeto rapidamente se tornavam desatualizados. Como resposta a esta problemática começaram a surgir metodologias que lidavam, em vez de ignorar, com este elevado nível de mudança que os projetos de desenvolvimento de *software* enfrentavam (Williams & Cockburn, 2003).

2.4.2 O manifesto ágil

Em fevereiro de 2001, numa estância de esqui no estado norte-americano Utah, um grupo de líderes da área do desenvolvimento de *software* reuniu-se para debater as semelhanças que as suas metodologias tinham em comum, as até então chamadas “metodologias leves”.

Desta reunião resultou um artefacto, assinado por todos os participantes, denominado Agile Manifesto (Beck et al., 2001). Este artefacto é composto por quatro valores nucleares e por doze princípios que suportam esses mesmos valores, para a atividade de desenvolvimento de *software*. Os valores são enumerados no manifesto da seguinte forma (Beck et al., 2001):

- Indivíduos e interações mais do que processos e ferramentas;
- *Software* funcional mais do que documentação abrangente;
- Colaboração com o cliente mais do que negociação contratual;
- Responder à mudança mais do que seguir um plano.

Em todos os valores enunciados é reconhecido o valor aos itens mencionados no fim das frases mas prioriza-se os itens do início.

Estes quatro valores são a base dos métodos ágeis e para que se possa verificar se estão a ser seguidos, foram enunciados doze princípios. Estes doze princípios são os seguintes (Beck et al., 2001):

1. Garantir a satisfação do cliente através de entregas de *software* com valor, feitas o mais cedo possível e de forma contínua;
2. Acomodar requisitos em mudança durante o processo de desenvolvimento;
3. Entrega frequente de *software* que funciona;
4. Colaboração entre as partes interessadas e os desenvolvedores, durante o projeto;
5. Suporte, confiança e motivação das pessoas envolvidas;
6. Promover interações cara-a-cara;
7. *Software* funcional é a principal medida de progresso;

8. Processos ágeis para suportar um ritmo de desenvolvimento consistente;
9. Prestar atenção aos detalhes técnicos e de *design* aumenta a agilidade do projeto;
10. Simplicidade;
11. As melhores arquiteturas, requisitos e *designs* emergem de equipas que se auto-organizam;
12. Reflexões regulares em como se pode ser mais eficaz.

A conceção destes princípios não tinha como objetivo definir o que é ou não é uma metodologia ágil. Visavam sim, servir como diretrizes para que o *software* a ser desenvolvido fosse entregue com qualidade, de uma forma ágil.

A divulgação do manifesto ágil chamou a atenção dos desenvolvedores de *software*, causando mudanças consideráveis na área (Dingsøyr et al., 2012). Estes valores e princípios são a base das abordagens ágeis que surgiram desde então e que são analisadas detalhadamente no subcapítulo 2.5. Mas primeiro importa entender como é que este paradigma se diferencia do seu antecessor.

2.4.3 Tradicional vs ágil

Assim como em áreas mais clássicas da gestão de projetos (e.g. engenharia civil), também a área de desenvolvimento de *software* tem demonstrado uma alta taxa de insucesso (Varajão et al., 2014).

No *Chaos Report* de 2015 do *Standish Group* (Hastie & Wojewoda, 2015) foi feita uma comparação do sucesso entre os projetos de desenvolvimento ágil e de desenvolvimento tradicional (representados pelos projetos em cascata). Nessa comparação foram analisados mais de 10 mil projetos de desenvolvimento de *software*, entre os anos de 2011 e 2015, inclusive.

Independentemente da dimensão dos projetos analisados, os resultados desta comparação indicaram que 39% dos projetos ágeis foram concluídos com sucesso enquanto apenas 11% dos projetos em cascata conseguiram o mesmo desfecho. No mesmo estudo são apresentados os principais fatores que conduziram esses projetos ao sucesso, sendo que os três mais preponderantes foram o suporte executivo, a maturidade emocional e o envolvimento do cliente.

O desenvolvimento de *software* é uma atividade complexa, caracterizada por tarefas e requisitos que exibem elevados graus de variabilidade (Nerur et al., 2005). Para entender a diferença de resultados abordada no parágrafo anterior é importante perceber como cada uma das abordagens se caracteriza. Esta comparação está ilustrada na Tabela 2.

Tabela 2- Desenvolvimento de software: Tradicional vs Ágil (adaptado de (Nerur et al., 2005))

Características	Tradicional	Ágil
Premissas fundamentais	Os sistemas são totalmente especificáveis, previsíveis e podem ser construídos através de planeamento meticuloso e extensivo.	Qualidade elevada; <i>software</i> adaptativo desenvolvido por equipas pequenas que usam os princípios de melhoramento contínuo do <i>design</i> e testes baseados em <i>feedback</i> , e mudança rápida.
Controlo	Centrado no processo	Centrado nas pessoas
Estilo de gestão	Comando e controlo	Liderança e colaboração
Gestão de Conhecimento	Explícito	Tácito
Atribuição de papéis	Individual – favorece especialização	Equipas auto-organizadas - encoraja permutabilidade de papéis
Comunicação	Formal	Informal
Papel do cliente	Importante	Crítico
Ciclo do projeto	Guiado por tarefas ou atividades	Guiado por funcionalidades do produto
Modelo de desenvolvimento	Modelo de ciclo de vida (cascata, espiral e outras variações)	Modelo de entrega evolucionário
Estrutura organizacional desejada	Mecanicista (burocrática com elevada formalização)	Orgânica (Ação social cooperativa, encorajadora, flexível e participativa)
Tecnologia	Sem restrições	Favorece tecnologia orientada a objetos

Como se pode constatar, as duas abordagens tem naturezas completamente diferentes. Logo à partida porque assentam em fundamentos opostos pois enquanto nas abordagens tradicionais se assume que os sistemas são totalmente especificáveis e, portanto, podem ser completamente planeados e divididos em tarefas *a priori*, nas abordagens ágeis assume-se que a mudança faz parte de qualquer projeto e, portanto, o projeto deve-se ir adaptando a essas mudanças e ajustando a prioridade da próxima

funcionalidade a desenvolver. Esta diferença provoca uma grande disparidade entre estes paradigmas do desenvolvimento de *software*.

Resumidamente, as abordagens tradicionais caracterizam-se pelo seu ciclo de vida sequencial com papéis hierárquicos claramente definidos, pela transmissão de conhecimento formal através de muita documentação, por interagir com o cliente principalmente no início do projeto e por se adaptar sobretudo a estruturas organizacionais funcionais. Já as abordagens ágeis caracterizam-se pelo seu modelo de entrega incremental, pela promoção do empoderamento e autogestão de equipas multifuncionais, pela transmissão de conhecimento sobretudo através de comunicação informal (i.e. conhecimento tácito), pela interação constante com o cliente durante o projeto e por se adaptar sobretudo a estruturas organizacionais “orgânicas” que promovam a colaboração entre elementos de equipa, cliente e gestão superior.

Contudo, tanto a abordagem tradicional como a ágil têm as suas vantagens e fraquezas em determinadas situações. O desafio de escolher qual é a mais apropriada consiste em encontrar quais são as situações ideais em que cada uma prospera. Estas situações ideais para a aplicação das abordagens ágeis ou tradicionais englobam um conjunto de condições sob as quais cada uma das abordagens é mais provável obter sucesso (Boehm & Turner, 2003).

A Tabela 3 descreve essas mesmas condições e permite comparar as situações ideais para cada abordagem. Constata-se que estas duas vertentes do desenvolvimento de *software* prosperam em ambientes distintos. Diferem quanto ao ambiente em que são aplicáveis, aos propósitos e objetivos que visam resolver, à forma como transmitem conhecimento, ao tipo de planeamento que fazem, ao modo como lidam com requisitos e quanto às visões díspares que apresentam em relação às interações entre *stakeholders*.

Tabela 3- Situações ideais das abordagens ágeis e tradicionais (adaptado de (Boehm & Turner, 2003))

Características do projeto		Situação ágil ideal	Situação tradicional ideal
Aplicação	Objetivos primários	Valor rápido e resposta à mudança	Previsibilidade, estabilidade, garantia elevada
	Tamanho	Equipas e projetos menores	Equipas e projetos maiores
	Ambiente	Turbulência, muitas mudanças, focado no projeto	Estável, poucas mudanças, focado no projeto e na organização

Características do projeto		Situação ágil ideal	Situação tradicional ideal
Gestão	Relações com clientes	Clientes dedicados no local, focados nos incrementos prioritizados	Conforme seja necessário, focado nas provisões de contrato
	Planeamento e controlo	Planos internalizados, controlo de qualidade	Planos documentados, controlo quantitativo
	Comunicações	Conhecimento tácito interpessoal	Conhecimento documentado explícito
Técnicas	Requisitos	Histórias informais e casos de teste prioritizados, sofrem mudanças imprevisíveis	Projeto formalizado, capacidade, interface, qualidade, requisitos de evolução previsível
	Desenvolvimento	<i>Design</i> simples, pequenos incrementos, <i>refactoring</i> assumido de baixo custo	<i>Design</i> extensivo, incrementos longos, <i>refactoring</i> assumido de elevado custo
	Teste	Casos de teste executáveis definem os requisitos, experimentação	Procedimentos e planos de teste documentados
Pessoal	Clientes	Dedicados, executadores Crack ³ , localizados com a equipa	Executadores Crack, nem sempre localizados com a equipa
	Desenvolvedores	Pelo menos 30% de especialistas de nível 2 e 3; nenhum de nível 1B ou -1; Níveis da escala revista de Cockburn ⁴	50% Especialistas de nível 3 inicialmente e 10 % durante o desenvolvimento; 30% especialistas de nível 1B; nenhum de nível -1; Níveis da escala revista de Cockburn ⁴
	Cultura	Conforto e empoderamento através de graus de liberdade (prospera no caos)	Conforto e empoderamento através de estruturas de políticas e procedimentos (prospera na ordem)

³ Crack é o acrónimo para “Collaborative, Representative, Authorized, Committed and Knowledgeable”

⁴ Esta escala pode ser encontrada no artigo de Boehm e Turner (2003).

Assim sendo, o processo de migração para as metodologias ágeis precisa ser feito com cautela e com uma percepção realista das necessidades e contexto da organização.

2.4.4 Migração para metodologias ágeis

A atratividade que resulta das oportunidades e benefícios associados às abordagens ágeis deve ser considerada atentamente quando se pondera utilizar ou integrar com práticas já existentes (Nerur et al., 2005). O processo de migração para as metodologias ágeis requer tempo de reflexão devido às características singulares que cada organização possui.

Um estudo de Serrador e Pinto (2015), efetuado com base em 1386 projetos, demonstrou que destes apenas cerca de 6% utilizaram totalmente ou quase totalmente metodologias ágeis (percentagem de tarefas que utilizaram práticas ágeis entre os 80-100%). Nesse estudo concluiu-se que quanto mais abrangente é a utilização da abordagem ágil, maior é a probabilidade do projeto ter sucesso.

De realçar que os dois critérios de sucesso utilizados foram: (1) fator de eficiência e (2) fator de sucesso dos *stakeholders*. O fator de eficiência dizia respeito à eficiência do projeto, ou seja, se este cumpriu os objetivos de custos, tempo e de âmbito. O fator de sucesso dos *stakeholders* estava relacionado com a satisfação das expectativas dos *stakeholders* do projeto.

No entanto, embora se verifique um aumento na eficiência, a utilização de metodologias ágeis parece ter também um grande impacto nos fatores de sucesso (Serrador & Pinto, 2015). Isto justifica-se com o próprio princípio ágil que afirma que a satisfação dos clientes é uma prioridade maior do que os processos utilizados durante o projeto.

Nem todas as organizações possuem as mesmas condições de migrar totalmente para os métodos ágeis. Aliás, como foi referido no subcapítulo 2.4.3, tantos os métodos ágeis como os métodos tradicionais têm ambiente favoráveis onde as características de determinados projetos têm mais probabilidade de obter sucesso (Boehm, 2002).

Embora o sucesso demonstrado nos resultados da aplicação de abordagens ágeis seja atrativo, cabe à organização entender qual o seu contexto e se de facto este paradigma lhe pode trazer mais-valias. Organizações com culturas propícias para a inovação podem abraçar mais facilmente as abordagens ágeis do que organizações baseadas em burocracia e formalidades (Serrador & Pinto, 2015).

2.5 Abordagens ágeis

Segundo um estudo que é realizado anualmente pela empresa VersionOne (2018), em 2017 a metodologia ágil mais utilizada foi o Scrum, seguida de abordagens híbridas que utilizam simultaneamente várias metodologias. Como é possível ver na Figura 8, 56% dos respondentes utilizam Scrum nas suas organizações. Nas posições seguintes, embora maior parte da sua utilização seja em abordagens híbridas, surgem o Kanban e o XP entre as metodologias com mais utilização.

Neste capítulo, visto que existem várias abordagens ágeis disponíveis atualmente, optou-se por explorar em detalhe apenas as três mais utilizadas, nomeadamente, o Scrum, o Extreme Programming e o Kanban.

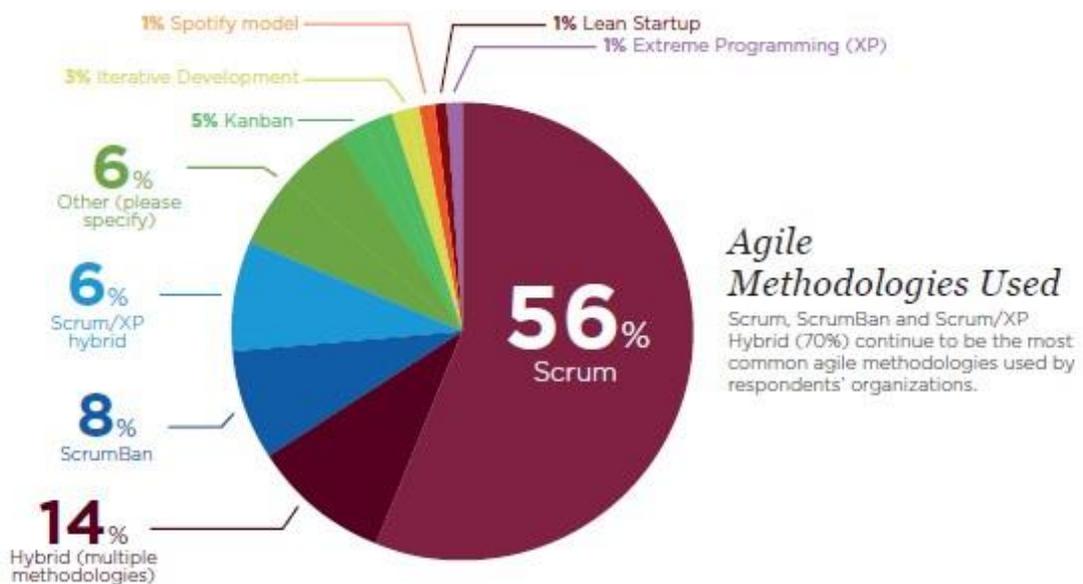


Figura 8- Metodologias ágeis mais usadas em 2017 (retirado de (Versionone.com, 2018))

2.5.1 Scrum

A utilização do termo “Scrum” remonta a 1986, altura em que foi publicado o artigo “The New New Product Development Game”, onde se relata o período de transição dos métodos sequenciais para as abordagens holísticas que à data começavam a surgir nas organizações dos Estados Unidos da América e do Japão, na área de desenvolvimento de produtos (Takeuchi & Nonaka, 1986). Nesse artigo os autores argumentavam que as melhores empresas utilizavam um processo flexível e com equipas autónomas constituídas por membros multidisciplinares (Takeuchi & Nonaka, 1986).

Anos mais tarde, em 1993, Jeff Sutherland começara a trabalhar numa organização e chegara à conclusão que utilizar o modelo em cascata não seria o mais apropriado para a realidade daquela empresa e, acompanhado pela sua equipa, procuram por alternativas até que encontraram o artigo

descrito no parágrafo anterior (Sutherland, 2016). Finalizado o seu primeiro projeto com base nas ideias de Takeuchi e Nonaka, em 1995 formalizou as práticas utilizadas num artigo em parceria com Ken Schwaber, que se intitulava “SCRUM Development Process” (Sutherland, 2016) .

O termo que dá nome à metodologia advém de uma jogada do rugby denominada *scrum*. Essa jogada consiste em fazer com que a bola seja passada entre os jogadores, para a frente e para trás, para que a equipa consiga avançar como uma unidade no campo (Takeuchi & Nonaka, 1986). Esta analogia pretende enfatizar a importância do trabalho de equipa no sucesso de um projeto.

O Scrum é uma *framework* utilizada para desenvolver e manter produtos complexos e, citando Sutherland e Schwaber (2016), é definido como “uma *framework* dentro da qual as pessoas podem resolver problemas adaptativos complexos, ao mesmo tempo que entregam produtiva e criativamente produtos com o maior valor possível”.

Segundo o Sutherland e Schwaber (2016), o Scrum é baseado em teorias empíricas, isto é, o conhecimento vem da experiência e da tomada de decisão baseada no que é sabido. Além disso, o Scrum adota uma abordagem iterativa e incremental para otimizar a previsibilidade e o controlo de risco. Como em qualquer implementação de controlo empírico do processo, o Scrum é sustentado por três pilares fundamentais: transparência, inspeção e adaptação.

A transparência é importante dentro da organização para garantir que todos os aspetos relevantes do processo são visíveis por todos os responsáveis dos resultados. Tais aspetos devem ser definidos por um padrão comum de modo a que quem os observe possa ter um entendimento do que está a ser visto.

Os participantes do projeto Scrum devem inspecionar regularmente os artefactos Scrum e medir o progresso para detetar variâncias indesejáveis. A frequência destas inspeções deve ser ajustada ao ritmo do projeto, de modo a não interferir no trabalho a ser realizado.

Como resultado de tais inspeções pode ser verificado que um determinado aspeto sofreu uma deviação para fora dos limites aceitáveis. Neste caso é necessário fazer um ajustamento para minimizar estas variações.

O Scrum recomenda quatro eventos formais para que a inspeção e consequente adaptação possam ser realizadas, nomeadamente, a reunião do planeamento do *Sprint*, a reunião diária Scrum, a reunião de revisão do *Sprint* e a reunião de retrospectiva do *Sprint*.

Comprometimento, coragem, foco, abertura e respeito são os cinco valores basilares de todos os projetos Scrum (descritos na Tabela 4) e todos os membros da Equipa Scrum devem incorporar estes valores para que os três pilares do Scrum sejam respeitados e consequentemente seja criada confiança entre todos os elementos.

Tabela 4- Descrição dos 5 valores do Scrum

Valor	Descrição
Comprometimento	Todos os participantes em projetos <i>Scrum</i> devem estar comprometidos em alcançar os objetivos definidos pela equipa Scrum.
Coragem	É necessário coragem para fazer o que está correto e para encarar problemas difíceis.
Foco	O foco de todos os participantes deve ser completar o trabalho definido para cada Sprint e atingir os objetivos da equipa Scrum
Abertura	A equipa Scrum e as partes interessadas devem estar completamente abertas para interagirem entre si sobre o trabalho e sobre os desafios inerentes ao trabalho em desenvolvimento.
Respeito	Todos os membros da equipa Scrum devem respeitar os colegas e as suas decisões.

Os projetos que seguem a metodologia Scrum devem ser realizados por equipas Scrum, dentro das quais existem três tipos de papéis distintos, nomeadamente, *Product Owner*, *Scrum Master* e *Development Team* (Sutherland & Schwaber, 2016). Cada um destes papéis possui responsabilidades diferentes, conforme descreve a Tabela 5.

Tabela 5- Papéis e responsabilidades do Scrum

Papel	Responsabilidade
<i>Product Owner</i>	Tem a responsabilidade de maximizar o valor do produto e do trabalho da Equipa de Desenvolvimento. Todas as suas decisões devem ser respeitadas. Tais decisões podem ser verificadas no conteúdo e ordem dos itens no <i>Backlog</i> de Produtos uma vez que o Dono do Produto é o único responsável pela sua gestão.
<i>Scrum Master</i>	Responsável por assegurar que o Scrum é compreendido por todos os participantes do projeto e que, de facto, é posto em prática.
<i>Development Team</i>	Papel atribuído a profissionais que executam o trabalho de modo a que o estado do incremento no final de cada <i>Sprint</i> seja “Feito”. Estas “Equipas de Desenvolvimento” têm a liberdade de se autogovernarem e são os únicos responsáveis pela transformação dos itens do <i>Backlog</i> de Produtos em incrementos funcionais. Os seus elementos devem ser multifuncionais.

Para além destes papéis associados aos constituintes da equipa Scrum, existem ainda mais duas entidades que interagem com o projeto e que estão envolvidos com regularidade no processo, nomeadamente o cliente e a gestão da organização. O cliente participa sobretudo na definição e priorização dos itens do *Backlog* de Produtos. A gestão participa na definição de objetivos e de requisitos e tem o poder de tomar a decisão definitiva, sendo também responsável pela escolha do *Product Owner* (Abrahamsson et al., 2002).

O Scrum prescreve um conjunto de eventos que devem ser realizados de modo a que ocorram reuniões com regularidade e para que se minimize a necessidade de reuniões extras. Todos estes eventos são limitados no tempo, isto é, têm uma duração máxima. Sobretudo o *Sprint* não deve ser alargado nem encurtado ao longo de um projeto (Sutherland & Schwaber, 2016). De seguida são descritos os eventos prescritos pela metodologia Scrum (Sutherland & Schwaber, 2016).

Sprint

Intervalo de tempo que cada iteração leva a ser realizada, sendo que o mais habitual são períodos de até um mês. Cada *Sprint* deve ser encarado como um projeto em que se pretende alcançar um determinado objetivo. Estes devem ser planeados durante as Reuniões de Planeamento do *Sprint*. Também em cada *Sprint*, como resultado da execução de todos os itens do *Backlog* do *Sprint*, deve ser desenvolvido algo funcional para o cliente.

Reunião de planeamento do *Sprint*

Este evento tem como objetivo planear tudo o que será realizado durante o próximo *Sprint*. Todos os membros da equipa Scrum devem estar envolvidos neste evento. A realização deste evento é assegurada pelo *Scrum Master*. Citando Sutherland e Schwaber (2016), durante estes eventos devem ser respondidas as perguntas seguintes:

- “O que pode ser entregue no Incremento resultante do próximo *Sprint*?”
- “Como é que o trabalho necessário para entregar o Incremento pode ser conseguido?”

Reunião diária Scrum

Como o próprio nome indica, este evento consiste numa reunião diária em que a Equipa de Desenvolvimento deve debater o progresso do *Sprint*. Citando mais uma vez Sutherland e Schwaber (2016), durante estas reuniões há três perguntas que devem ser respondidas:

- “O que foi feito desde a última reunião diária Scrum que contribui para cumprir os objetivos do Sprint?”
- “O que é que vai fazer hoje para ajudar a Equipa de Desenvolvimento a cumprir os objetivos do Sprint?”
- “Existe algum impedimento que impeça a Equipa de Desenvolvimento de atingir os objetivos do Sprint?”

Estas reuniões devem ser breves, não devendo durar mais do que 15 minutos. Nestas reuniões deve ser analisado o trabalho feito desde a reunião diária anterior e previsto o trabalho que se espera ter concluído até à próxima, para responder às perguntas acima enumeradas.

Reunião de revisão do *Sprint*

Esta reunião ocorre depois de o Sprint terminar e visa inspecionar o Incremento desenvolvido e modificar o *Backlog* de Produtos consoante as necessidades que surjam. O Incremento é apresentado a todas as partes interessadas, ou seja, aos clientes, à gestão, ao *Product Owner*, ao *Scrum Master*, à Equipa de Desenvolvimento e a qualquer outra pessoa. É uma reunião aberta onde todos os participantes verificam o Incremento resultante. Depois de avaliado o Incremento, são debatidas quais as atividades que se irão seguir. Nestas reuniões é comum que surjam novos itens para o *Backlog* do Produto e/ou que se alterem as prioridades de itens já existentes.

Retrospectiva do *Sprint*

Este evento ocorre depois da Reunião de revisão do Sprint e antes da próxima Reunião de planeamento do Sprint. Esta reunião deve-se focar na autoinspeção por parte da equipa do que correu bem e do que correu mal durante o Sprint e deve ser criado um plano para que os aspetos menos positivos possam ser melhorados durante o Sprint seguinte.

Uma nota importante para esta reunião é que, para que possa ser eficaz, deve ocorrer numa atmosfera de confiança e os participantes devem demonstrar maturidade emocional. Estes aspetos são importantes pois nesta reunião o objetivo é avaliar o processo e é determinante que os membros se sintam confortáveis para levantar questões que produzam soluções, sem fazer acusações. Simultaneamente, quando estas questões são levantadas é necessário que os restantes participantes demonstrem maturidade para absorver as opiniões dos colegas e sejam capazes de ajudar a equipa a encontrar uma solução para o problema detetado (Sutherland, 2016).

Os artefactos definidos pelo Scrum foram concebidos para maximizar a transparência de informação chave para que todos possam ser entendidos por todos. Na metodologia são prescritos três artefactos: *Backlog* do Produto, *Backlog* do Sprint e o Incremento (Sutherland & Schwaber, 2016).

***Backlog* do Produto**

Este artefacto é uma lista ordenada de tudo o que poderá ser necessário no produto final e é a única fonte de requisitos para qualquer mudança que venha a ser feita no produto (Sutherland & Schwaber, 2016). Todo o trabalho a ser feito é definido através dos itens do *Backlog* do Produto. Este artefacto é atualizado várias vezes no decorrer do projeto, evoluindo à medida que o produto evolui. Os itens encontrados neste artefacto podem ser funções, funcionalidades, correção de defeitos, pedidos de melhoramentos do produto e melhorias nas tecnologias utilizadas. Como já foi referido na Tabela 5, o *Product Owner* é o responsável por este artefacto.

***Backlog* do Sprint**

Este artefacto é um subconjunto de itens selecionados do *Backlog* do Produto que serão desenvolvidos durante o próximo *Sprint*. Este artefacto costuma ser acompanhado por um plano de como entregar o Incremento e cumprir os objetivos do *Sprint*. Estes itens são escolhidos durante a reunião de planeamento de cada *Sprint* e a sua seleção tem como base a ordem de prioridades do *Backlog* do Produto e os objetivos delimitados para aquele *Sprint*. Este artefacto, durante o *Sprint*, só pode ser alterado pela Equipa de Desenvolvimento.

Resumidamente, constitui uma representação, em tempo-real, do trabalho que está planeado para ser executado pela Equipa de desenvolvimento durante o Sprint em questão.

Incremento

Pode ser definido como a soma de todos os itens do *Backlog* do Produto que foram concluídos durante o Sprint e do valor de todos os incrementos dos Sprints anteriores (Sutherland & Schwaber, 2016).

Segundo Schwaber e Beedle (2002), o processo completo do Scrum é composto por três fases: (1) pré-jogo, (2) jogo (desenvolvimento) e (3) pós jogo.

A fase do pré-jogo inclui duas subfases. A primeira subfase é o planeamento, onde é criado o *Backlog* do Produto inicial. Os itens do *backlog* são priorizados e é feita uma estimativa do esforço para o

seu desenvolvimento. A segunda subfase consiste na conceção da arquitetura de alto nível dos requisitos. Num caso do projeto se tratar de um melhoramento de um sistema existente, as mudanças necessárias para implementar o *Backlog* do Produto deve ser acompanhado pelos problemas que podem resultar dessas mudanças

A fase de desenvolvimento (Game Fase) constitui a parte ágil do Scrum. Esta fase consiste na realização dos vários *Sprints* realizando-se o ciclo ilustrado na Figura 9.

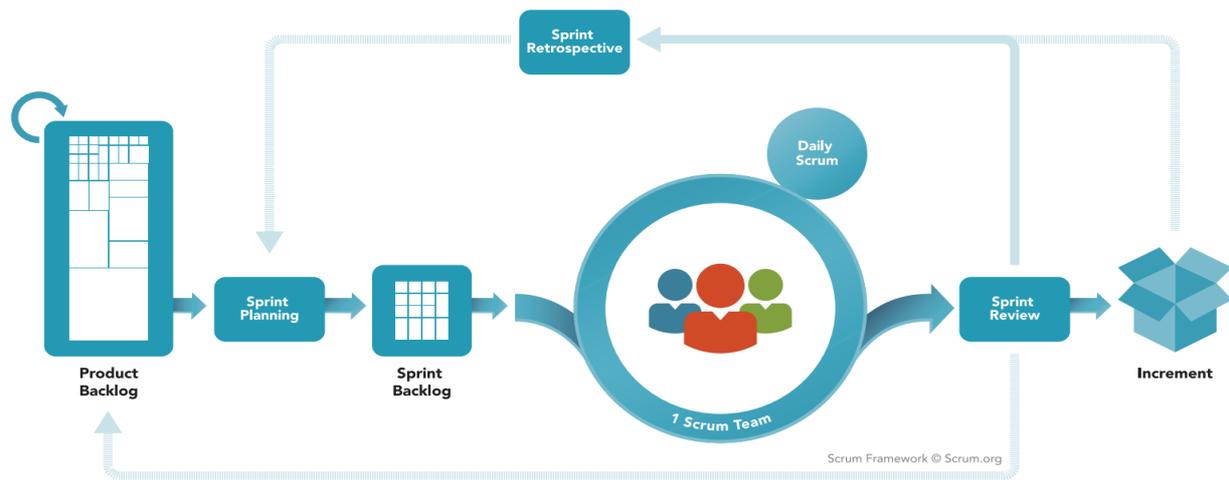


Figura 9- Framework do Scrum (retirado de (Scrum.org, 2018))

A fase do pós-jogo contém o encerramento e o lançamento final. Entra-se nesta fase quando se chega a um acordo quanto ao cumprimento dos requisitos. Nesta fase faz-se a integração do sistema como um todo, correm-se os testes do sistema completo, finaliza-se a documentação e é feito lançamento final do produto.

2.5.2 Extreme Programming

Extreme programming, (XP) é uma das populares metodologias ágeis de desenvolvimento de *software*. O XP foi inicialmente desenvolvido para suportar equipas de desenvolvimento de *software* de pequena dimensão, que lidavam constantemente com mudanças nos requisitos (Beck, 1999b). Tal suporte é conseguido através do respeito de um conjunto de valores e princípios que devem ser aplicados através de um conjunto de práticas.

As práticas não são originais mas sim práticas que nas décadas anteriores se verificaram eficazes no processo de desenvolvimento de *software*. Apesar de não serem novas, elas foram alinhadas para funcionarem umas com as outras de uma nova maneira formando assim uma nova metodologia para o desenvolvimento de *software* (Abrahamsson et al., 2002). O próprio termo “Extreme” surge porque se adotaram essas práticas do senso comum e levaram-se a níveis extremos (Beck, 1999b). Elas foram

reunidas e aplicadas com sucesso em alguns projetos em múltiplas organizações e viriam, mais tarde, a ser utilizadas na formalização da metodologia XP por Kent Beck (1999b).

Inicialmente foram enunciados quatro valores chave, nomeadamente, comunicação, simplicidade, *feedback* e coragem. (Beck, 1999b) Mais tarde foi adicionado um quinto valor, o respeito (Beck & Andres, 2004). Estes cinco valores fundamentais devem ser seguidos durante o desenvolvimento de qualquer projeto XP e encontram-se descritos na Tabela 6.

Tabela 6- Descrição dos 5 valores do XP

Valor	Descrição
Comunicação	A má comunicação entre os participantes do projeto é uma das principais causas de projetos fracassados. No XP, a comunicação é assegurada através da promoção de práticas que requerem a comunicação entre intervenientes, obrigando-os a comunicarem entre si. No entanto, como em qualquer projeto, há sempre a possibilidade de surgirem obstruções que impeçam a comunicação num projeto XP. Para amenizar tais problemas, o XP encarrega o <i>Coach</i> de garantir que todos os participantes comunicam entre si do modo desejado.
Simplicidade	Deve ser respondida a pergunta “ Qual é a coisa mais simples que pode, possivelmente, funcionar?”
<i>Feedback</i>	O <i>feedback</i> concreto sobre o estado atual do sistema é de um valor inestimável. Funciona em várias escalas, desde minutos até meses.
Coragem	É preciso ter coragem no momento de reestruturar algo que já tenha sido feito, mesmo que já esteja num estado avançado. Da mesma forma, também é preciso coragem para deitar código fora. Este valor revela o seu verdadeiro valor quando combinado com os três anteriores.
Respeito	Este valor suporta os restantes quatro. Para que os outros valores possam ser respeitados, o projeto deve ser realizado num ambiente de confiança que só pode existir com base em interações respeitosas entre participantes. Tais interações envolvem a compreensão e o respeito das opiniões de todos os participantes.

No entanto, os valores por si só podem ser interpretados como algo vago e pouco objetivo e por isso, o XP providencia cinco princípios (Beck, 1999b). Os cinco princípios fundamentais são, *feedback* rápido, presumir simplicidade, mudança incremental, abraçar a mudança e trabalho de qualidade. Estes princípios são explicados na Tabela 7.

Tabela 7- Descrição dos 5 princípios fundamentais do XP

Princípio	Descrição
Feedback rápido	O <i>feedback</i> recebido deve ser interpretado de modo a que o que é aprendido seja rapidamente colocado em prática e implementado no sistema.
Presumir simplicidade	Todos os problemas com que a equipa do projeto se depara devem ser assumido como sendo fáceis de resolver.
Mudança incremental	Cada grande problema deve ser dividido e tratado como problemas menores, evitando-se fazer grandes alterações de uma só vez. Também durante o decorrer do projeto várias pequenas alterações irão ocorrer, quer seja na conceção, no plano ou até mesmo na equipa.
Abraçar mudanças	A melhor estratégia é aquela que preserva o máximo de opções enquanto realmente se resolve o problema mais relevante. A equipa deve estar aberta a eventuais mudanças que possam surgir durante o ciclo de desenvolvimento.
Trabalho de qualidade	Permitir que a equipa possa entregar trabalho de qualidade aumenta a satisfação com o seu emprego.

O XP foca-se em permitir que o desenvolvimento de *software* possa ter sucesso, independentemente da alteração constante de variáveis do projeto (e.g. os requisitos), seguindo os valores e os princípios já descritos na Tabela 6 e na Tabela 7. Devem ser feitas pequenas iterações com pequenas versões do *software* que devem gerar incrementos que serão continuamente integrados e testados com o *software* já existente (Abrahamsson et al., 2002). Estas são apenas algumas das características do XP pois, segundo Beck (1999a), o XP possui doze grandes práticas. Estas podem ser explicadas, resumidamente, da seguinte forma:

- **Jogo de Planeamento:** A interação entre clientes e programadores deve ser próxima. Durante o jogo de planeamento os requisitos devem ser convertidos em *Story Cards* (ver Figura 10). Depois os programadores estimam o esforço necessário para implementar os requisitos e os clientes definem o âmbito e o *timing* dos lançamentos. Os programadores só devem implementar as funcionalidades pedidas para as histórias associadas a cada iteração.

- **Pequenas entregas:** O sistema é posto em produção rapidamente, pelo menos uma vez a cada 2/3 meses. Novas versões são depois lançadas frequentemente podendo ser desde diárias até mensais.
- **Metáfora:** O sistema é definido por uma metáfora ou por um conjunto de metáforas, partilhadas entre o cliente e os programadores. Estas metáforas guiam todo o desenvolvimento ao descreverem como o sistema funciona.
- **Design simples:** A solução apresentada, a qualquer momento, deve ser o mais simples possível. Isto implica que complexidade desnecessária deve ser removida do código. Não deve existir código duplicado e o número de classes e métodos deve ser o menor possível.
- **Testes:** O desenvolvimento de *software* é orientado pelos testes. Os testes devem ser corridos continuamente durante o projeto. Os clientes escrevem os testes funcionais para cada iteração.
- **Refactoring:** O *design* do sistema é evoluído através de transformações feitas ao *design* existente.
- **Programação por pares:** O código deve ser pensado e escrito por duas pessoas num só computador.
- **Integração contínua:** Pedacos de novo código devem ser integrados no sistema atual numa questão de horas. No momento da integração o sistema é construído de novo e todos os testes devem ser corridos e passados com sucesso de modo a que o novo incremento possa ser aceite.
- **Propriedade coletiva:** Qualquer programador pode melhorar qualquer parte do código, a qualquer momento.
- **Cliente no local:** Um cliente deve estar presente e disponível para a equipa de desenvolvimento a tempo integral.
- **40 Horas semanais:** Ninguém pode trabalhar duas semanas consecutivas com horas extras. O limite semanal é de 40 horas por semana, no máximo. Em caso de ocorrência, esse tempo extra deve ser tratado como um sinal de um problema que precisa de ser resolvido.
- **Espaço de trabalho amplo:** A equipa deve trabalhar numa sala grande com pequenos cubículos na periferia da sala. Quem programa a pares deve ser colocado no centro da sala.
- **Apenas regras:** ao pertencer a uma equipa *Extreme* o participante deve, por defeito, seguir as regras associadas à metodologia. No entanto a equipa pode alterar as regras, em qualquer momento, desde que essas alterações sejam aceites por todos e que o seu impacto seja avaliado.

Customer Story and Task Card B/W Development / COLA

DATE: 3/19/98 TYPE OF ACTIVITY: NEW: FIX: ENHANCE: FUNC. TEST:

STORY NUMBER: ~~1275~~ 1275 PRIORITY: USER: _____ TECH: _____

PRIOR REFERENCE: _____ RISK: _____ TECH ESTIMATE: _____

TASK DESCRIPTION:
 SPLIT COLA: When the COLA rate slips in the middle of the B/W Pay Period, we will want to pay the 1st week of the pay period at the OLD COLA rate and the 2nd week of the pay period at the NEW COLA rate. Should occur automatically based on system design.

NOTES: on system design.
 For the OT, we will write a program that will pay or calc the COLA on the 2nd week of OT. The plant currently retransmits the hours data for the 2nd week exclusively so that we can calc COLA. This will come into the Model as a "2144" COLA

TASK TRACKING: Gross Pay Adjustment, Create RM Boundary and Place in DEEnt Exec COLA

Date	Status	To Do	Comments

Figura 10- Story Card (retirado de (Beck, 1999b))

Esta metodologia considera a existência de vários papéis dentro de uma equipa de projeto, os quais variam entre si relativamente às responsabilidades que possuem, no contexto de um determinado projeto. Tendo em conta as definições dadas para cada um destes papéis por Beck (1999b), a Tabela 8 descreve, resumidamente, as responsabilidades associadas a cada um.

Tabela 8- Papéis e responsabilidades do XP

Papel	Responsabilidades
Programador	Escreve o código e os testes necessários. Deve mantê-los o mais simples e definitivos possíveis
Cliente	Responsável por escrever as histórias e os testes funcionais. Define a prioridade dos requisitos e decide quando um requisito está cumprido.
Testador	Responsável por correr os testes funcionais regularmente e por divulgar os resultados. Pode auxiliar o cliente na escrita de testes funcionais.
Rastreador	Verifica as estimativas dadas pela equipa e dá o <i>feedback</i> da precisão para melhorar futuras estimativas. Também é responsável por seguir o progresso de cada iteração e avaliar se os objetivos previstos podem ser cumpridos consoante os recursos e restrições existentes.
Coach	Responsável pelo processo como um todo e por guiar os restantes participantes para seguirem o processo e os valores fundamentais do XP. Este papel deve ser ocupado por alguém que possua um entendimento profundo do XP.
Consultor	Responsável por ajudar a equipa a resolver problemas específicos. É um membro externo à equipa do projeto que possui o conhecimento técnico específico para as necessidades identificadas.

Papéis	Responsabilidades
Gestor	Responsável pelas tomadas de decisão. Tais decisões devem ser baseadas no estado, limitações e dificuldades do projeto, sendo estas levantadas através da comunicação com os restantes membros da equipa do projeto.

Não existem dois projetos XP exatamente iguais e como tal, o fluxo do projeto varia entre projetos. Segundo Beck (1999b), um ciclo de vida de um projeto XP ideal consiste num conjunto de seis fases, nomeadamente: Exploração, Planeamento, Iterações para lançamento, Produção, Manutenção e Morte (ver Figura 11).

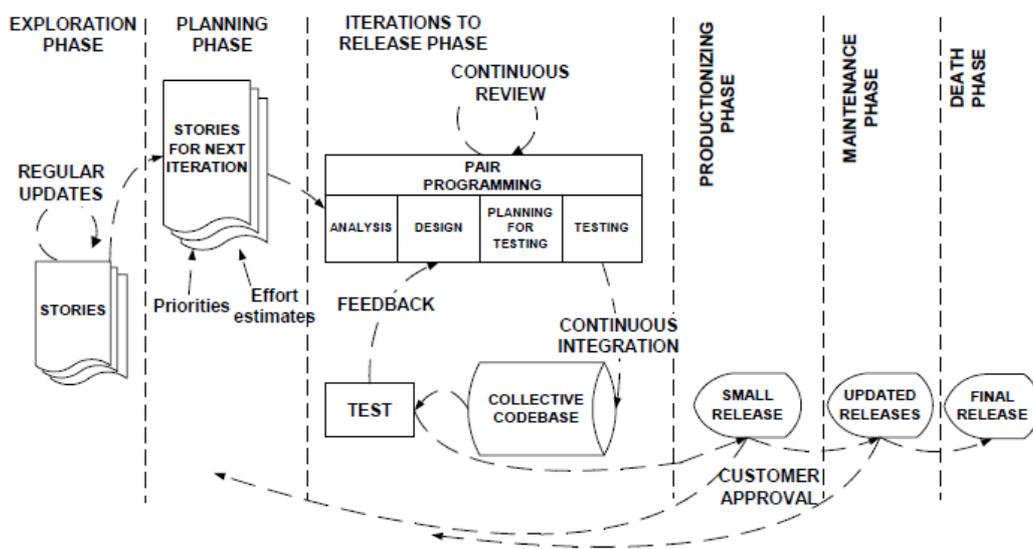


Figura 11- Ciclo de vida do processo do XP (retirado de (Abrahamsson et al., 2002))

A fase de Exploração acontece quando os membros da equipa começam a ganhar confiança entre si, a ganhar confiança nas tecnologias que vão utilizar, a se familiarizarem com as práticas e acima de tudo, quando estão aptos para partir para a produção. O tempo desta fase varia conforme o nível de conhecimento que os membros da equipa tem entre si e das tecnologias a serem utilizadas. Numa equipa que já se conheça e que esteja minimamente familiarizada com a tecnologia, esta fase pode durar apenas umas semanas. Numa equipa que seja totalmente novata numa determinada tecnologia já dependeria alguns meses nesta fase. Para além desta exploração, os clientes também escrevem as histórias que querem ver concluídas. Para terminar esta fase os clientes têm de estar confiantes que existe material suficiente para alcançar uma boa primeira versão e os programadores têm de estar confiantes em todos os aspetos já falados neste parágrafo.

A fase que se segue é a de Planeamento com o propósito dos clientes e programadores acordarem uma data em que o menor e mais valioso conjunto de histórias tenha de estar pronto. O plano para o primeiro lançamento deve durar entre dois e seis meses.

Na fase de iterações o calendário acordado na fase anterior é dividido em iterações que demorem entre uma a quatro semanas. Na primeira iteração são escolhidas as histórias para criar um a arquitetura de todo o sistema. Para as iterações seguintes as histórias serão escolhidas pelos clientes e os testes funcionais dos clientes são corridos no final de cada iteração. No fim da última iteração o sistema estará pronto para ir para produção.

A fase da produção requer que seja verificado o desempenho do sistema antes de este ser lançado para o cliente e será testado para provar que está preparado para esse lançamento. As ideias que possam surgir para acrescentar à primeira versão devem ser registadas para que possam ser implementadas mais tarde.

Na fase de manutenção produz-se novas funcionalidades, mantem-se o sistema já produzido a correr e eventualmente podem-se incorporar novas pessoas na equipa. Todas estas atividades em simultâneo fazem com que a velocidade de desenvolvimento desacelere.

A fase da morte inicia-se quando um cliente não tem mais histórias que necessitem ser implementadas, ou seja, o sistema satisfaz todas as necessidades do cliente. Nesta fase escreve-se a documentação do sistema uma vez que a partir deste momento ele não irá sofrer mais alterações. Outras razões para o sistema morrer poderão ser porque não é capaz de entregar as funcionalidades que o cliente deseja, ou por motivos económicos.

2.5.3 Kanban

No final da década de 1940, a Toyota era à data uma empresa de pequena dimensão que pretendia construir carros acessíveis. A resposta mais comum para este problema era construir carros em massa mas não havia mercado suficiente. Para solucionar este problema surgiu o Sistema de Produção da Toyota, para repensar todo o processo de construção, logística e entrega dos automóveis, iniciando-se assim a corrente de pensamento *Lean*. Este sistema foi pensado por Taiichi Ono que instituiu como valor principal a eliminação do desperdício, redefinindo o termo “desperdício” como tudo o que não cria valor para o cliente (Poppendieck & Poppendieck, 2003). A partir do Sistema de Produção da Toyota foi desenvolvido o sistema kanban, que é um sistema de visualização que recorre a cartões e cujo principal objetivo é gerir e potencializar o processo de fabrico, evitando desperdício.

A adaptação do Kanban para o desenvolvimento de *software* foi feita por David Anderson (Stellman & Greene, 2014) e em 2007 foi formalizado o método Kanban (Anderson & Carmichael, 2016). No seu livro, Anderson (2010) define o Kanban como o método de mudança evolucionário que utiliza a visualização por cartões e outras ferramentas para catalisar a introdução de ideias *Lean* no desenvolvimento de tecnologias e operações de tecnologias de informação.

O Kanban não é um método para gerir projetos mas sim para melhorar os processos usados por equipas ágeis (Anderson, 2010). Similar ao Scrum e ao XP, o Kanban também tem os seus próprios princípios específicos, estando neste caso alinhados com a mentalidade *Lean*. Este método consiste num processo incremental cujo objetivo é limitar o número de atividades que decorrem em simultâneo e eliminar o desperdício.

No entanto, ao contrário do Scrum e do XP, o Kanban não define qualquer papel nem quaisquer eventos ou artefactos. Apenas fornece um conjunto de princípios e práticas, o conceito do quadro Kanban e o conceito de fluxo de trabalho. Os seus princípios fundamentais (ver Tabela 9) são fulcrais para que a aplicação das práticas sugeridas seja bem-sucedida. Estes princípios podem ser divididos em dois grupos: os princípios de gestão de mudança e os princípios de entrega de serviço (Anderson & Carmichael, 2016).

Os princípios de gestão da mudança abordam os aspetos humanos das organizações, nomeadamente, a resistência à mudança despoletada pela natureza psicológica e social da rede de indivíduos que constitui a organização. Os princípios de entrega de serviços abordam o facto de que qualquer organização com um tamanho considerável funciona como um ecossistema de serviços interdependentes (Anderson & Carmichael, 2016).

Da aplicação do método Kanban apenas podem resultar melhorias no fluxo de gestão e melhorias contínuas nos processos de desenvolvimento de *software* (Anderson, 2010). Para além destes aperfeiçoamentos, este método também oferece a possibilidade de as organizações adotarem os valores e princípios ágeis sem terem de fazer mudanças consideráveis nas suas culturas.

Em suma, o Kanban não é uma metodologia de gestão de projetos. O seu principal foco é o melhoramento dos processos que é alcançado com base nos seis princípios fundamentais descritos na Tabela 9.

Tabela 9- Descrição dos 6 princípios fundamentais do Kanban

Tipo de princípios	Princípio	Descrição
Princípios de gestão da mudança	Começar com o que já se faz	O Kanban visa melhorar os processos de uma equipa de desenvolvimento de <i>software</i> . Só é possível melhorar algo se existir algo para melhorar. Como ponto de partida devem ser utilizados os processos que atualmente a equipa realiza. Este princípio visa diminuir a resistência à mudança das pessoas envolvidas.
	Perseguir mudança incremental e evolucionária	Depois da equipa ter começado com o que já se faz, o objetivo será fazer pequenos melhoramentos a esse sistema de modo a que a resistência à mudança seja o mínimo possível. O Kanban incentiva mudanças incrementais e evolucionárias pois criam menos resistência e incerteza do que mudanças radicais.
	Inicialmente, respeitar os papéis, responsabilidades e cargos existentes.	Visto que é reconhecido valor no processo existente, também é reconhecido valor nos papéis, responsabilidades e cargos que já existem. Com as melhorias incrementais que poderão ocorrer, os papéis, responsabilidades e cargos poderão ser alterados mas o Kanban não prescreve quais papéis deverão existir.
Princípios de entrega de serviço	Entender e focar nas necessidades e expectativas do cliente	A procura por melhoramentos do sistema de desenvolvimento deve ser orientada aos clientes. É necessário um entendimento claro de quem são os clientes, o que esperam da equipa e quais as necessidades que verem ver resolvidas.
	Gerir o trabalho deixando as pessoas auto-organizarem-se em torno dele	As tarefas de trabalho de conhecimento criativo características do desenvolvimento de <i>software</i> não constituem um problema determinístico. Em vez de se gerir as pessoas, gere-se o trabalho e deixa-se que os colaboradores escolham e produzam o que é pedido, entreguem ao cliente e que avancem para o próximo trabalho. O quadro Kanban permite visualizar o trabalho desta perspetiva.

Tipo de princípios	Princípio	Descrição
	Evoluir as políticas para melhorar os resultados da organização e dos clientes	As políticas ditam como, quem e em que ordem se deve realizar o trabalho. Os processos existentes devem ser dissecados num conjunto de políticas. Estas devem ser definidas explicitamente, tornando-as discutíveis o que tornará possível a sua revisão e consequente evolução.

Tendo já estes princípios interiorizados, deve ser adotado um conjunto de práticas cujo objetivo é providenciar uma forma de estabilizar e melhorar o sistema de desenvolvimento de *software*. De salientar que não é expectável que estas práticas sejam todas adotadas inicialmente, deve ser um processo gradual (Stellman & Greene, 2014). As práticas são descritas sucintamente na lista que se segue:

- **Visualizar:** o método Kanban recorre a quadros com cartões para facilitar a visualização do fluxo de trabalho, do início ao fim. Podem conter os limites de WIP (*Work in Progress*), as políticas quanto ao trabalho que deve ser desenvolvido, a ponto de entrega, e o ponto de comprometimento;
- **Limitar o trabalho em progresso (WIP):** Devem ser impostos limites quanto ao trabalho que pode estar simultaneamente em progresso, de modo a tornar possível as entregas mais frequentes e se possa reduzir os tempos de espera.
- **Gerir e otimizar o fluxo de processos:** Deve-se tentar encontrar, continuamente, formas de melhorar os processos. Estes melhoramentos são muitas vezes conseguidos através da redução de tempos de espera e da deteção e remoção de obstáculos do fluxo.
- **Tornar as políticas dos processos explícitas:** as políticas devem ser simples, bem-definidas, visíveis e prontas para serem alteradas pelas pessoas a trabalharem no processo associado.
- **Implementar loops de feedback:** Esta recorrência na recolha e análise de *feedback* é fundamental para qualquer sistema que procure mudança evolucionária.
- **Melhorar de forma colaborativa e evoluir experimentalmente:** Este método inicia-se com o estado atual de um determinado processo e vai sendo melhorado incrementalmente.

No Kanban adaptado para desenvolvimento de *software* é sugerido que o quadro *kanban* organize as suas colunas consoante as fases de desenvolvimento que a equipa costuma utilizar (ver Figura 12). A sequência de todas as atividades do Kanban dá-se o nome de fluxo de trabalho. Devido à natureza de

melhoramentos incrementais e contínuos do método, é natural iniciar-se com um quadro simples que pode evoluir para um mais complexo. A primeira coluna costuma ser um *backlog* de todas as atividades que são necessárias ser concluídas e à medida que vão sendo desenvolvidas o seu respetivo cartão deve mover-se para as colunas mais à direita. A coluna mais à direita costuma ser a das atividades concluídas (Anderson & Carmichael, 2016).

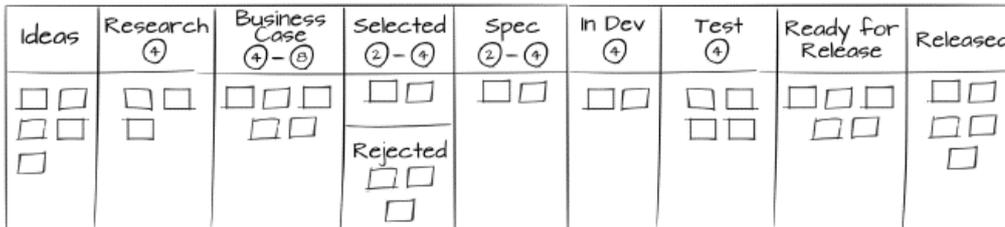


Figura 12- Exemplo de um quadro kanban (retirado de (Anderson & Carmichael, 2016))

2.6 Desenvolvimento ágil de larga escala

As abordagens ágeis foram inicialmente concebidas para equipas de pequena dimensão, independentes e a trabalharem na mesma localização mas, hoje em dia, são também aplicadas em projetos de *software* de grandes dimensões, por vezes com várias equipas a trabalharem em localizações distintas (Paasivaara et al., 2012). Estas abordagens não tecem quaisquer considerações relativamente aos esforços de desenvolvimento de *software* para além do nível de projeto, isto é, não oferecem recomendações para escalar o desenvolvimento para além do nível da própria equipa (Dikert et al., 2016).

Para além disso, a definição de “desenvolvimento ágil de larga escala” é pouco consensual. Por exemplo, é utilizada para descrever esforços de desenvolvimento ágil em que o número de equipas de desenvolvimento é maior do que dois (Dingsøyr et al., 2014), no entanto esta definição em particular é limitada a apenas uma dimensão, o número de equipas. A designação é também utilizada quer para descrever desenvolvimento ágil com base numa variedade de outros critérios (como o número de pessoas envolvidas no desenvolvimento, linhas de código do produto final e número de locais de desenvolvimento), quer para descrever a utilização de abordagens ágeis em grandes organizações (Dingsøyr & Moe, 2014).

Esta área tem sido identificada como desafiante desde 2003, época em que já se debatiam as dificuldades de conciliar os métodos ágeis com níveis mais elevados do que apenas o da gestão da equipa do projeto (Reifer et al., 2003). Também tem sido identificada como uma questão proeminente

pelos praticantes, que votaram “ágil de escala” como questão crucial e prioritária de investigação (Freudenberg & Sharp, 2010).

À medida que as práticas de desenvolvimento ágil foram sendo utilizadas por projetos de larga escala, novos desafios foram surgindo. Os principais desafios identificados, inerentes a esta atividade, estão representados na Tabela 10, estando organizados decrescentemente quanto ao seu grau de importância (Dingsøyr & Moe, 2013).

Tabela 10- Agenda de pesquisa sugerida para o desenvolvimento ágil de larga escala (retirado de (Dingsøyr & Moe, 2013))

Rank	Tópico	Descrição
1	Coordenação entre equipas	Coordenação do trabalho entre equipas no desenvolvimento de larga escala
2	Organização com projetos grandes/ gestão de portefólio	Quais são as organizações estruturais e os modelos de colaboração eficazes, em grandes projetos? Como lidar com uma organização distribuída?
3	Arquitetura de planeamento de lançamentos	Como é que os grandes projetos são planeados? Como é que o âmbito pode ser reduzido? Qual o papel da arquitetura no ágil de larga escala?
4	Escalar práticas ágeis	Quais práticas ágeis podem ou não ser escaladas? Porquê e quando é que as práticas ágeis escalam?
5	Colaboração do cliente	Como é que os donos do produto e os clientes colaboram com os desenvolvedores em projetos de larga escala?
6	Transformação ágil em grande-escala	Como é que as práticas ágeis podem ser adotadas eficientemente em grandes projetos?
7	Partilha de conhecimento e melhorias	Quando é que o quadro branco não é suficiente? Como é que as comunidades de prática podem ser estabelecidas? Quais medidas são relevantes para promover melhorias?
8	Contratos ágeis	Como é que os contratos podem alterar a mentalidade dos clientes de planeamento adiantado para os princípios ágeis? Quais as limitações legais existentes em contratos que reduzem a agilidade de grandes projetos?

Como resposta a estes desafios, várias *frameworks* foram criadas para estender as práticas ágeis para o ambiente de larga escala (Vaidya, 2014). Em 2017, as mais utilizadas eram, respetivamente, o

Scaled Agile Framework (29%), o *Scrum-of-Scrums* (19%), métodos criados internamente (10%), o *Disciplined Agile Delivery* (5%), o *Large Scale Scrum* (5%), *Enterprise Scrum* (3%), *Lean Management* (3%), *Agile Portfolio Management* (3%), *Nexus* (1%) e *Recipes for Agile Governance in the Enterprise* (menos de 1%) (Versionone.com, 2018).

Quando analisadas, embora sejam todas semelhantes na perspectiva em que partilham o mesmo objetivo de resolverem o problema da escalabilidade ágil, estas *frameworks* apresentam discrepâncias em vários aspetos como o treino e certificação recomendada, os métodos e práticas a adotar, a dimensão das equipas, as técnicas que devem ser utilizadas e a forma como se organizam (Alqudah & Razali, 2016). Ao contrário do caso das abordagens ágeis, destas *frameworks* nenhuma se destaca claramente como a mais utilizada, evidenciando um mercado dividido.

Para além disso, embora as empresas estejam a reportar que utilizam estas *frameworks*, os casos de uso encontrados na literatura são muito escassos (Dikert et al., 2016). Portanto, o estudo científico destas *frameworks* é fundamental para entender como é que são usadas e como é que podem ser customizadas (Dikert et al., 2016).

De modo a promover a investigação e a aprofundar o conhecimento sobre escalabilidade ágil, Dingsoyr e Moe (2014) realizaram um seminário dedicado à identificação dos potenciais princípios do desenvolvimento ágil em larga escala. O seminário focou-se em quatro aspetos, nomeadamente, arquitetura, coordenação entre equipas, gestão do portefólio e escalação de práticas.

Relativamente à arquitetura, propõem os dois seguintes princípios:

- A arquitetura tem um papel fundamental em definir como é que o trabalho é coordenado em esforços de desenvolvimento de larga escala;
- O nível de mudança e o nível de incerteza vão influenciar como é que os trabalhos arquitetónicos devem ser organizados;

No que toca à coordenação entre equipas, os seguintes princípios foram propostos:

- Normas e valores comuns facilitam a coordenação entre equipas;
- Redes eficazes de conhecimento são essenciais no desenvolvimento de larga escala devido à natureza intensiva de conhecimento do desenvolvimento de *software*;

Quanto à gestão de portefólios, os autores propõem os seguintes princípios:

- *Feedback* contínuo do portefólio para os níveis de projeto habilitam as equipas e os elementos do projeto a tomar decisões que são consistentes com os objetivos de portefólios ágeis de larga escala;

- *Feedback* contínuo do nível de projeto para o nível de portfólio possibilita que o portfólio seja alterado para otimizar o valor de portfólios ágeis de larga escala;

Finalmente, no que toca a escalar práticas ágeis, os seguintes princípios são sugeridos:

- Descrever o contexto para a agilidade e escala é essencial para entender como melhorar a agilidade da mentalidade ágil de larga escala;
- Para o desenvolvimento de sistemas embebidos em larga escala, a agilidade deve escalar respeitando o número de equipas envolvidas e as atividades de engenharia de sistemas, em cada iteração devido à dependência entre o desenvolvimento de *software* e *hardware*.

Embora ainda sejam muito recentes, o alinhamento com estes princípios permite que se sugiram soluções para muitas das problemáticas apresentadas. Assim sendo, é fundamental que o modelo resultante desta dissertação esteja alinhado com estes princípios.

3. CONCEPTUALIZAÇÃO DO *AGILE COORDINATION OFFICE*

Neste capítulo é apresentada a conceptualização do *Agile Coordination Office* (ACO) sendo que, primeiro, começa-se por apresentar a análise preliminar de trabalhos afetos a esta temática, onde se averiguou os valores, princípios e, sobretudo, práticas por eles propostos.

Segundo, é apresentado o processo de desenvolvimento incluindo a estrutura, composta por um conjunto de submodelos e módulos ajustáveis a um conjunto de fatores que influenciam a atividade de um PMO em contexto ágil, e como essa estrutura organiza as práticas recolhidas.

Terceiro, é apresentada a caracterização do modelo que foi desenvolvido com base quer nos casos de estudo analisados, quer no conhecimento recolhido durante a revisão de literatura.

Por fim, é feita uma exploração de como utilizar o ACO, sugerindo várias configurações conforme o cenário em que o modelo será implementado.

3.1 Análise de propostas semelhantes

Ao adotar processos ágeis, os PMOs têm de alterar a forma como funcionam porque embora possam continuar com o mesmo propósito de suportar os projetos sob sua alçada, as práticas que desempenha devem-se ajustar a um ambiente que acolhe a mudança (Sliger, 2007). Isto implica alterações por exemplo, na forma como recolhe métricas para apoiar a gestão executiva, no tipo de treino que providenciado ou no método de alocação de recursos.

O foco desta análise consistiu em explicar os casos de estudo encontrados sobre os chamados PMOs ágeis e sobre os PMOs em transição do tradicional para o ágil, assim como toda a literatura identificada que explora a conceptualização e/ou caracterização de uma estrutura organizacional semelhante à que se deseja que resulte desta dissertação. A partir desta análise foi possível: (1) recolher práticas e recomendações dos casos de estudo existentes, (2) analisar as estruturas que promovem, (3) fazer um levantamento dos desafios que enfrentam e (4) aprender com as experiências que vivenciaram. Todo este conhecimento foi utilizado posteriormente na concetualização do ACO.

3.1.1 Agile Office

Ken Power (Power, 2011) no seu artigo apresenta a experiência da unidade de negócio unificada de comunicações da Cisco ao estabelecer o chamado “Agile Office”. Neste artigo teve como base o livro

“Succeeding with Agile” de Mike Cohn (2010), onde é proposto que um PMO no momento de transição deve ser renomeado, dando como exemplo a designação *Scrum Office* pois a gestão dos projetos passaria a ser feita através de um gabinete gerido pela metodologia Scrum. Apesar de seguir este livro, Power e a sua equipa decidiram não adotar completamente este conceito por dois motivos: (1) porque a utilização do Scrum Office limitaria a adoção de práticas ágeis a uma só metodologia, nomeadamente o Scrum e (2) porque nem toda a organização iria transitar para as práticas ágeis, mantendo-se alguns projetos geridos por metodologias mais tradicionais. A solução apresentada tem, inicialmente, como objetivo principal governar a adoção de métodos ágeis e, mais tarde, assegurar a melhoria contínua através das práticas ágeis. Neste artigo apenas é relatada a fase inicial.

O *Agile Office* foi criado como uma entidade distinta, ao invés de tentar mudar o propósito do PMO existente. Os dois a trabalharem em conjunto apresentam como práticas a formação às equipas sobre abordagens ágeis, o estabelecimento de estruturas de comunicação e coordenação e o suporte do cumprimento de conformidades (e.g. requisitos regulamentares). Também os elementos da equipa a comunicar melhor com as outras partes interessadas, promovem a partilha de experiências, mantêm-se a par das tendências da comunidade ágil e assumem o papel de evangelizadores dos valores e práticas ágeis. Todas as equipas estão sob um programa. A forma como engajam com as equipas depende de (1) nível de fluência com práticas ágeis e (2) fase do ciclo de vida, do programa, em que se encontram.

O modelo de governação deste gabinete é como se tratasse de uma equipa ágil com uma equipa dona do produto, um *backlog* de trabalhos priorizados e com uma equipa que os vai realizando em iterações mensais. Mas maior parte desse trabalho, em todas as iterações, é o suporte das equipas.

No final o autor faz um conjunto de sugestões, baseadas nas experiências vivenciadas na sua organização. No entanto, apenas nos interessam as que dizem respeito ao próprio funcionamento do *Agile Office*, depois de a transição para o ágil estar terminada, visto que se nesta dissertação se deseja caracterizar um gabinete de suporte à gestão de projetos exclusivamente ágeis.

Dessas sugestões há quatro que se destacam: (1) deve ser dada autonomia ao *Agile Office*, (2) este deve ser operado com base em práticas ágeis, isto é, o *Agile Office* deve ser gerido da mesma forma que uma equipa ágil, (3) a gestão de topo precisa de estar ativamente envolvida, não pode só apoiar passivamente e (4) deve-se entender quem são as partes interessadas do *Agile Office* e deve-se garantir o seu envolvimento continuamente.

3.1.2 Agile PMO

Esta proposta descreve a experiência da divisão de TI da empresa Capital One Auto Finance em transitar de um PMO tradicional para um PMO ágil (Tengshe & Noble, 2007). Ao contrário do caso de estudo anterior, o objetivo desta transição foi converter o PMO já existente num PMO ágil.

No artigo de Tengshe e Noble (2007) é inicialmente dado um enquadramento de qual era a situação da divisão da empresa antes de se iniciar a transição para as abordagens ágeis. Trata-se de uma divisão de dimensão considerável (cerca de 500 pessoas) que utilizava metodologias baseadas no modelo de cascata ligeiramente diferentes consoante o tamanho do projeto. A utilização destas metodologias fazia com que demorasse muito tempo para entregar valor de negócio criando uma má imagem nos clientes.

Decidiram então começar a sua transição para a abordagem ágil. Citando os autores, eles destacam “ [...] como o PMO central (Portfolio Management Office) foi instrumental em vez de prejudicial na propagação, aceitação e suporte do Agile”. A abordagem de transição começou a ser pensada por um membro do próprio PMO que já havia planeado um projeto piloto na empresa. Nesse projeto utilizaram o Scrum e, apesar de reconhecerem utilidade ao Extreme Programming, decidiram não o utilizar nesta primeira experiência. Para avaliar os seus objetivos definiram recolher duas métricas para medir o sucesso do projeto: *time-to-market* e a avaliação da satisfação dos clientes. Os objetivos eram diminuir o *time-to-market* e aumentar a avaliação de satisfação dos clientes.

Para esse primeiro projeto-piloto foi-lhes disponibilizado um espaço, para que pudessem trabalhar num ambiente partilhado. Definiram um *Scrum Master* e estabeleceram que o *Product Owner* devia reunir com a equipa quatro horas por dia. Iniciado o projeto, começaram a deparar-se com vários obstáculos devido sobretudo à mentalidade das metodologias em cascata que estava enraizada nos colaboradores. Estes problemas levaram ao fracasso do projeto no entanto, serviu para retirar algumas lições como o facto de a abordagem errada conduzir a resultados desastrosos. Depois deste desfecho, decidiram que a documentação típica dos modelos em cascata não era a mais adequada e que os membros da equipa tinham de estar focados a tempo inteiro na equipa do projeto.

Simultaneamente ao projeto piloto foi posto em funcionamento um plano de treino ágil. Este treino era facultativo e aberto a todo o pessoal da divisão e era constituído por um conjunto de aulas cujo objetivo era propagar os valores ágeis e capacitar os funcionários. Isto permitiu reduzir a resistência dos funcionários à mudança de metodologia, que advém da redução de papéis formais e, conseqüentemente, de autoridade dentro da organização.

Entretanto as equipas ágeis formadas já tinham um conhecimento mais aprofundado e já conseguiam entregar valor com maior rapidez aos clientes dos seus projetos. Contudo, devido à grande dimensão da divisão, existiam bastantes projetos ágeis a serem realizados simultaneamente. O próximo passo foi aumentar eficiência da gestão desses portefólios de projetos.

Optaram por conjugar a algumas ideias ágeis e *Lean* para conseguir esse melhoramento na gestão dos portefólios de projetos, nomeadamente: (1) cada portefólio criou um *backlog* com os projetos priorizados pelo cliente, (2) atribuíram 2 a 3 equipas ágeis a cada portefólio, (3) criaram um ambiente de teste dedicado para cada equipa do portefólio e (4) estabeleceram um *Product Owner* para cada portefólio. Para além disso criaram um processo de calendarização de lançamentos. A implementação destas ideias funcionou com sucesso para a divisão. Os autores realçam que estas ideias obtiveram sucesso devido ao tipo de projetos existentes, que não necessitavam muitos aprimoramentos.

Depois de mais de quarenta projetos concluídos, os objetivos inicialmente estabelecidos foram cumpridos. O *time-to-market* diminuiu 50% e a avaliação da satisfação dos clientes aumentou para 100%. Apontam como principal causa deste sucesso o aumento do foco em entregar valor ao cliente em cada projeto ao invés de gastar tempo em planejar projetos futuros.

Concluindo, este artigo realça a utilidade que o PMO pode ter no processo de transição para as abordagens ágeis e descreve de que forma isso foi possível, no caso da divisão de sistemas de informação da empresa em questão.

3.1.3 The Lean-Agile PMO

Esta proposta de Augustine e Cuellar (2006) é diferente das apresentadas até aqui porque oferece um conjunto de recomendações sobre como combinar as entregas de projetos ágeis ao nível do projeto com o pensamento *lean* ao nível do portefólio, de modo a aumentar significativamente o fluxo do projeto, o desempenho do investimento financeiro e a satisfação do patrocinador de negócio.

Os autores para definirem o *Lean-Agile PMO* abordaram dois aspetos, designadamente: (1) a sua estrutura organizacional e a sua posição relativa à gestão executiva e às equipas ágeis dos projetos, e (2) os seus princípios operacionais recolhidos sobretudo do pensamento *lean*.

Quanto à estrutura organizacional, esta proposta recomenda que esteja intimamente ligada à gestão executiva que é responsável por determinar os objetivos estratégicos, mas garantido que é autónoma o suficiente para evitar gestão autocrática. Deve também ser colaborativa com as equipas ágeis dos projetos, responsáveis por entregar os objetivos estratégicos. Além disso, também deve ser auto-organizada para se possa adaptar à mudança.

Os pontos proeminentes desta organização estrutural são os seus representantes de ligação, os representantes eleitos e a gestão ágil. Conforme ilustra a Figura 13, este PMO liga-se aos níveis de gestão executiva e às equipas de projeto através dos representantes de ligação. Esses elementos são eleitos pelas equipas de projetos para interagirem com o PMO, que por sua vez elege os seus próprios representantes para interagir com a gestão executiva. Para além disso, o modelo está configurado para usar práticas ágeis ao nível dos projetos, sendo que os autores dão como exemplo as práticas do Scrum.

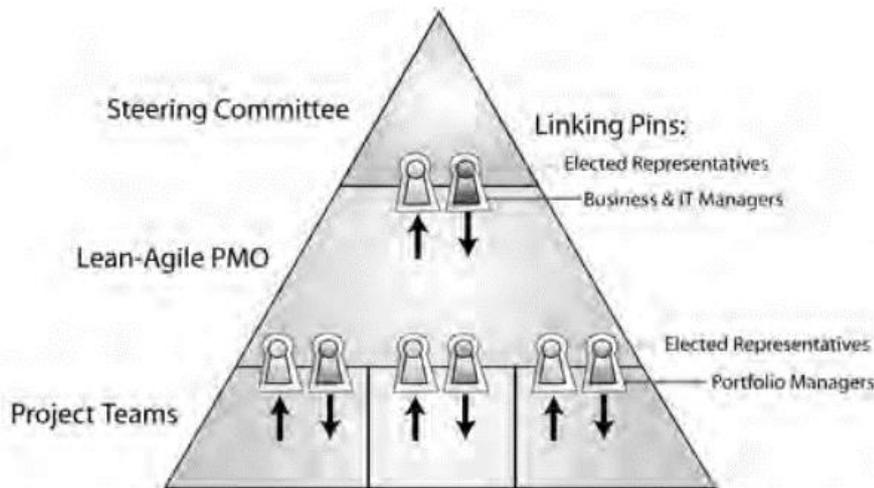


Figura 13- Estrutura organizacional do *Lean-Agile PMO* (retirado de (Augustine & Cuellar, 2006))

Esta estrutura organizacional pretende evitar que se assuma uma gestão tradicional, *top-down* e baseada no comando e controlo, prevenindo potenciais tensões entre o *Lean-Agile PMO* e as equipas ágeis, as quais está encarregue de suportar. A combinação de elementos atribuídos pela gestão executiva com elementos eleitos por parte das equipas foi a forma encontrada para assegurar que o *Lean-Agile PMO* não se torna autocrático.

Quanto aos princípios operacionais, os autores basearam-se no pensamento *lean* e nos métodos ágeis para delinear três princípios cujo objetivo é aumentar a eficácia da gestão de portefólios de projeto. Os princípios operacionais propostos são os seguintes: alinhamento contínuo, gerir o fluxo do projeto e gerir as restrições do sistema. A cada um destes princípios corresponde um conjunto de práticas, enumeradas na Tabela 11.

Tabela 11- Princípios e práticas do *Lean-Agile PMO* (retirado de (Augustine & Cuellar, 2006))

Princípio	Prática
Alinhamento contínuo	<ul style="list-style-type: none"> • Comunicar a intenção estratégica • Idealizar conforme a estratégia • Medir os resultados do negócio • Fazer classificações e seleções abertas e visíveis • Repriorizar regularmente
Gerir o fluxo do projeto	<ul style="list-style-type: none"> • Calendarizar o projeto de forma <i>lean</i> • Reduzir o trabalho em progresso do projeto
Gerir as restrições do sistema	<ul style="list-style-type: none"> • Identificar restrições organizacionais e de processos • Otimizar e elevar as restrições

Resumidamente, esta proposta foca-se sobretudo na resolução de problemas relacionados com a gestão de portefólio de projetos. Mais especificamente, o seu foco é o alinhamento do PMO (cujo foco são as práticas de gestão dos projetos) com a função de gestão de portefólio. Para alcançar este alinhamento é proposto que as práticas ágeis utilizadas ao nível do projeto sejam complementadas com princípios *lean* ao nível do portefólio. Com isto, a proposta visa diminuir o *lead time* e, conseqüentemente aumentar o fluxo do projeto, o que significa um aumento na velocidade com que os requisitos do cliente são concluídos resultando num retorno de valor mais rápido para a organização

3.1.4 Agile Practice Guide

Outra obra que foi explorada com algum detalhe foi o “Agile Practice Guide” do Project Management Institute (2017). Este livro contém uma secção dedicado a considerações organizacionais para a agilidade de projeto, explorando fatores organizacionais que impactam o uso de práticas ágeis, como a cultura, a preparação, as práticas de negócio e, o mais relevante para esta dissertação, o papel do *project management office*.

Como a abordagem ágil cria mudança cultural, a organização pode ter de mudar também, incluindo o PMO. Nesta obra é recomendado que um PMO ágil tenha três características fundamentais: seja conduzido pelo valor, orientado por convite e que seja multidisciplinar.

A primeira característica diz respeito ao objetivo que um PMO ágil deve assumir de entregar o valor correto, à audiência correta, no tempo certo. O PMO ágil deve esforçar-se para entregar o que seja necessário, mantendo-se a par das necessidades dos clientes para garantir que é capaz de se adaptar às suas necessidades, podendo suportar os projetos da forma mais adequada.

A segunda característica é ser orientado ao convite, isto é, o PMO deve convidar apenas os interessados para se engajarem com os serviços que o PMO pretende fornecer. Deve evitar ordenar que determinadas abordagens sejam utilizadas, pelo contrário, deve providenciar um ambiente em que o desejo de engajamento com as práticas do PMO, por parte dos funcionários, seja promovido.

Finalmente, o PMO ágil deve ser multidisciplinar, ou seja, o PMO deve ser capaz e proficiente em várias competências porque projetos diferentes têm requisitos diferentes. No livro são enumerados um conjunto de serviços de alguns PMOs que se transformaram em PMOs ágeis, em conformidade com esta multidisciplinaridade. Nomeadamente, desenvolver e implementar *standards*, desenvolver pessoal através de treino e mentoria, gestão de vários projetos, facilitar aprendizagem organizacional, gerir *stakeholders*, executar tarefas especializadas para os projetos e recrutar e, selecionar e avaliar os líderes de equipa.

3.1.5 Succeeding with Agile

Em "*Succeeding with agile: Software development using Scrum*" Mike Cohn (2010) apresenta um conjunto de considerações sobre os PMO no contexto ágil.

Começa por argumentar que um PMO, na transição dos métodos tradicionais para os ágeis, pode ser uma mais-valia se tiver engajado e suportar essa transição. Caso contrário, o PMO pode revelar-se uma fonte de resistência à mudança, ao tentar defender os processos atuais em vez de os melhorar. Portanto, o autor argumenta que o PMO tem o potencial para ajudar a implementar e a disseminar práticas ágeis por toda a organização.

Neste ambiente de transição, o autor refere que os motivos para os membros de um PMO, dito tradicional, apresentarem-se resistentes a alterações advêm de dois fatores: (1) a mudança é pessoalmente e profissionalmente assustadora e (2) na maior parte da literatura de metodologias ágeis o PMO está ausente.

Tendo em conta estas preocupações e de modo a ameniza-las, o autor explorou o tipo de trabalho desenvolvido por PMOs em organizações que decidiram fazer a transição para o Scrum. O autor explora o trabalho desenvolvido por estes PMOs em três áreas, nomeadamente, pessoas, projetos e processos.

Na área das pessoas, o autor recomenda que o PMO ágil deve seguir um conjunto de práticas, nomeadamente, desenvolver um programa de treino, fornecer *coaching*, selecionar e treinar *coaches*, e desafiar comportamentos existentes.

Na área dos projetos, o autor refere que algumas das responsabilidades de um PMO dito tradicional para um PMO ágil desaparecem, no entanto algumas devem permanecer, nomeadamente, assistir com o *reporting*, assistir com necessidades de conformidade e gerir a entrada de novos projetos.

Quanto à área de processos, o autor enumera um conjunto de atividades a serem desenvolvidas pelo PMO, nomeadamente, fornecer e manter ferramentas, assistir em estabelecer e recolher métricas, reduzir o desperdício, ajudar no estabelecimento e suporte de comunidades de prática, criar uma de consistência apropriada entre as equipas, coordenar as equipas, utilizar o Scrum como forma de trabalhar do próprio PMO e assistir em relações com outros grupos organizacionais ou *stakeholders*.

Finalmente, Mike Cohn constata que vários PMOs optaram por alterar o seu nome para algo que se adegue melhor ao seu novo papel. Acrescenta ainda um conjunto de exemplos desses mesmos nomes, como “Scrum center of Excellence”, “Scrum Competence Center”, “Scrum Office” ou “Development Support”. No entanto, faz uma ressalva no final de que não basta apenas mudar o nome do PMO para que este seja bem-sucedido após a transição dos processos de desenvolvimento sequencial para as abordagens ágeis, é necessário ir muito mais além como demonstrado nas práticas recomendadas pelo autor.

3.1.6 Análise crítica

Os dois primeiros casos de estudo analisados relatam experiências de implementação por conta própria, não seguindo nenhum modelo previamente definido. O terceiro caso de estudo, o *Lean-Agile PMO*, apresenta um modelo fundamentado com valores princípios e práticas que visam resolver problemas da gestão de portefólios de projetos. As duas últimas obras analisadas não apresentam um modelo de PMO ágil nem um relato de experiência de implementação, mas sim um conjunto de práticas e de sugestões que um gabinete dessa natureza deve possuir. O estudo aprofundado desta literatura permitiu retirar algumas conclusões.

Primeiro, permitiu concluir que, conforme indicado na revisão de literatura, a estrutura organizacional influencia a capacidade da organização responder a mudanças e pode ser um

impedimento à colaboração (no caso da funcional). As estruturas devem ser as mais planas possíveis para promover a coordenação entre a gestão dos projetos e a gestão executiva.

Segundo, no que toca à gestão de portefólios, o *Agile PMO* e o *Lean-Agile PMO* sugerem a conjugação de princípios ágeis com princípios *lean* para a gestão de portefólio, sobre a premissa que esta união irá aumentar a eficiência ao nível de portefólio. O *Agile Office* também realça a importância que um portefólio de projetos prioritizados oferece ao processo de tomada de decisão do seu gabinete. Estes casos de estudo permitiram concluir que a gestão de projetos a este nível precisa sobretudo de comunicar mais diretamente com os níveis de gestão abaixo, aplicando para isso princípios *lean* que visam encurtar as vias de comunicação entre os vários níveis de gestão.

Outra evidência é que todos os casos de estudo indicam a importância de comprovar o valor da implementação de um PMO ágil para que a restante organização possa entender a sua mais-valia. É fundamental para garantir a sua continuidade.

Mais um fator comum a todos os casos de estudo foi o facto de todos recomendarem a gestão do PMO ágil como se tratasse de uma equipa ágil servindo como modelo para os projetos sob a sua alçada.

A última conclusão diz respeito às práticas que preconizam. Os dois últimos casos de estudo, como se tratam de livros com sugestões, descrevem as práticas fundamentando a sua importância. O *Lean-Agile PMO* também faz isto ao basear as suas práticas em valores e princípios que se alinham com a mentalidade ágil. Já o *Agile Office* e o *Agile PMO* apenas descrevem as práticas utilizadas nas respetivas experiências relatadas, contudo, as suas práticas são corroboradas pelas práticas dos restantes casos de estudo.

Esta corroboração foi interessante porque permitiu que as práticas provenientes de perspetivas diferentes, culminassem em recomendações semelhantes acerca de quais as funções que um gabinete de gestão de projetos ágeis deve assumir. Esta validação entre autores permitiu que se recolhesse um conjunto de práticas consensuais e valorizadas por vários autores.

Durante esta dissertação verificou-se que a literatura sobre relatos de implementações ou descrições de modelos de PMOs ágeis é ainda muito reduzida. Assim sendo, as comunicações do conhecimento resultante desta dissertação (i.e. o artigo científico e este relatório de dissertação) constituem um contributo para a redução desta lacuna, graças ao novo modelo que descrevem.

Em resumo, da análise destes casos de estudo foi possível entender os potenciais papéis de um PMO ágil, quais as estruturas organizacionais ideais, quais as práticas que devem realizar, como se podem governar e a importância de comprovar a sua necessidade e valor ao resto organização. Este

conhecimento serviu de base para o desenvolvimento do *Agile Coordination Office*, descrito no subcapítulo seguinte.

3.2 Desenvolvimento do modelo

O desenvolvimento do modelo começou com uma revisão de literatura aprofundada, conforme se apresenta no capítulo 2. Depois de adquirir esta base de conhecimento, prosseguiu-se com o levantamento e análise dos casos de estudo existentes, relacionados com a área. Os que mais se assemelhavam ao tema desta dissertação, foram escortinados com maior detalhe como reportado no subcapítulo anterior.

Durante a análise destes trabalhos relacionados foi recolhido um conjunto de práticas. No entanto, algumas dessas práticas encontravam-se representadas com terminologia específica de determinadas metodologias ágeis. Visto que o propósito desta dissertação é obter um modelo para suportar projetos ágeis, independentemente das metodologias utilizadas, abstraímos-nos desses termos e generalizamo-los para outros sem conotação às metodologias. Para além disso, algumas das práticas encontradas coincidiam com práticas de outras obras e, portanto, reduziu-se essa redundância fundindo práticas semelhantes de obras diferentes. O resultado foi um conjunto de 19 práticas recomendadas (ver Tabela 12), únicas entre si, com base nos PMOs ágeis analisados.

Tabela 12- Práticas recolhidas

Práticas	Autores
Apoio no estabelecimento e recolha das métricas	(Cohn, 2010; Tengshe & Noble, 2007)
Assistir a coordenação de equipas	(Cohn, 2010; Project Management Institute, 2017)
Assistir as equipas na interação com outros <i>stakeholders</i>	(Cohn, 2010; Power, 2011)
Criar uma consistência adequada entre equipas	(Cohn, 2010; Power, 2011)
Desafiar o comportamento existente	(Cohn, 2010; Tengshe & Noble, 2007)
Desenvolver e implementar <i>standards</i>	(Cohn, 2010; Project Management Institute, 2017)
Disseminar boas práticas	(Cohn, 2010; Power, 2011)
Facilitar aprendizagem organizacional	(Project Management Institute, 2017)
Fornecer e configurar ferramentas	(Cohn, 2010; Tengshe & Noble, 2007)

Práticas	Autores
Gerir a fluência de novos projetos	(Augustine & Cuellar, 2006; Cohn, 2010)
Gestão de um <i>backlog</i> partilhado	(Cohn, 2010; Power, 2011)
Mentorias e <i>coaching</i> das equipas continuamente	(Cohn, 2010; Power, 2011; Project Management Institute, 2017; Tengshe & Noble, 2007)
Promoção da comunicação entre equipas	(Cohn, 2010; Project Management Institute, 2017)
Promover alinhamento contínuo com a estratégia organizacional	(Augustine & Cuellar, 2006)
Promover e recolher métricas ágeis particulares	(Power, 2011; Tengshe & Noble, 2007)
Reduzir o desperdício	(Augustine & Cuellar, 2006; Cohn, 2010)
Selecionar e priorizar projetos regularmente	(Augustine & Cuellar, 2006; Tengshe & Noble, 2007)
Suportar o estabelecimento de métricas para a gestão do portefólio de projetos	(Augustine & Cuellar, 2006; Cohn, 2010)
Transmissão de conhecimento e lições aprendidas	(Cohn, 2010; Project Management Institute, 2017; Tengshe & Noble, 2007)

Depois de se recolher estas práticas procedeu-se ao seu agrupamento e definiu-se um conjunto de critérios para efetuar essa organização. Importa realçar que a motivação central para proceder com esta divisão era a simplificação do processo de implementação do ACO. Tanto os casos de estudo de PMOs ágeis, como a revisão de literatura sobre os PMOs, revelaram a importância deste processo no reconhecimento de valor por parte da restante organização.

Tendo em conta a motivação supra descrita, esta divisão foi feita com base nos critérios definidos de seguida. O primeiro critério relaciona-se com os principais níveis de gestão normalmente percecionados em diferentes tipos de PMO, nomeadamente, a gestão de projetos, de programas e de portefólios, que assumem, respetivamente papéis ao nível operacional, tático e estratégico (Monteiro et al., 2016). Ou seja, o primeiro critério visava dividir as práticas de acordo com esses níveis de gestão.

O segundo critério foi garantir que esta estrutura se podia ajustar e reajustar às necessidades da organização, evitando demasiada informação em contextos de baixa complexidade de gestão e evitando informação insuficiente em contextos de grande complexidade de gestão. Este critério foi escolhido com

o intuito de garantir flexibilidade ao ACO para se adaptar às mudanças que possam surgir, alinhando-se simultaneamente com o valor ágil de responder à mudança em vez de manter o mesmo plano.

Nas propostas analisadas, as práticas relativas à gestão de portfólios eram devidamente identificadas como tal, permitindo criar uma divisão com o agrupamento destas práticas à qual se deu o nome de “Módulo de Portefólio” ou “MP”. No entanto, esta divisão baseada apenas nos três níveis de gestão revelou-se ineficaz na tarefa de agrupar as restantes práticas recolhidas. Verificou-se que na literatura existiam várias práticas comuns entre o nível da gestão de projetos e de programas, isto é, práticas que tanto se poderiam aplicar ao nível de programa como a alguns casos do nível de projeto. De realçar que, na literatura, essas práticas não eram atribuídas a um nível de gestão em específico.

Este acontecimento levou-nos a alterar a divisão planeada com base nos três níveis de gestão, passando a existir uma nova divisão que agrupava as práticas comuns entre o nível de gestão de programas e o nível de gestão de projetos. A este agrupamento deu-se o nome de “Módulo do Ambiente Comum”, ou “MAC”.

Estando perante a gestão de projetos ágeis, fez-se questão de ter em conta os valores e princípios promovidos pelo manifesto ágil, pelos quais todas as metodologias ágeis se guiam, sendo estas utilizadas pelas equipas de desenvolvimento. Um dos valores ágeis fundamentais é a autogestão feita pelas equipas e, para além de mais, todos os valores e princípios ágeis foram pensados para equipas de pequena dimensão, localizadas no mesmo lugar. Logo, o nível de projeto tinha um problema quanto às práticas que lhe eram atribuídas. O problema consistia na dimensão dos projetos pois, práticas aplicáveis a projetos de grande dimensão, ou com múltiplas equipas eram irrelevantes para o suporte dos pequenos projetos independentes. Contudo, as práticas aplicáveis a projetos de grande dimensão, ou com várias equipas também eram aplicáveis a programas.

Assim sendo, criaram-se dois novos módulos: (1) o “Módulo dos Projetos Relacionados”, ou “MPR”, com as práticas que tanto se aplicavam a projetos de grande escala e/ou com várias equipas, como a programas, e (2) o “Módulo dos Projetos Independentes”, ou MPI, com as práticas aplicáveis apenas aos projetos pequenos e isolados. De realçar que, como o “Módulo do Ambiente Comum” agrupa práticas que são comuns a estes dois módulos, as suas práticas são herdadas pelos mesmos. Isto permite que as práticas do MPI e do MPR só sejam sugeridas para casos em que sejam pertinentes, cumprindo assim o segundo critério de divisão.

O resultado de todo deste processo de desenvolvimento foi o modelo encontra-se explicado detalhadamente no subcapítulo seguinte.

Para além disso, tendo em conta que a principal motivação desta estruturação era facilitar o processo de implementação do ACO, após este estar desenvolvido, explorou-se um conjunto de cenários para demonstrar de que forma o ACO lhes pode dar resposta. O resultado desta exploração encontra-se detalhado no subcapítulo 3.4.

Este modelo inicial foi apresentado à comunidade interessada sob a forma de um artigo que foi submetido e aceite numa conferência de referência na gestão de projetos (Pinto & Ribeiro, 2018). A descrição que se segue apresenta a versão aceite para essa mesma conferência, já com as sugestões, dos revisores, incorporadas.

3.3 Caracterização inicial do modelo

Antes de mais importa realçar os pressupostos assumidos para este modelo. Primeiro, como verificado na revisão de literatura, nem todas as organizações se encontram em contextos adequados para a adoção da mentalidade ágil (Boehm & Turner, 2003), portanto cabe a cada organização fazer esta introspeção e avaliação para identificar as suas reais necessidades.

Segundo, o ACO beneficia mais em ser implementado numa estrutura organizacional pouco hierarquizada, que dê poder às equipas multifuncionais para se autogerirem, do que numa estrutura organizacional muito hierarquizada pois, no primeiro cenário tem as condições ideais para desempenhar o seu papel de suporte e no segundo cenário a sua atividade é limitada pela partilha de responsabilidade inerente aos silos funcionais.

E terceiro, como descrito na finalidade desta dissertação, o propósito deste modelo é suportar apenas projetos de natureza ágil, portanto, neste trabalho não se inclui práticas exclusivas da gestão de projetos tradicionais visto que essas se encontram devidamente exploradas nas diversas tipologias de PMOs.

Passando para a descrição, o ACO é dividido em dois submodelos, o ACO básico e o ACO avançado, sendo compostos por três e um módulo, respetivamente. A principal motivação para criar estes vários módulos foi reduzir a complexidade do processo de implementação do ACO. Dependendo da realidade da organização que decide adotar o ACO, os diferentes módulos podem ser combinados de várias formas para se adequarem o melhor possível à sua complexidade. Isto é importante para evitar um excesso de informação para empresas de menor dimensão, permitindo que o ACO seja implementado gradualmente e/ou rearranjado ao longo do tempo, de acordo com as necessidades em constante mudança da

organização. Tendo em conta esta motivação e os critérios de divisão explicados no subcapítulo anterior, chegou-se finalmente ao modelo inicial do ACO.

A Figura 14 representa uma visão interna do ACO. As relações que os seus módulos têm entre si e com entidades externas são representadas por linhas tracejadas.

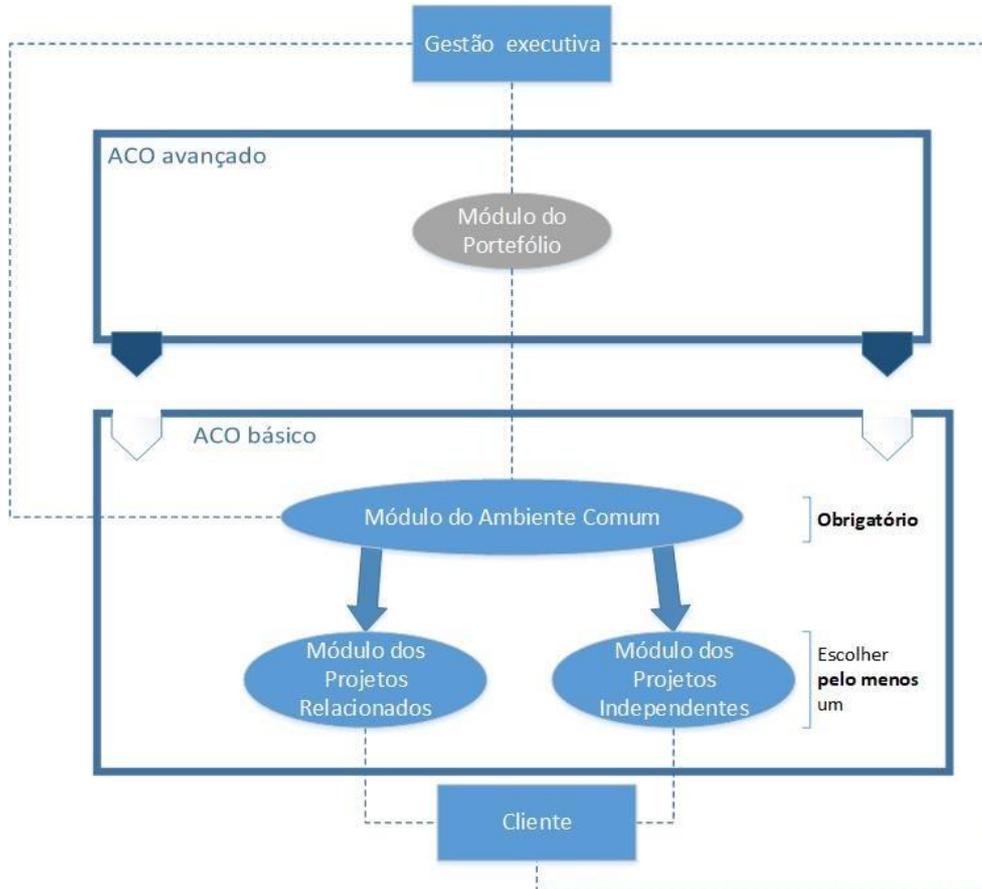


Figura 14- Estrutura inicial do ACO

Resumindo, os módulos criados foram os seguintes: “Módulo do Ambiente Comum” (MAC) para acomodar as práticas comuns entre níveis de gestão, “Módulo de Projetos Independentes” (MPI) para suportar projetos não relacionados, independentes entre si, e o “Módulo de Projetos Relacionados” (MPR) para suportar programas de projetos e/ou grande projetos com várias equipas. Estes três módulos constituem o ACO básico, realçando que tudo que é promovido pelo “Módulo do Ambiente Comum” é herdado pelos dois restantes módulos (conforme indicam as setas azuis do MAC para o MPR e MPI, na Figura 14). O ACO avançado é focado unicamente no suporte da gestão de portefólios e é constituído por apenas um módulo (“Módulo do Portefólio” - MP), cujas práticas se baseiam nesse mesmo foco.

Tendo isto em conta, distribuiu-se as práticas recolhidas (enumeradas na Tabela 12) pelos módulos do ACO a que melhor se adequavam e definiu-se um conjunto de papéis responsáveis pela aplicação das mesmas (ver Figura 15).

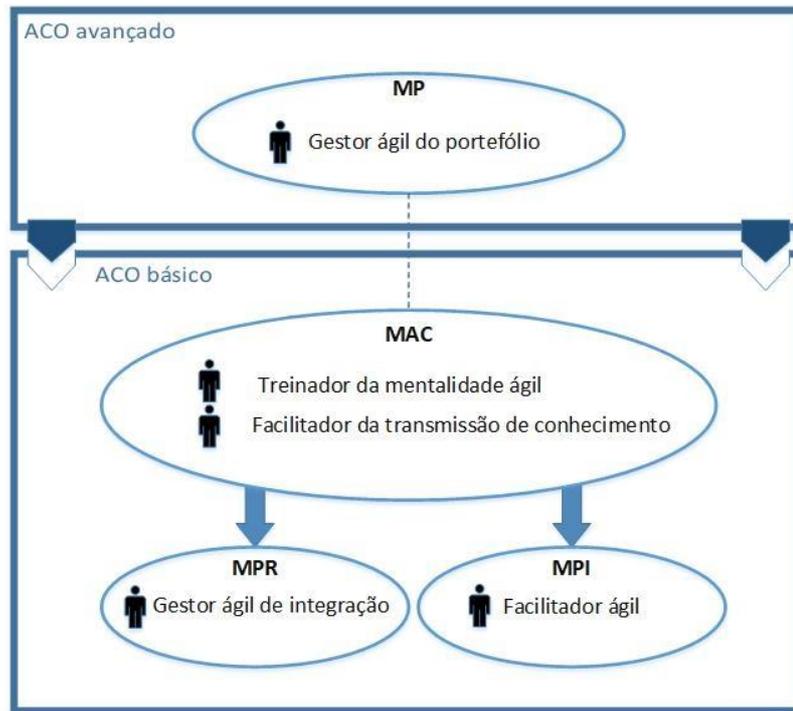


Figura 15- Vista dos papéis do ACO

Na Figura 15, o tracejado representa a interação entre os papéis do ACO avançado e do ACO básico. O “gestor ágil de portfólio” precisa de receber *feedback* contínuo dos projetos e programas para otimizar o valor do portfólio e, simultaneamente, deve-lhes fornecer *feedback* contínuo para os habilitar a tomar decisões em concordância com os objetivos estratégicos.

Assim como acontece com as práticas, também os papéis do MAC são herdados pelo MPR e pelo MPI (representado pelas setas azuis que partem do MAC para o MPR e para o MPI). Isto implica que o “gestor ágil do portfólio” tem linhas de comunicação com todos os papéis do ACO básico e vice-versa.

De seguida são descritos detalhadamente os dois submodelos.

3.3.1 ACO básico

Este submodelo é tripartido nos módulos que servem como base ao processo de implementação do ACO. Considera-se este submodelo como possuidor das fundações do ACO porque agrupa todos os serviços mínimos que o ACO deve providenciar.

O “Módulo do Ambiente Comum” é obrigatório tendo em conta que agrupa tudo o que existe em comum entre os outros dois módulos do ACO básico. Para além disso, pelo menos um dos outros dois módulos tem de ser escolhido porque o “módulo do ambiente comum” por si só não constituiu um suporte apropriado, funciona mais como o primeiro passo do processo de implementação.

No “Módulo de Projetos Independentes” o objetivo principal é providenciar um grupo de práticas, papéis e outros aspetos relacionados com o suporte dos gestores de projetos a respetivos membros de equipa (i.e. ao nível de gestão de projetos de pequenos projetos independentes. As práticas deste módulo seriam as seguintes:

- Promover e recolher métricas ágeis particulares: Projetos independentes devem recolher métricas baseadas em escalas diferentes, para evitar comparações diretas entre projetos não-relacionados, visto que métricas como a velocidade são fortemente influenciadas pelo contexto e natureza do próprio projeto.
- Disseminar boas práticas: Assegurar que as pessoas corretas comunicam entre si. Como os projetos não são relacionados, o ACO deve assumir a responsabilidade de capturar e disseminar boas práticas pelas várias equipas. Isto pode ser conseguido através de *coaches* que simultaneamente ajudam as várias equipas, ou que vão alternando entre equipas, espalhando assim as boas práticas que vão recolhendo das equipas com quem trabalham.

Para aplicar estas práticas recomenda-se que exista o papel de um “Facilitador Ágil”. Este papel pode ser assumido por qualquer individuo mesmo com outro papel dentro dos projetos, visto que se tratam de projetos pequenos e muitas vezes os recursos humanos são limitados. No entanto, também pode ser assumido por um elemento externo às equipas de desenvolvimento, como é o caso do *coach* referido anteriormente, promovendo um acompanhamento mais sistemático e contínuo do que com um elemento que partilha responsabilidades.

No “Módulo de Projetos Relacionados” encontram-se as responsabilidades que o ACO deve ter para assegurar a partilha de conhecimento e experiências entre as equipas relacionadas, para suportar a alocação e realocação de recursos de acordo com os objetivos estratégicos da organização e para promover a coordenação entre equipas. Neste módulo as práticas recomendadas são as seguintes:

- Promoção da comunicação entre equipas: Criar/promover canais de comunicação entre equipas de modo a que se possam coordenar eficientemente. Dado que as metodologias ágeis são baseadas em comunicação informal, a coordenação baseada em *feedback* rápido é primordial para alcançar coordenação apropriada entre equipas (Dingsøyr et al., 2017).
- Assistir a coordenação de equipas: Promover reuniões para que as equipas possam partilhar o seu progresso, experiências e impedimentos entre si. Fundamental para manter as equipas interrelacionadas a funcionarem a ritmos semelhantes. Algumas metodologias ágeis já têm

técnicas para lidar com esta situação (e.g. *Scrum-of-Scrums*) que podem ser providenciadas pelo ACO.

- Gestão de um *backlog* partilhado: Ao nível da gestão de programas ou no caso dos grandes projetos com várias equipas, é comum a criação de um *backlog* para que as equipas possam coordenar o trabalho entre si. Assim sendo, o ACO pode guiar essas equipas através da atribuição do trabalho mais apropriado para cada equipa, para realizar na iteração seguinte. Esta divisão e consequente atribuição do trabalho já é abordada pelas chamadas *frameworks* ágeis de larga escala (duas das mais conhecidas são a *Scaled Agile Framework* e o *Scrum-of-Scrums* (Versionone.com, 2018)), às quais o ACO poderá recorrer. Para além disto, o ACO recolhe e analisa as métricas das diferentes equipas, ocupando uma posição privilegiada para identificar quando duas equipas começam a divergir ou a sobrepor-se.
- Criar uma consistência adequada entre equipas: A melhor maneira para alcançar consistência entre equipas é através de uma aceitação geral entre elas de que uma determinada prática é de facto útil. Realçando que o objetivo não é forçar práticas comuns a todas as equipas, o que se deseja é que as equipas consigam convergir utilizando métodos semelhantes entre si, para facilitar a sua coordenação. Esta prática está intimamente relacionada com a partilha de conhecimento entre equipas e é crucial para garantir a coordenação contínua exigida, especialmente, às equipas intimamente relacionadas, que dependem umas das outras para que o projeto possa continuar.

No caso deste módulo, o que se recomenda é que esteja presente um papel denominado “Gestor ágil de integração”. Este papel poderá ser assumido apenas por uma pessoa a tempo inteiro, no entanto, este módulo lida com múltiplas equipas que comunicam e trabalham umas com as outras constantemente e por isso uma equipa de pessoas (no mínimo duas) será o mais apropriado para assumir este papel.

O “Módulo de Ambiente Comum” foi concebido com o intuito de agrupar as práticas transversais aos outros dois módulos do ACO básico. Este módulo é obrigatório em qualquer implementação do ACO que seja feita. As seguintes práticas são recomendadas para este módulo:

- Mentorias e *coaching* às equipas continuamente: Desenvolver todo o pessoal envolvido através de treino e mentorias. Isto pode ser conseguido através de iniciativas de treino e *coaching* para temas específicos que se pretendam difundir pelas equipas. O ACO deve promover estes eventos para catalisar o processo de adoção da mentalidade ágil, assim como a sua manutenção e

contínua atualização. Num estágio mais avançado deste processo e adoção, o ACO pode educar e treinar os seus próprios *coaches*.

- Transmissão de conhecimento e lições aprendidas: Transversalmente dentro de uma organização, os indivíduos encontram impedimentos, procuram e encontram soluções e adquirem conhecimento que pode ser útil para outros. Quer estejam na mesma equipa ou não, o ACO deve assegurar que esta partilha acontece. Suportar e promover comunidades de prática e criar seminários abertos a todos os funcionários, são dois exemplos de ferramentas que podem ser utilizadas neste âmbito.
- Facilitar aprendizagem organizacional: Esta prática está relacionada com o monitoramento do próprio projeto/programa. As métricas devem ser pensadas e estabelecidas para que possam ser seguidas continuamente, possibilitando um monitoramento constante.
- Assistir as equipas na interação com outros *stakeholders*: Providenciar suporte às equipas que trabalham com outros envolvidos no projeto, internos ou externos à empresa, como os recursos humanos ou clientes, respetivamente. O ACO pode treinar membros das equipas que lidam com estas interações (e.g. *Product Owners*) ajudando-os a articular e a comunicar melhor a sua visão, e a decifrar e compreender a visão dos outros (e.g. criar *User Stories*).
- Fornecer e configurar ferramentas: O ACO não deve ser o responsável por decidir quais as ferramentas que vão ser usadas, mas sim suportar os processos de aquisição e configuração de ferramentas escolhidas pelas próprias equipas de desenvolvimento.
- Desafiar o comportamento existente: É importante que o ACO esteja constantemente a avaliar a mentalidade ágil das equipas, procurando equipas ou indivíduos que estejam a regressar a hábitos antigos, que os estão a impedir de aplicar os valores e princípios ágeis na sua plenitude. Esta prática é de extrema importância em organizações iniciantes com abordagens ágeis devido à potencial resistência à mudança consequente de uma alteração paradigmática desta natureza.
- Desenvolver e implementar *standards*: Fornecer *templates* e ajudar a estabelecer qualquer tipo de *standard* (e.g. escolher qual a metodologia ágil ou *framework* para um projeto com várias equipas). Também inclui assistir as equipas relativamente às necessidades de conformidade, instruindo-as de quais as conformidades inerentes a determinados campos de desenvolvimento (i.e. regulamentação, segurança de dados, privacidade etc.).
- Apoio no estabelecimento e recolha das métricas: Assistir a organização no estabelecimento e recolha de métricas fim-a-fim, que possam ser usadas para apoiar a tomada de decisão da

gestão e que, ao mesmo tempo, não constituam um fardo para as equipas de desenvolvimento. Exemplos disto são métricas como o *time-to-market* ou a taxa de valor entregue.

- Reduzir o desperdício: Todas as atividades e artefactos desperdiçadores dos processos das equipas devem ser eliminados. Isto é sobretudo válido em organizações que recentemente transitaram do tradicional para o ágil, sendo necessário avaliar os processos pré-existentes e seleccionar quais de facto acrescentam valor. O ACO deve evitar introduzir algo que não seja absolutamente necessário e ajudar as equipas a rever regularmente os seus processos, potencializando a sua capacidade de identificar quais desses processos não adicionam valor ao negócio, promovendo assim o a melhoria contínua.

Este módulo agrupa sobretudo práticas dedicadas ao suporte das equipas e da manutenção da boa forma da mentalidade ágil. Assim sendo, recomenda-se dois papéis associados à implementação deste módulo. O primeiro designa-se por “Treinador da mentalidade ágil” e engloba todos os responsáveis pela transmissão e treino de valores ágeis, incluindo mentores, *coaches* e qualquer outro tipo de profissional cuja função seja a promoção destes valores. Este papel deve ser assumido por elementos com muita experiência nas abordagens ágeis mas, caso a organização não tenha nenhum, deve-se inicialmente recorrer a consultores externos experientes.

O outro papel é designado por “Facilitador da transmissão de conhecimento” e a complexidade da empresa determina qual o formato mais apropriado para este cargo. Organizações com projetos de larga escala ou programas, requerem que a transmissão de conhecimento seja assegurada continuamente entre um vasto leque de equipas e indivíduos logo, a quantidade de facilitadores da transmissão de conhecimento deve adequar-se ao nível de coordenação exigido. Contudo projetos independentes de menor escala também beneficiam deste tipo de serviços, só que numa escala menor, podendo neste contexto fundir-se os dois papéis propostos neste módulo.

3.3.2 ACO avançado

Neste submodelo existe apenas um módulo, o “Módulo do Portfolio”. Este é direccionado para organizações de larga escala que necessitam de operar a um nível estratégico. As práticas mencionadas neste módulo também foram recolhidas da literatura sobre PMOs ágeis, no entanto algumas são muito similares àquelas encontradas nos PMOs tradicionais, devido ao baixo impacto que os processos ágeis têm neste nível de gestão, visto que as metodologias ágeis se focam primariamente no nível de projeto. As práticas propostas por este módulo são as seguintes:

- Selecionar e priorizar projetos regularmente: Mudanças na estratégia organizacional ou nas prioridades dos clientes devem ser refletidas, rapidamente, nas prioridades dos portfólios sendo isto possível devido aos processos adaptativos e iterativos característicos das abordagens ágeis. O ACO deve recolher e analisar as métricas que medem o desempenho dos projetos para fazer a alocação de recursos consoante as prioridades da organização.
- Gerir a afluência de novos projetos: O ACO deve apoiar a organização no processo de admissão de novos projetos para serem desenvolvidos. Portanto, o ACO pode limitar a quantidade de trabalho a ser iniciado, evitando a tentação de começar demasiados projetos em simultâneo. Para conseguir isto, o ACO deve realizar atividades como a avaliação e seleção dos novos projetos tendo em consideração a capacidade das equipas de desenvolvimento. Um sistema de ranking é fundamental para avaliar e comparar os projetos.
- Promover alinhamento contínuo com a estratégia organizacional: Os objetivos estratégicos podem ser alterados ao longo do curso de um projeto, e assim sendo, é necessário reformular as métricas utilizadas para refletir os novos objetivos. Este alinhamento contínuo pode ser feito através da comunicação da intenção estratégica a todos os que são afetados diretamente por ela. O processo de ranking e seleção dos projetos deve ser visível a todos os interessados, para transmitir claramente quais as prioridades da organização.
- Suportar o estabelecimento de métricas para a gestão do portefólio de projetos: Como mencionado na prática anterior, é necessário estabelecer métricas que avaliem o desempenho dos projetos à luz dos objetivos estratégicos. A gestão executiva fornece os objetivos estratégicos e o ACO deve idealizar e estabelecer as métricas que melhor os possam medir. Devem garantir que medem de forma confiável o desempenho dos projetos e programas sem constituir um impedimento para as abordagens ágeis adotadas.

Para aplicar estas práticas recomenda-se um papel designado por “Gestor ágil do portefólio”. Dependendo do tamanho do portefólio, este papel deve ser assumido por uma equipa com uma dimensão proporcional à dimensão do portefólio. O *feedback* contínuo recebido do ACO básico é fundamental para que esta função possa ter sucesso. O mesmo se aplica em sentido inverso, o *feedback* contínuo enviado por este(s) colaborador(es) para o outro submodelo é fundamental para garantir que os colaboradores do ACO básico tenham capacidade de se alinhar com os objetivos estratégicos. Esta transmissão e receção de *feedback* está na base de todas as práticas do MP.

Assim se finaliza a descrição dos dois submodelos do ACO. De seguida são descritos vários cenários de implementação e respetivas combinações de submodelos e módulos do ACO que devem ser escolhidas.

3.4 Implementação do modelo

Primeiramente importa realçar que as práticas recomendadas no ACO não são obrigatórias, não precisam de ser seguidas e aplicadas rigidamente. São somente recomendações e como tal, apenas as práticas que preencham as necessidades da organização devem ser utilizadas. As organizações devem passar por um processo de introspeção antes de iniciar a implementação do ACO, para adquirir uma noção realista acerca das suas necessidades e das suas peculiaridades.

Existem várias maneiras de implementar o ACO. Para cobrir o máximo de casos possíveis, ponderou-se um conjunto de cenários. Para cada cenário é proposta uma configuração do ACO constituída pelos vários módulos que se recomendam que sejam escolhidos (ver Tabela 13).

Tabela 13- Modos de implementação do ACO

Cenário	Combinação de módulos recomendada
Projetos grandes, com várias equipas, programas ou qualquer grupo de projetos relacionados	ACO básico (MAC + MPR)
Projetos independentes	ACO básico (MAC + MPI)
Portefólio de projetos relacionados e programas	ACO básico (MAC + MPR) + ACO avançado (MP)
Portefólio de projetos independentes	ACO básico (MAC + MPI) + ACO avançado (MP)
Portefólio misto	ACO básico (MAC + MPI + MPR) + ACO avançado (MP)

Esta proposta visa padronizar boas práticas que suportem a gestão de projetos, programas e portefólios envolvidos no ambiente ágil, sem ser demasiado prescritivo. E para o implementar pode-se até optar por uma abordagem comum na literatura, que consiste no próprio processo de implementação de um PMO ágil ser visto como um projeto ágil com um *backlog* de ações prioritizadas e com implementações graduais (Alsadeq, 2011; Cohn, 2010; Sliger, 2007).

Conforme ilustra a Tabela 14, a implementação do ACO permite responder a grande parte dos potenciais princípios de desenvolvimento ágil de larga escala (Dingsøyr & Moe, 2014), descritos no subcapítulo 2.6.

Tabela 14- Princípios de desenvolvimento ágil de larga escala refletidos no modelo

Princípios de desenvolvimento ágil de larga escala	Como se refletem no modelo
Arquitetura que facilita a coordenação do trabalho.	Estrutura do ACO
Arquitetura adaptável ao nível de incerteza.	
Normas e valores comuns facilitam coordenação.	Práticas e papéis do ACO
Redes eficazes de para transmitir conhecimento	
<i>Feedback</i> contínuo dos projetos para o portfólio.	
<i>Feedback</i> contínuo dos portfólios para o projetos.	Pressupostos do ACO
Descrever o contexto para a agilidade	

Isto permite que o ACO possa justificar a sua importância, demonstrando o seu valor perante o resto da organização ao resolver as problemáticas em que se baseiam estes princípios. Estes princípios refletem-se nas várias componentes do ACO, nomeadamente a sua estrutura, as práticas e papéis sugeridos e os pressupostos em que assenta.

Como já foi referido, esta é a descrição do modelo presente no artigo que foi avaliado e aceite pela conferência de gestão de projetos. Depois de conceber e avaliar o modelo através desse artigo, voltou-se a avaliá-lo através da opinião de profissionais experientes na área. O capítulo seguinte descreve todo esse processo de avaliação e o modelo refinado que dele resultou.

4. AVALIAÇÃO DO MODELO

Conforme descrito nas etapas planeadas para esta dissertação, o modelo desenvolvido passou por dois momentos de avaliação.

Inicialmente o modelo foi avaliado através da sua submissão, na forma de um artigo científico, para uma conferência especializada na área da gestão de projetos. Foi apreciado por quatro revisores da área em questão, que contribuíram com os seus comentários e sugestões para o melhoramento do modelo, o qual está descrito no capítulo anterior. O artigo depois de modificado foi aceite como *full paper* contudo, o artigo por si só não constitui um método replicável de avaliação do modelo do ACO, pelo que esta avaliação serviu sobretudo para o aprimoramento do modelo.

Assim sendo e porque se queria receber uma segunda perspetiva, agora do ponto de vista dos praticantes, foi consultado um conjunto de gestores especializados na área de gestão de projetos, sobretudo com experiência em PMOs encarregues por projetos ágeis. Esta consulta foi feita através de entrevistas, cujo principal objetivo era receber o *feedback* dos profissionais sobre um conjunto de temas para assim avaliar a utilidade do modelo desenvolvido.

O processo de desenvolvimento das entrevistas encontra-se explicado pormenorizadamente no subcapítulo 4.1, assim como os procedimentos estabelecidos para seguir durante as entrevistas e o perfil dos entrevistados.

Para além disso, neste capítulo são também analisados os comentários ao modelo inicial do ACO, nomeadamente, as melhorias e correções recomendadas pelos entrevistados, e a corroboração e validação do ACO e seus pressupostos por parte dos mesmos.

Este capítulo é concluído com a descrição das alterações ao modelo refinado do ACO, já com as alterações resultantes das recomendações feitas pelos entrevistados.

4.1 Entrevistas

Com a proposta inicial do ACO já desenvolvida e refinada com base no contributo dos revisores do artigo, prosseguiu-se para a última etapa de recolha de *feedback*. O objetivo desta atividade foi obter o parecer de profissionais experientes e ativos em PMOs, onde uma das funções fosse gerir e/ou suportar projetos ágeis. Este parecer foi utilizado para aprimorar o modelo do ACO e para validar a sua utilidade na perspetiva dos praticantes.

Para este efeito, decidiu-se efetuar entrevistas porque é uma técnica apropriada para investigações cujos objetivos de pesquisa sejam de cariz qualitativo (Hove & Anda, 2005), sendo esse o caso da presente investigação. Trata-se portanto de uma recolha de dados qualitativos, no entanto existem vários tipos de entrevistas, nomeadamente, estruturadas, não-estruturadas ou semiestruturadas, e entrevistas de grupo (Fontana & Frey, 2000).

O primeiro tipo segue rigidamente uma estrutura previamente estabelecida, ideal para casos em que as entrevistas sejam conduzidas por alguém que não é o investigador da pesquisa. O segundo tipo segue um roteiro incompleto que funciona como espinha dorsal da entrevista mas que oferece espaço para aprofundar ou mudar o rumo da entrevista de acordo com o contributo do entrevistado. O terceiro tipo são as entrevistas de grupo em que duas ou mais pessoas são entrevistadas em simultâneo.

Optou-se pela entrevista semiestruturada sobretudo devido à flexibilidade que oferece ao entrevistado para elaborar as suas respostas, permitindo captar detalhadamente as suas perspetivas e opiniões. Isto é conseguido devido à combinação de questões específicas e questões abertas, sendo as primeiras para recolher a informação prevista e as segundas para obter tipos inesperados de informações (Hove & Anda, 2005).

Este tipo de entrevista também se justifica porque permite ao entrevistador ajustar o rumo da entrevista à medida que esta decorre, não se restringindo a um conjunto de questões fechadas previamente planeadas, como aconteceria com entrevistas estruturadas. No contexto desta atividade de validação, o objetivo era captar o ponto de vista profissional acerca do modelo conceptual desenvolvido e, assim sendo, era fundamental que o entrevistado pudesse expressar-se totalmente e que tivesse a hipótese de mudar o rumo da entrevista, podendo demonstrar novas perspetivas que até então poderiam não ter sido consideradas.

Para além disso, as entrevistas semiestruturadas revelam-se bastante úteis quando só existe possibilidade de uma única entrevista pois permitem que o entrevistado tenha a oportunidade de abordar todos os assuntos que achar relevantes (Cohen & Crabtree, 2006). E este foi o caso desta atividade, a janela de oportunidade para realizar cada entrevista era bastante reduzida.

A utilidade dos dados recolhidos durante a entrevista dependem muito da forma como esta é planeada e conduzida (Hove & Anda, 2005). Para avaliar os resultados obtidos é importante que pelo menos um determinado conjunto de informações seja reportado, nomeadamente, o número de entrevistados e qual o perfil utilizado para a sua seleção, descrição da própria entrevista (como duração, localização, quantidade de entrevistas feitas com cada entrevistado), número de entrevistadores e quais

os seus papéis na pesquisa, e a documentação utilizada durante a entrevista (como o guião da entrevista e artefactos da pesquisa) (Hove & Anda, 2005).

Visto tratar-se de uma entrevista semiestruturada, no seu planeamento foram idealizadas e concebidas algumas questões para guiarem a entrevista e para garantir que determinados assuntos são abordados mas, ao mesmo tempo, permitindo que o entrevistador/pesquisador pudesse seguir assuntos ou linhas de raciocínio levantadas pelo entrevistado. Com base nessas questões foi desenvolvido um guião para a entrevista (ver Anexo I – Guião da entrevista).

4.1.1 Desenvolvimento das entrevistas

A entrevista divide-se em três partes, refletidas nas três caixas de texto do guião da entrevista. A primeira parte consiste na identificação do perfil profissional do entrevistado e está representado no guião como o quadro da “Identificação do entrevistado”. A segunda parte consiste na caracterização do cargo que ocupa atualmente e está representado no guião no quadro “Dados sobre a organização”. Por fim, a terceira parte consiste no conjunto de questões a serem colocadas ao entrevistado, representado no guião no quadro das “Questões para o entrevistado”. Devido à flexibilidade inerente às entrevistas semiestruturadas, reservou-se um espaço para a recolha de observações para além do âmbito inicialmente planeado, representado no guião como a caixa de texto das “Observações”.

Na primeira parte são colocadas duas questões diretas acerca da experiência profissional do entrevistado. Na primeira pergunta foi questionado qual o cargo que ocupa atualmente. Na segunda pergunta foi questionado qual a experiência anterior na área que está a ser debatida. O objetivo destas questões é recolher dados sobre o background profissional dos entrevistados para depois poder comparar as respostas dadas na parte 3 da entrevista.

Na segunda parte são colocadas seis questões relativas à estrutura organizacional em que se encontra envolvido atualmente. Na primeira pergunta é pedida uma estimativa de quantos projetos é que estão sobre a alçada do seu PMO. Na pergunta seguinte é questionado a dimensão das equipas de desenvolvimento. Nesta questão optou-se por especificar o tipo de equipa porque o âmbito desta dissertação são as empresas de desenvolvimento de *software*, mais especificamente de TSI. Na terceira pergunta foi questionado qual a natureza dos projetos geridos. Nesta questão o objetivo era entender se todos os projetos sob a alçada do seu PMO eram, de facto, de desenvolvimento de *software*. Na quarta pergunta foi questionado quais as abordagens ágeis utilizadas pelas equipas de desenvolvimento. Na quinta pergunta foram questionadas ferramentas utilizadas pela equipa de desenvolvimento para

aplicarem a mentalidade ágil. Por fim, na sexta pergunta foi questionado quais as métricas que estabelecem e recolhem para monitorizar das equipas de desenvolvimento.

Na terceira parte surgem as questões orientadoras da entrevista. Estas questões dividem-se em três categorias:

- Perguntas relacionadas com a perceção que o entrevistado tem do contexto atual da relação que os PMOs tem com os projetos ágeis e de que forma o ACO poderia ser introduzido ou não neste contexto. Nesta categoria enquadram-se as perguntas 1 e 2 do guião, respetivamente, “Quais as lacunas de um PMO quando os seus projetos adotam o *agile*? Quais as mais-valias?” e “As responsabilidades do PMO alteram-se perante projetos ágeis relativamente aos projetos tradicionais? O nível de controlo sofre alterações?”.
- Perguntas relacionadas com a experiência do entrevistado em lidar com projetos ágeis. Nesta categoria enquadram-se as perguntas 3,4 e 5, respetivamente, “Dada a sua experiência, quais as principais diferenças em relação ao modo como as práticas são executadas nestes diferentes contextos? O que muda entre o contexto de tradicional e ágil? ”, “Na sua experiência profissional como lida com a gestão de vários projetos cujas métricas são de natureza ágil?” e “Já escalou algum processo ágil? Ou recorreu a alguma *framework* para escalar o *agile* na organização? Se sim, quais as mais-valias que poderia acrescentar a um ACO? Utilizou alguma *framework* para fazer a gestão de programas/portefólios? (NEXUS, SAFe, etc.) E algum método como o Scrum-of-Scrums?”.
- Perguntas relacionadas com o modelo do ACO em si. Nesta categoria enquadram-se as perguntas 6,7,8 e 9, respetivamente “. Considera que esta divisão é útil para simplificar o processo de implementação e o reajustamento do ACO? Que outras práticas recomendaria para o ACO?”, “Concorda com este papel de suporte do ACO”, “Que vantagens podia trazer um ACO no contexto atual de trabalho? E desafios?” e, finalmente, “Tem mais alguma sugestão?”

Como já foi referido, estas perguntas eram de resposta aberta, permitindo que o entrevistado pudesse encadear noutra tema que, se se verificasse útil para o âmbito da entrevista, era continuado.

4.1.2 Procedimentos para as entrevistas

Para que as entrevistas fossem o mais homogéneas possíveis, foi estabelecido um procedimento a seguir. Obviamente que, como se tratam de entrevistas semiestruturadas, o decorrer das entrevistas não foi totalmente idêntico, devido à abertura que é dada ao entrevistado para se extraviar do guião.

Uma apresentação de PowerPoint com o resumo da proposta inicial do ACO, assim como o artigo, produzido anteriormente, com a explicação formal da proposta, foram enviados antecipadamente para todos os entrevistados.

Estes artefactos foram também utilizados para introduzir e contextualizar os entrevistados com o tema das entrevista e para auxiliar os mesmos durante a entrevista. O conhecimento dos entrevistados por vezes revela-se tácito e difícil de explicar e estes artefactos auxiliam no esclarecimento de ideias, tornam mais fácil que o entrevistado faça boas perguntas de seguimento e possibilita que os entrevistados providenciem informações valiosas e elucidativas com base nos artigos que têm à disposição (Hove & Anda, 2005).

Estando o entrevistado já devidamente contextualizado, prosseguiu-se para a entrevista propriamente dita, iniciando-se com um pedido de permissão ao entrevistado para que pudesse ser gravado o áudio da conversa.

O gravador de áudio foi utilizado para manter um registo completo da entrevista para, mais tarde, a análise poder ser feita com base em dados precisos do que foi dito (Hove & Anda, 2005). As vantagens e desvantagens da utilização desta ferramenta são várias mas, de grosso modo, resumem-se ao equilíbrio entre ter o material necessário para transcrever com precisão o que foi dito e permitir ao entrevistador estar concentrado no que está a ser dito e não em tirar notas, e entre o entrevistado se sentir incomodado por estar a ser gravado. No entanto nenhum dos entrevistados se opôs ou apresentou algum tipo de preocupação.

A entrevista prosseguiu de acordo com o guião da entrevista e foi concluída com um agradecimento pela disponibilidade e pelo contributo.

4.1.3 Perfil dos entrevistados

Antes de se proceder com a entrevista, traçou-se um perfil relativo ao tipo de entrevistado que mais poderia contribuir para este momento de avaliação.

Definiu-se como requisito obrigatório que se tratassem de profissionais responsáveis por PMO ou estruturas semelhantes encarregues de gerir e coordenar projetos ágeis, visto que é esse o contexto em que o ACO está inserido. Outro requisito prioritário consistiu em garantir que os PMOs aos quais os entrevistados pertenciam tinham de ter sob sua alçada projetos de desenvolvimento de tecnologia, para garantir que estão em consonância com a proposta que foi pensada para as empresas de TSI. Como requisitos secundários, definiu-se que os entrevistados tivessem proveniência de áreas de *software*, que

os projetos que suportavam fossem de desenvolvimento de *software* e que tivessem um papel ativo na coordenação desses projetos.

Tendo em conta estes critérios foram selecionados três candidatos que satisfaziam os requisitos mencionados. Os três candidatos selecionados tinham, atualmente, um papel ativo num PMO, embora em diferentes níveis. Todos os candidatos estavam também inseridos em PMOs responsáveis por projetos ágeis, no entanto, a proporção de projetos ágeis em relação aos tradicionais era bastante reduzida. Alguns cumpriram os requisitos secundários, nomeadamente, terem um background apurado em áreas de *software*. As informações detalhadas de cada entrevistado podem ser consultadas na Tabela 15.

Tabela 15- Perfis dos entrevistados A,B e C

		Entrevistado A	Entrevistado B	Entrevistado C
Dados biográficos	Cargo que ocupa atualmente	QA do PMO	Gestor de programa	Coordenadora de desenvolvimento, gestão de portefólio
	Experiência na área	Implementações de PMOs durante 5 anos	Gestão de projetos tecnológicos há cerca de 20 anos	15 anos em projetos e portefólios aplicados a I&D
Dados da estrutura organizacional	Nº de projetos sob alçada (estimativa)	15	30	14
	Dimensão das equipas de desenvolvimento	Entre 10 e 15	10	Entre 4-5 elementos, distribuídos
	Natureza dos projetos sob alçada	Investigação e Desenvolvimento	Investigação de <i>software</i> , materiais, física entre outros	Investigação e Desenvolvimento
	Abordagens utilizadas	Scrum, <i>waterfall</i>	Scrum, <i>waterfall</i>	Scrum, <i>waterfall</i> e híbrido

		Entrevistado A	Entrevistado B	Entrevistado C
	Ferramentas utilizadas	Quadro <i>kanban</i> , promovidas pelo Scrum	Excel, Word	TFS, Jira, Open Project
	Métricas utilizadas	<i>Deliverables</i> , patentes produzidas	Ao nível dos outputs	Outputs e internas (velocidade e progresso em tempo-real)

Os dados compilados na Tabela 15 são meramente informacionais, permitindo analisar as respostas que os entrevistados deram na terceira fase da entrevista, com base no seu perfil profissional e nas características do PMO em que trabalha. Essa análise é apresentada no subcapítulo seguinte.

4.2 Comentários ao modelo inicial

Neste capítulo são analisados os dados recolhidos nas entrevistas, sendo detalhado de que forma contribuíram para o melhoramento, validação e identificação de limitações do ACO.

4.2.1 Sugestões

Durante a entrevista foram recolhidas algumas sugestões e correções proferidas explícita ou implicitamente pelos profissionais entrevistados, estando compiladas na Tabela 16, onde para cada sugestão é identificada a citação do transcrito que lhe deu origem e a remodelação a ser feita no modelo final do ACO.

Tabela 16- Sugestões dos entrevistados e respetivas alterações

Sugestão	Citação	Medida remodeladora
1. Alterar a representação gráfica do modelo, colocando o Módulo do Ambiente Comum em	Entrevista A - "...alteraria a representação e colocaria o ambiente comum em baixo para ser mais intuitivo que se trata da	Alterar a representação gráfica do modelo.

Sugestão	Citação	Medida remodeladora
baixo dos restantes módulos do ACO básico.	base. Para transmitir que é algo comum a ambas”	
2. Explicar a designação dada ao modelo.	Entrevista A – “[...] o nome à primeira vista destoa um pouco do conceito de PMO”	Explicar a designação dada.
3. Explicar a terminologia utilizada na divisão dos submodelos	Entrevista C – “Não sei se lhe chamaria avançado porque tem mais a ver com o tipo de gestão [...] porque avançado quer dizer que eu vou ter práticas mais avançadas e não é o caso, o tipo de gestão é que é diferente”	Incluir uma explicação detalhada do porquê da terminologia utilizada.
4. Utilizar ferramentas específicas para suportar a gestão de projetos ágeis.	Entrevista C – “[...] utilizamos várias ferramentas que são muito boas para gerir os projetos ágeis [...] utilizamos ferramentas como o TFS, o Jira e o Open Project”	Acrescentar as ferramentas às recomendações da proposta.
5. Considerar práticas de supervisão na gestão de portefólio	Entrevistado C- “[...] existe uma <i>pool</i> de práticas que o gestor de projeto tem de fazer mas que o gestor de portefólio simplesmente supervisiona, que não estão incluídas neste modelo”	Estabelecer como trabalho futuro devido à limitação temporal.
6. Definir como futuro trabalho prioritário a implementação do modelo numa organização.	Entrevista B – “Se pudesse aconselhar no que seria o próximo passo neste trabalho seria aplicar isto numa organização”	Dar prioridade a esta atividade como trabalho futuro.

Todas as sugestões da Tabela 16 resultaram em alterações, quer à proposta do ACO quer a outras componentes da dissertação, designadamente, o trabalho futuro recomendado. Essas alterações são descritas no modelo final do ACO, presente no subcapítulo 4.3.

4.2.2 Validações

Mais do que contribuir para a avaliação e refinamento da proposta, os entrevistados também contribuíram para a validação da mesma. A corroboração por parte dos entrevistados, experientes na gestão e coordenação de projetos ágeis, permite que este modelo obtenha um certo nível de validação quando à sua utilidade e, sobretudo, quanto à adequabilidade dos pressupostos em que se baseia e quanto ao conteúdo que recomenda ser aplicado.

De seguida é explicado como esta validação foi feita. Ressalva-se que os resultados obtidos carecem de maior validação, sobretudo através da implementação do ACO e consequente análise da sua utilidade. Essa atividade é prioritária para os trabalhos futuros.

As práticas ao nível da gestão de portefólio de projetos ágeis têm maior semelhança com as práticas da gestão de portefólios de projetos tradicionais, do que as do nível de gestão de equipas ágeis têm com as da gestão de equipas tradicionais. Esta afirmação é fundamentada com o facto de as abordagens ágeis serem direcionadas, exclusivamente, para o nível de gestão de equipas, incentivando a autogestão das equipas. Essa autogestão faz com que o ACO fique limitado no tipo de práticas que pode aplicar a esse nível. No entanto ao nível de portefólio estas limitações são praticamente inexistentes.

Este fenómeno foi corroborado pelo entrevistado C, que é gestor de portefólio, que defendeu que na sua atividade profissional é indiferente lidar com projetos de desenvolvimento ágil ou em cascata, excetuando na alocação de recursos, que com projetos ágeis é feita recorrentemente, ao contrário do que acontece com os projetos em cascata. O objetivo primordial em ambos os casos é alinhar os resultados desses projetos com a estratégia estabelecida pela gestão executiva.

O empoderamento das equipas tem outras implicações, nomeadamente, o tipo de controlo que é possível exercer sobre as equipas. Esta linha de pensamento foi corroborada pelos três entrevistados que afirmaram que o ACO devia, de facto, assumir apenas um papel de suporte, estando impossibilitado de assumir papéis de controlo ou diretivos por causa desse empoderamento das equipas. Isto vai ao encontro do pressuposto de que o ACO deve assumir um papel de suporte.

Quanto às métricas que são utilizadas para acompanhar os projetos, o ACO recomenda um conjunto de práticas, distribuídas pelos vários módulos, que visam abordar as peculiaridades da mentalidade ágil. Estas métricas diferenciam-se das métricas dos projetos tradicionais porque a

iteratividade dos projetos ágeis permite fazer um acompanhamento mais frequente. Estas diferenças foram corroboradas por todos os entrevistados, que afirmaram ter de adaptar as métricas que recolhem tendo em conta tipo de desenvolvimento.

Quando questionados sobre a utilidade do ACO para ser implementado no contexto atual do mercado, todos os entrevistados responderam afirmativamente, no entanto houve algumas ressalvas. O entrevistado C aludiu para o facto das práticas do portefólio irem além das que são propostas no ACO, sobretudo no que toca à supervisão de gestores de projetos e programas, pelo que a eventual implementação deveria começar pelo ACO básico, dando a oportunidade de maturar o modelo progressivamente. O entrevistado B também salientou que a implementação deveria ser feita primeiro com um projeto piloto, promovendo uma inovação incremental.

Finalmente, quanto à estrutura e práticas do ACO, a opinião unânime dos entrevistados foi de que as práticas são as mais indicadas para suportar e coordenar os projetos ágeis, assim como os programas e portefólios que constituem. Apenas o entrevistado C, como já foi referido, avisou para o facto de existirem outras práticas, sobretudo de supervisão, que poderiam ser adicionadas. Devido à limitação de tempo esta sugestão só poderá ser abordada num trabalho futuro. Quanto à distribuição feita, com a divisão em submodelos e módulos, todos os entrevistados concordaram que seria a mais adequada.

4.2.3 Limitações

Para além do melhoramento e da validação do ACO, os entrevistados também contribuíram para a identificação de limitações no modelo, como o facto de o âmbito do ACO se restringir apenas a organizações de desenvolvimento de *software*. No entanto, esta limitação advém do próprio contexto em que a investigação está inserida que são as empresas de TSI, estando prevista na definição da finalidade da dissertação.

Outra limitação identificada prende-se com a premissa do ACO suportar exclusivamente projetos ágeis, não oferecendo soluções para organizações com projetos de ambos os paradigmas da gestão de projetos. Mais uma vez, esta limitação estava prevista na definição da finalidade desta dissertação, pois o objetivo era desenvolver um artefacto com características exclusivas de suporte aos projetos ágeis.

4.3 Caracterização final do modelo

Tendo em conta as sugestões identificadas no capítulo anterior, realizou-se um conjunto de alterações ao trabalho até então desenvolvido.

Na Figura 16 está a visão interna da proposta final do ACO. Continua organizada da mesma forma, com dois submodelos, o ACO básico e o ACO avançado, sendo que cada um tem os mesmos módulos. Apenas foi alterada a representação gráfica dos módulos do ACO básico para tornar mais intuitiva a sua interpretação, ficando o MAC por baixo do MPR e do MPI aludindo ao facto de ser a fundação destes dois módulos. Com esta alteração atendeu-se à sugestão 1.

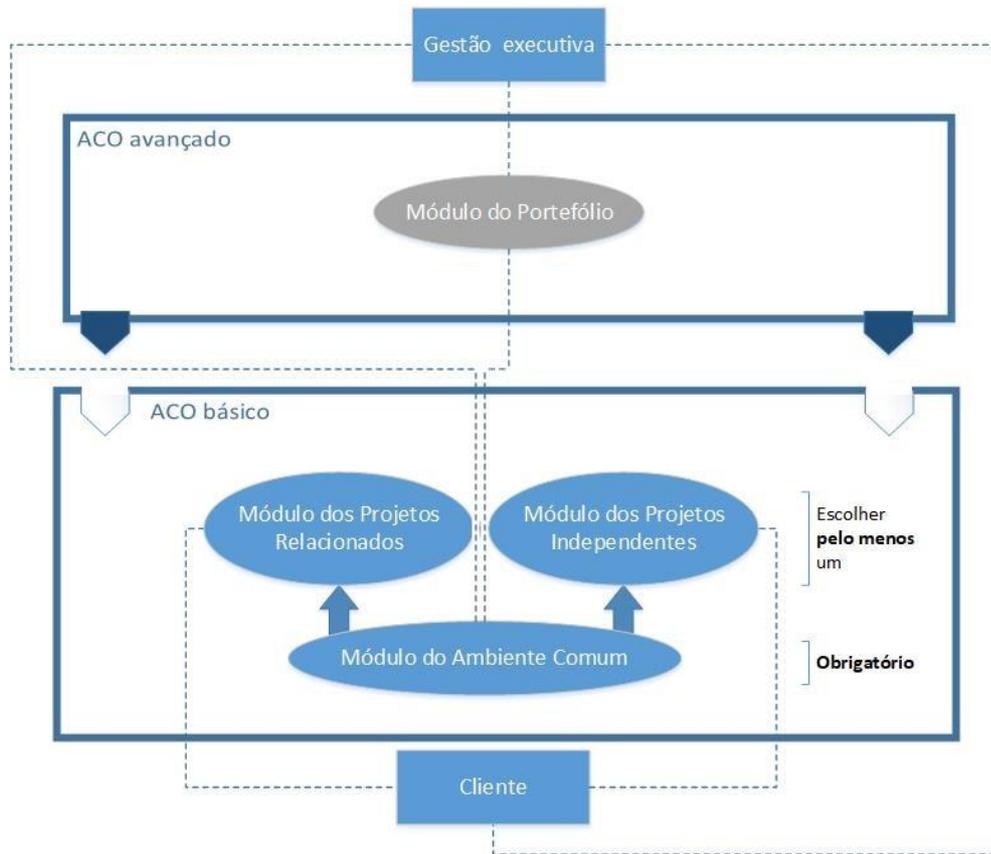


Figura 16- Estrutura refinada do ACO

Tendo em conta a sugestão 2 e seguindo a argumentação de Mike Cohn (2010) explicada no subcapítulo 3.1.5, optou-se por renomear o PMO ágil desenvolvido. Não se utilizou esta designação porque não era representativa do papel que se espera que assuma. O nome definido, “*Agile Coordination Office*”, foi escolhido com base em dois fatores:

- Pretendia-se que fosse claro que não está limitado a uma única abordagem ágil, como é o caso dos exemplos enumerados no subcapítulo 3.1.5 que se referiam sobretudo ao Scrum. Por esse motivo optou-se pela designação “Agile” que é mais abrangente.
- Como o seu papel é primariamente de suporte, decidiu-se não utilizar o termo “management” pois pretendia-se realçar que o ACO não assume papéis de maior controlo sob os projetos. Optou-se pelo termo “Coordination” pois as práticas recomendadas pelo ACO, explicadas

anteriormente, são maioritariamente relacionadas com a coordenação entre equipas, o auxílio da coordenação dentro das equipas e a coordenação dos vários projetos e programas dentro dos portefólios.

Para clarificar as dúvidas que surgiram na sugestão 3, quanto à terminologia utilizada nos submodelos do ACO, é importante referir que os termos utilizados, nomeadamente “básico” e “avançado”, foram escolhidos com base em dois critérios: (1) o número de projetos e/ou programas pelo qual o ACO vai ser responsável (2) a evolução desse número ao longo do tempo. A terminologia não visa retratar a complexidade das práticas que cada modelo apregoa.

Quanto ao primeiro critério, o ACO básico destina-se a organizações cujo número de projetos e/ou programas é ainda pequeno. Já o ACO avançado só deve ser implementado quando as organizações já tem um conjunto considerável de projetos e/ou programas que necessitam de ser geridos como um todo, alinhando-os com a estratégia da organização.

Quanto ao segundo critério, uma organização, ao longo do tempo, pode aumentar o seu número de projetos e/ou programas, surgindo então a necessidade de um ACO avançado ou vice-versa, terminar vários projetos e/ou programas e deixar de ter a necessidade deste modelo. A terminologia neste caso tem o propósito de refletir a flexibilidade que a conjugação deste dois submodelos confere à organização implementadora, num determinado momento, sendo necessário apenas um pacote básico para um cenário cujo número de projetos e/ou programas é limitado e um pacote avançado num cenário de maior escala.

A necessidade de ter uma gestão de portefólio também pode surgir do nível de maturidade da organização e assim sendo, uma organização muito madura pode querer logo à partida gerir o portefólio de projetos mesmo sendo o número de projetos limitado. Já por isso dentro de cada submodelo existe uma divisão em módulos. Neste caso aqui descrito a organização embora não tivesse ou número considerável de projetos poderia escolher o módulo do ACO básico em que mais se enquadrava.

No que toca à sugestão 4, decidiu-se acrescentar a utilização de ferramentas às recomendações do ACO. No entanto, por não haver tempo suficiente para investigar estas ferramentas, aconselha-se como trabalho futuro a sua investigação e respetivo mapeamento com as práticas do ACO, permitindo identificar os módulos para os quais seriam mais adequadas.

A sugestão 5 não pôde ser incluída nesta dissertação pois implicaria um novo momento de pesquisa. Como estas sugestões foram recolhidas numa fase final da dissertação, o tempo restante era bastante limitado. Como tal, esta sugestão é aconselhada como um trabalho futuro para melhorar o modelo.

Por fim, a sugestão 6 foi direcionada para o trabalho de investigação e não para o modelo desenvolvido. Tendo em conta a recomendação dos entrevistados para priorizar como próximo passo da investigação uma implementação do modelo desenvolvido, aconselha-se que o trabalho sucessor a esta dissertação seja essa mesma implementação. Este e outros trabalhos futuros são detalhados no próximo capítulo.

No que toca às práticas recomendadas, em ambos os processos de validação, não sofreram qualquer alteração, mantendo-se portanto o modelo final do ACO com as mesmas 19 práticas supramencionadas no modelo inicial do ACO. O mesmo aconteceu com os papéis sugeridos e portanto, mantêm-se com a mesma designação e alocados aos mesmos módulos.

Tal como referido na proposta inicial, o ACO apresenta grande flexibilidade de adaptação aos contextos das organizações que o pretendam implementar, simplificando o processo para empresas com menor maturidade e menor complexidade e permitindo que o ACO seja gradualmente incrementado com novos módulos ao ritmo que as necessidades vão surgindo.

De um modo geral, o contributo do ACO é salientado pela abrangência de ações que pode efetuar, resultado da conjugação dos três fatores que não se encontrou em mais nenhuma dos casos de estudo analisados: (1) o suporte a todos os níveis de gestão, (2) a definição clara de uma estrutura que organiza os seus serviços e (3) a definição de processos para o implementar.

5. CONCLUSÕES E TRABALHOS FUTUROS

Neste capítulo final é feito o balanço entre os objetivos propostos e os resultados obtidos, apresentadas as principais ilações alcançadas, explicados os contributos resultantes, descritas as limitações que surgiram e, finalmente, sugerido um conjunto de trabalhos futuros que devem ser realizados para avançar e complementar o conhecimento desenvolvido durante esta dissertação.

5.1 Conclusões

A finalidade desta dissertação era caracterizar um modelo de um *Agile Coordination Office* para empresas de TSI, com base na revisão de literatura relacionada e na análise de propostas de gabinetes ágeis já existentes. Para cumprir esta finalidade, no início da dissertação foi definido um conjunto de objetivos, os quais foram sendo cumpridos conforme se explica de seguida.

Inicialmente foi feita uma revisão de literatura sobre os principais temas relacionados com os gabinetes de gestão de projetos ágeis. Este era o primeiro objetivo e ao longo da dissertação, à medida que o conhecimento foi sendo aprofundado, consequentemente, a revisão de literatura foi sendo atualizada para refletir toda a fundamentação utilizada durante a investigação.

A etapa seguinte foi a análise de propostas semelhantes, cumprindo-se portanto o segundo objetivo. Esta análise revelou que as propostas existentes são focadas, sobretudo, em PMOs tradicionais adaptados às abordagens ágeis ou em gabinetes de gestão de projetos ágeis implementados durante o momento de transição da abordagem tradicional para a ágil. No entanto, não foram encontradas nenhuma proposta relativas à definição e caracterização detalhada de um gabinete de gestão de projetos ágeis. Também não foi encontrado nenhuma proposta estruturada de como manter um gabinete deste género a operar depois de implementado. Esta investigação contribuiu para colmatar esta lacuna na literatura ao apresentar um modelo que cumpre os requisitos citados.

O terceiro objetivo era a conceção de um modelo de um ACO. O modelo inicial foi feito com base no conhecimento adquirido na revisão de literatura e na análise das propostas. Este modelo para além de providenciar soluções para os vários níveis de gestão e de promover um processo de implementação gradual e flexível, também reflete os potenciais princípios de desenvolvimento ágil de larga escala identificados na literatura.

Esse modelo foi submetido como um artigo para uma conferência internacional na área de gestão de projetos (ver Anexo II – Artigo publicado), onde foi avaliado por um conjunto de revisores, cujos

pareceres contribuíram para que fosse refinado. Depois de efetuados os melhoramentos, o artigo foi aceite como *full paper* (Pinto & Ribeiro, 2018).

Tendo concluído este primeiro momento de avaliação e refinação do artefacto, procedeu-se para um segundo momento de avaliação tendo em conta a perspetiva dos praticantes, recorrendo-se para tal a entrevistas semiestruturadas. Estes dois momentos de avaliação permitiram cumprir o objetivo de avaliar e comunicar o conhecimento desenvolvido, através do artigo publicado e do presente relatório de dissertação.

Deste segundo momento de avaliação resultou um conjunto de sugestões que foram tidas em consideração e que deram origem a alterações no artefacto proposto. Para além das alterações, os entrevistados também corroboraram e validaram os pressupostos e a estruturação do ACO, com base na sua vasta experiência profissional. Isto permitiu-nos alcançar com sucesso o penúltimo objetivo, de caracterizar o modelo do ACO avaliado, refinado e validado.

Assim sendo, pode-se concluir que todos os objetivos delineados foram concretizados. Cumpriu-se a finalidade desta dissertação, de caracterizar um *Agile Coordination Office* para empresas de TSI, através do modelo devidamente caracterizado e fundamentado neste relatório.

Esta dissertação contribuiu para a comunidade científica com um modelo teórico, baseado na literatura e validado conceptualmente pelos revisores do artigo e pelos experientes profissionais entrevistados. Embora não tenha sido validado com uma aplicação prática, contribuiu também para a comunidade de praticantes como um ponto de partida para uma futura implementação de um gabinete de gestão de projetos ágeis.

Posto isto, durante o decorrer desta investigação obtiveram-se outras ilações interessantes cuja sua partilha também contribui para o avanço do conhecimento. A primeira foi que as abordagens ágeis influenciam mais diretamente as práticas de um PMO ao nível do projeto do que ao nível do portefólio. Teoricamente, isto é corroborado pelo facto de as próprias metodologias ágeis terem sido concebidas para a gestão de projetos de pequena dimensão e na perspetiva dos praticantes, durante as entrevistas, constatou-se que esta ilação se verifica pois partilhavam das dificuldades expostas na literatura, relativamente às problemáticas do desenvolvimento ágil de larga escala.

Outra conclusão foi de que um PMO ágil deve assumir uma atitude de suporte. Enquanto na gestão de projetos tradicionais os PMOs variam quanto ao seu nível de controlo e comando, no caso da gestão de projetos ágeis este nível limita-se ao suporte devido à natureza das abordagens ágeis, que promovem a autogestão da equipa, dando-lhes autonomia e poder de decisão, sendo portanto incompatível com um papel mais diretivo por parte do PMO.

Contudo, como em qualquer investigação, este estudo apresentou algumas limitações. Uma das limitações foi o facto de o artefacto desenvolvido nesta dissertação não poder ser avaliado e validado por outros métodos de pesquisa. Isto porque se trata de um novo modelo conceptual para uma área ainda em desenvolvimento, sendo difícil encontrar praticantes cuja atividade profissional se enquadre neste contexto. Portanto, apenas se conseguiu utilizar métodos qualitativos para avaliar o artefacto. O recurso a métodos quantitativos requereria que um elevado número de profissionais fosse consultado, para que a amostra da população fosse significativa.

Outra limitação foi o facto dos entrevistados serem todos de áreas semelhantes, nomeadamente de investigação e desenvolvimento. Apesar de ocuparem cargos em níveis diferentes dentro das organizações para as quais trabalham, a área de negócio dessas organizações era muito semelhante. Esta limitação ocorreu devido à mesma premissa da limitação anterior, ou seja, por se tratar de uma área bastante subdesenvolvida não se encontraram mais candidatos dentro do perfil pretendido, de outras áreas, que tivessem dispostos a contribuir para esta investigação.

Por fim, o tempo definido para a investigação também constituiu uma limitação visto que, algumas sugestões dos entrevistados não puderam ser implementadas pois foram recolhidas numa etapa final da avaliação do artefacto desenvolvido. No entanto, para aproveitar o contributo dos entrevistados, essas sugestões foram recomendadas como trabalho futuro, conforme se descreve no subcapítulo seguinte.

5.2 Trabalho futuro

O trabalho desenvolvido durante esta dissertação constitui uma base sólida, baseada na literatura e validada pelos revisores e profissionais da área, para que o modelo do ACO possa continuar a ser avançado em trabalhos futuros. Assim sendo, e tendo em conta as sugestões recolhidas durante as avaliações, identificou-se um conjunto de propostas para trabalhos futuros.

O que se demonstra mais prioritário, tendo em conta a opinião dos especialistas entrevistados, é a implementação do ACO. Num primeiro momento a prioridade deverá ser implementar o ACO numa organização da área para a qual ele foi concebido, nomeadamente, das TSI. O resultado desta implementação seria a validação prática do modelo desenvolvido nesta dissertação.

Também como identificado nas sugestões dos entrevistados, recomenda-se que seja feito um mapeamento entre as práticas recomendadas pelo ACO e as ferramentas disponíveis para suportar a gestão de projetos ágeis, permitindo identificar quais os módulos que poderão beneficiar da sua aplicação. Os entrevistados enumeraram algumas ferramentas (e.g. JIRA, Open Project, TFS) que podem

servir como base para este trabalho. O resultado seria um modelo atualizado do ACO com recomendações alusivas à forma de utilizar tais ferramentas.

Também se recomenda que sejam exploradas e adicionadas novas práticas ao MP do ACO avançado, sobretudo práticas relacionadas com supervisão, tornando-o mais completo. Tal como já foi referido, esta atividade e a descrita no parágrafo anterior não foram feitas no presente trabalho devido à limitação do tempo disponível e por não terem sido detetadas na literatura relacionada analisada.

Outra proposta para trabalho futuro seria o mapeamento entre o ACO e as *frameworks* de desenvolvimento ágil de larga escala. Seria importante perceber de que forma as práticas promovidas por essas *frameworks* coincidem ou complementam as práticas recomendadas pelo ACO. Este trabalho também serviria como resposta à necessidade identificada na revisão de literatura de que estas *frameworks* devem ser mais estudadas cientificamente.

Para ter uma validação objetiva relativamente à utilidade do ACO seria primeiro necessário implementá-lo numa organização. Depois de implementado seria então possível comparar o estado antes e depois da organização, demonstrando as mais-valias que teoricamente lhe fornece.

Finalmente, a proposta mais ambiciosa para um trabalho futuro seria a generalização do modelo para que possa ser adaptado a organizações de outras áreas. Assim como as metodologias ágeis já não se limitam apenas ao domínio do desenvolvimento de *software*, também o ACO deverá acompanhar as áreas onde estas metodologias sejam adotadas. Isto é, onde quer que haja gestão de projetos ágeis, há um mercado potencial para o ACO ser implementado.

REFERÊNCIAS

- Abrahamsson, P., Salo, O., Ronkainen, J., & Warsta, J. (2002). Agile Software Development Methods : Review and Analysis. *VTT Technical Report*.
- Aghina, W., Lackey, G., Lurie, M., & Murarka, M. (2018). The five trademarks of agile organizations. Retrieved from <https://www.mckinsey.com/business-functions/organization/our-insights/the-five-trademarks-of-agile-organizations>
- Alqudah, M., & Razali, R. (2016). A Review of Scaling Agile Methods in Large Software Development. *International Journal on Advanced Science, Engineering and Information Technology*, 6(6), 828. <https://doi.org/10.18517/ijaseit.6.6.1374>
- Alsadeq, I. (2011). Establishing a Project Management Office (PMO) Using the Agile Approach PMO : Models and Functions. *PMI Global Congress Proceedings*, 1–7.
- Alter, S. (1992). *Information Systems: A Management Perspective*. Addison-Wesley.
- Amaral, L. A. M. d. (1994). *PRAXIS: Um referencial para o Planeamento de Sistemas de Informação*. Universidade do Minho. Retrieved from <http://hdl.handle.net/1822/49>
- Anderson, D. (2010). *Kanban: Successfull evolutionary change for your technology business*. Sequim, Washington: Blue Hole Press.
- Anderson, D., & Carmichael, A. (2016). *Essential kanban condensed* (1st ed.). Seattle, Washington: Blue Hole Press.
- APM. (2012). *APM Body of Knowledge*. (6th, Ed.). APM.
- Aubry, M., Hobbs, B., & Thuillier, D. (2007). A new framework for understanding organisational project management through the PMO. *International Journal of Project Management*, 25, 328–336.
- Aubry, M., Müller, R., Hobbs, B., & Blomquist, T. (2010). Project management offices in transition. *International Journal of Project Management*, 28(8), 766–778. <https://doi.org/10.1016/j.ijproman.2010.05.006>
- Augustine, S., & Cuellar, R. (2006). The Lean-Agile PMO : Using lean thinking to accelerate project delivery (Vol. 7, pp. 1–24). Cutter Consortium.
- Beck, K. (1999a). Embracing Change with Extreme Programming. *Computer*, 32(10), 70–77. <https://doi.org/10.1109/2.796139>
- Beck, K. (1999b). *Extreme programming explained: embrace change* (1st ed.). Addison-Wesley.
- Beck, K., & Andres, C. (2004). *Extreme programming explained: embrace change* (2nd ed.). Addison-

- Wesley.
- Beck, K., Beedle, M., Bennekum, A. van, Cockburn, A., Cunningham, W., Fowler, M., ... Thomas, D. (2001). Manifesto for Agile Software Development. Retrieved October 18, 2017, from <http://agilemanifesto.org/>
- Boehm, B. (1988). A spiral Model of Software Development and Enhancement. *Computer*. <https://doi.org/10.1109/2.59>
- Boehm, B. (2002). Get ready for agile methods, with care. *Computer*, 35(1), 64–69. <https://doi.org/10.1109/2.976920>
- Boehm, B., & Turner, R. (2003). Using risk to balance agile and plan-driven methods. *Computer*, 36(6), 57–66. <https://doi.org/10.1109/MC.2003.1204376>
- Buckingham, R. A., Hirschheim, R. A., Land, F. F., & Tully, C. J. (1987). Information systems curriculum: a basis for course design. In *Information Systems Education: Recommendations and Implementation*. Cambridge University Press.
- Carvalho, J. A. (2000). Information System? Which One Do You Mean? (pp. 259–277). <https://doi.org/10.1007/978-0-387-35500-9>
- Cohen, D., & Crabtree, B. (2006). Semi-structured Interviews Recording Semi-Structured interviews. *Qualitative Research Guidelines Project*, 2. Retrieved from <http://www.qualres.org/HomeSemi-3629.html>
- Cohn, M. (2010). *Succeeding With Agile : Software development using Scrum*. Upper Saddle River, NJ: Addison-Wesley.
- Crawford, J. K. (2010). *The Strategic Project Office* (2nd ed.). Boca Raton ,Florida: CRC Press.
- Desouza, K. C., & Evaristo, J. R. (2006). Project management offices : A case of knowledge-based archetypes. *International Journal of Information Management*, 26, 414–423. <https://doi.org/10.1016/j.ijinfomgt.2006.07.002>
- Dikert, K., Paasivaara, M., & Lassenius, C. (2016). Challenges and success factors for large-scale agile transformations: A systematic literature review. *Journal of Systems and Software*, 119, 87–108. <https://doi.org/10.1016/j.jss.2016.06.013>
- Dingsøyr, T., Fægri, T. E., & Itkonen, J. (2014). What Is Large in Large-Scale? A Taxonomy of Scale for Agile Software Development. In A. Jedlitschka, P. Kuvaja, M. Kuhrmann, T. Männistö, J. Münch, & M. Raatikainen (Eds.), *PROFES 2014: Product-Focused Software Process Improvement* (Vol. 8892, pp. 273–276). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-13835-0_20

- Dingsøyr, T., & Moe, N. B. (2013). Research challenges in large-scale agile software development. *ACM SIGSOFT Software Engineering Notes*, 38(5), 38. Retrieved from <http://dl.acm.org/citation.cfm?doid=2507288.2507322>
- Dingsøyr, T., & Moe, N. B. (2014). Towards Principles of Large-Scale Agile Development - A Summary of the Workshop at XP2014 and a Revised Research Agenda. *Agile Methods. Large-Scale Development, Refactoring, Testing, and Estimation, Volume 199*, 1–8. <https://doi.org/10.1007/978-3-319-14358-3>
- Dingsøyr, T., Nerur, S., Baliyepally, V., & Moe, N. B. (2012). A decade of agile methodologies: Towards explaining agile software development. *Journal of Systems and Software*, 85(6), 1213–1221. <https://doi.org/10.1016/j.jss.2012.02.033>
- Dingsøyr, T., Rolland, K., Moe, N. B., & Seim, E. A. (2017). Coordination in multi-team programmes: An investigation of the group mode in large-scale agile software development. *Procedia Computer Science*, 121, 123–128.
- Falkenberg, E. D., Hesse, W., Lindgreen, P., Nilsson, B. E., Han Oei, J. L., Rolland, C., ... Voss, K. (1998). FRISCO: A Framework of Information Systems Concepts. *International Federation for Information Processing (Web Edition)*.
- Fernandes, J. M., & Machado, R. J. (2016). *Requirements in Engineering Projects*. Springer International Publishing. <https://doi.org/10.1007/978-3-319-18597-2>
- Ferreira, M., Tereso, A., Ribeiro, P., Fernandes, G., & Loureiro, I. (2013). Project Management Practices in Private Portuguese Organizations. *Procedia Technology*, 9, 608–617. <https://doi.org/10.1016/j.protcy.2013.12.067>
- Fontana, A., & Frey, J. H. (2000). The interview: from structured questions to negotiated text. In N. K. Denzin & Y. S. Lincoln (Eds.), *Handbook of qualitative research* (2nd ed., pp. 645–672). Thousand Oaks, CA.
- Freudenberg, S., & Sharp, H. (2010). The top 10 burning research questions from practitioners. *IEEE Software*, 27(5), 8–9. <https://doi.org/10.1109/MS.2010.129>
- Gonçalves, D., Cruz, B., & Varjão, J. (2008). Particularidades dos diferentes tipos de projetos de desenvolvimento de software. In *21º Congresso Internacional de Administração-Gestão estratégica na era do conhecimento (ADM)*. Brasil.
- Gregor, S., & Hevner, A. (2013). Positioning and Presenting Design Science Research for Maximum Impact. *MIS Quarterly*, 37(2), 337–355.
- Gregory, P., Barroca, L., Sharp, H., Deshpande, A., & Taylor, K. (2016). The challenges that challenge:

- Engaging with agile practitioners' concerns. *Information and Software Technology*, 77, 92–104.
- Hastie, S., & Wojewoda, S. (2015). <https://www.infoq.com>. Retrieved January 9, 2018, from <https://www.infoq.com/articles/standish-chaos-2015>
- Hevner, A., March, S. T., Park, J., & Ram, S. (2004). Design Science in Information Systems Research. *MIS Quarterly*, 28(1), 75–105.
- Hill, G. M. (2008). *The Complete Project Management Office Handbook*. Auerbach Publications (2nd ed.). Boca Raton, New York: Auerbach Publications.
- Hobbs, J. ., & Aubry, M. (2007). A multi-phase research program investigating project management offices (PMOs): the results of phase I. *Project Management Journal*, 38(1), 74–86.
- Hove, S. E., & Anda, B. (2005). Experiences from Conducting Semi-structured Interviews in Empirical Software Engineering Research. *11th IEEE International Symposium on Software Metrics (METRICS 2005), Como, Italy, (Metrics)*, 23–. <https://doi.org/10.1109/METRICS.2005.24>
- Hubbard, D. G., & Bolles, D. L. (2015). PMO Framework and PMO Models for Project Business Management. *PM World Journal*, 4(1), 22.
- Jalal, M. P., & Koosha, S. M. (2015). Identifying organizational variables affecting project management office characteristics and analyzing their correlations in the Iranian project-oriented organizations of the construction industry. *International Journal of Project Management*, 33(2), 458–466. <https://doi.org/10.1016/j.ijproman.2014.06.010>
- Jones, G. R. (2013). Organizations and organizational effectiveness. In *Organizational Theory, Design, and Change* (pp. 1–27). Boston ,MA: Pearson/Prentice Hall Company.
- Kerzner, H. (2009). *Project management: a systems approach to planning, scheduling, and controlling* (10th ed.). Hoboken, New Jersey: John Wiley & Sons.
- Monteiro, A., Santos, V., & Varajão, J. (2016). Project Management Office Models – a review. *Procedia Computer Science*, 100, 1085–1094. Retrieved from <http://dx.doi.org/10.1016/j.procs.2016.09.254>
- Nerur, S., Mahapatra, R., & Mangalaaraj, G. (2005). Challenges of migrating to agile methodologies. *Communications of the ACM*, 48(5), 72–78.
- OGC. (2009). *Managing Successful Projects with PRINCE2* (5th ed.). Office of Government Commerce.
- Paasivaara, M., Lassenius, C., & Heikkilä, V. T. (2012). Inter-team coordination in large-scale globally distributed scrum. *Proceedings of the ACM-IEEE International Symposium on Empirical Software Engineering and Measurement - ESEM '12*, 235. <https://doi.org/10.1145/2372251.2372294>
- Pinto, J., & Ribeiro, P. (2018). Characterization of an Agile Coordination Office for IST companies.

- Procedia Computer Science*, 138, 859–866. <https://doi.org/10.1016/j.procs.2018.10.112>
- Poppendieck, M., & Poppendieck, T. (2003). *Lean software development: an agile toolkit*. Computer (Vol. 36). Addison-Wesley.
- Power, K. (2011). The Agile Office: Experience Report from Cisco's Unified Communications Business Unit. In *2011 Agile Conference* (pp. 201–208). Salt Lake City, UT. <https://doi.org/10.1109/AGILE.2011.7>
- Project Management Institute. (2008). *The Standard for Portfolio Management*. Pennsylvania: Project Management Institute.
- Project Management Institute. (2013a). *A Guide to the Project Management Body of Knowledge* (5th ed.). Pennsylvania: Project Management Institute.
- Project Management Institute. (2013b). Pulse of the Profession: PMO Frameworks Report. Retrieved from <https://www.pmi.org/-/media/pmi/documents/public/pdf/learning/thought-leadership/pulse/pmo-frameworks.pdf>
- Project Management Institute. (2013c). *The Standard for Program Management*. Pennsylvania: Project Management Institute.
- Project Management Institute. (2017). *Agile Practice Guide*. (Project Management Institute, Ed.).
- Reifer, D. J., Maurer, F., & Erdogmus, H. (2003). Scaling agile methods. *IEEE Software*, 20(4), 12–14. <https://doi.org/10.1109/MS.2003.1207448>
- Royce, W. W. (1970). Managing the Development of Large Software Systems. In *IEEE WESCON* (pp. 1–9). Los Angeles, USA. Retrieved from <http://www.cs.umd.edu/class/spring2003/cmsc838p/Process/waterfall.pdf>
- Schwaber, K. (1994). SCRUM Development Process, (April 1987), 10–19.
- Schwaber, K., & Beedle, M. (2002). *Agile Software Development With Scrum*. Upper Saddle River, NJ: Prentice-Hall.
- Scrum.org. (2018). The Scrum Framework Poster. Retrieved January 13, 2018, from <https://www.scrum.org/resources/scrum-framework-poster>
- Serrador, P., & Pinto, J. K. (2015). Does Agile work? — A quantitative analysis of agile project success. *International Journal of Project Management*, 33(5), 1040–1051. <https://doi.org/10.1016/j.ijproman.2015.01.006>
- Singh, R. (2009). Identifying and overcoming the challenges of implementing a project management office. *European Journal of Information Systems*, 18(5), 409–427. <https://doi.org/10.1057/ejis.2009.29>

- Sliger, M. (2007). *A Project Manager's Survival Guide to Going Agile*.
- Soares, D. d. S. (1998). *Planeamento de Sistemas de Informação: Estudo das Variáveis que Condicionam a sua Estratégia de Execução*. Universidade do Minho.
- Soares, D. d. S., & Amaral, L. (2014). Reflections on the Concept of Interoperability in Information Systems. In *Proceedings of the 16th International Conference on Enterprise Information Systems* (pp. 331–339). <https://doi.org/10.5220/0004969703310339>
- Sommerville, I. (2010). *Software Engineering. Software Engineering* (9th ed.). Addison-Wesley. <https://doi.org/10.1111/j.1365-2362.2005.01463.x>
- Stellman, A., & Greene, J. (2014). *Learning Agile: Understanding Scrum, XP, Lean and Kanban*. O'Reilly Media, Inc.
- Sutherland, J. (2016). *SCRUM: A arte de fazer o dobro em metade do tempo* (1st ed.). Alfragide: Lua de Papel.
- Sutherland, J., & Schwaber, K. (2016). The scrum guide. The Definitive Guide to Scrum : The Rules of the Game, (July).
- Takeuchi, H., & Nonaka, I. (1986). The new new product development game. *Harvard Business Review*, (January).
- Tengshe, A., & Noble, S. (2007). Establishing the agile PMO: Managing variability across projects and portfolios. In *Agile 2007 (AGILE 2007)* (pp. 188–193). Washington, DC, USA.: IEEE. <https://doi.org/10.1109/AGILE.2007.24>
- Vaidya, A. (2014). Does DAD Know Best, Is it Better to do LeSS or Just be SAFe? Adapting Scaling Agile Practices into the Enterprise. *Thirty-Second Annual Pacific Northwest Software Quality Conference*, 1–18.
- Vaishnavi, V., Kuechler, B., & Petter, S. (2004). Design Science Research in Information Systems, 1–66. Retrieved from <http://www.desrist.org/design-research-in-information-systems/>
- Varajão, J. E., Domingues, C. E., Ribeiro, P. A., & Paiva, A. C. (2014). Failures in software project management - are we alone? A comparison with construction industry. *Journal of Modern Project Management*. <https://doi.org/10.3963/jmpm.v2i1.59>
- Versionone.com. (2018). The 12th State of agile report. Retrieved January 13, 2018, from <https://explore.versionone.com/state-of-agile/versionone-12th-annual-state-of-agile-report>
- Ward, J., Griffiths, P., & Whitmore, P. (1990). *Strategic Planning for Information Systems*. Chichester: John Wiley & Sons.
- Williams, L., & Cockburn, A. (2003). Agile Software Development: it's about Feedback and Change, *36*(6),

Referências

39–43. <https://doi.org/10.1109/MC.2003.1204373>

ANEXO I – GUIÃO DA ENTREVISTA

Dados biográficos

Cargo que ocupa atualmente: _____

Experiência na área: _____

Dados sobre a organização

Nº de projetos geridos (estimativa): ____ Dimensão das equipas de desenvolvimento: _____

Natureza dos projetos geridos: _____ Abordagens utilizadas: _____

Ferramentas utilizadas: _____ Métricas utilizadas: _____

Questões para o entrevistando

1. Quais as lacunas de um PMO quando os seus projetos adotam o *agile*? Quais as mais-valias?
2. As responsabilidades do PMO alteram-se perante projetos ágeis relativamente aos projetos *waterfall*? O nível de controlo sofre alterações?
3. Na literatura, maior parte das práticas são comuns a um PMO tradicional, o que muda é a forma como são realizadas, é consideravelmente diferente. Dada a sua experiência, quais as principais diferenças em relação ao modo como as práticas são executadas nestes diferentes contextos? O que muda entre o contexto de tradicional e ágil?
4. Durante a pesquisa concluímos que sobretudo ao nível da gestão de portefólios, as práticas mantêm-se praticamente imutáveis, alterando apenas questões como as métricas que são transmitidas para os superiores. Na sua experiência profissional como lida com a gestão de vários projetos cujas métricas são de natureza ágil?
5. Já escalou algum processo ágil? Ou recorreu a alguma *framework* para escalar o *agile* na organização? Se sim, quais as mais-valias que poderia acrescentar a um ACO? Utilizou alguma *framework* para fazer a gestão de programas/portefólios? (NEXUS, SAFe, etc.) E algum método como o “Scrum-of-Scrums”?
6. Optamos por dividir em submodelos e módulos o ACO do mesmo modo que uma tipologia de PMOs pode apresentar vários modelos, para diferentes contextos. Considera que esta divisão é útil para simplificar o processo de implementação e o reajustamento do ACO? Que outras práticas recomendaria para o ACO?
7. Concorda com este papel de suporte do ACO?
8. Que vantagens podia trazer um ACO no contexto atual de trabalho? E desafios?
9. Tem mais alguma sugestão?

Observações

(Registo de sugestões e de ramificações de temas que não estavam previamente definidas)

ANEXO II – ARTIGO PUBLICADO

Autores: José Pinto, Pedro Ribeiro

Conferência: *Projman 2018 – International Conference on Project MANagement*

Estado: Publicado

Abstract: Typically the Project Management Office activity is linked to the management and coordination of plan-driven projects, also known as waterfall or traditional projects. However, with the advent of agile methodologies in organizations of software development (the case of many information systems and technologies (IST)) no longer value the traditional PMO. It needs to be changed according to the agile values, so the organizations can extract benefits from such structure. These need to be fundamental changes in the responsibilities, practices and roles that a PMO should have. Also, it seems appropriate to rename it to something more descriptive and we chose to name it Agile Coordination Office (ACO). This paper presents the initial proposal of the ACO based on the existing literature. We propose the ACO to assume a behavior mainly supportive, due to the empowerment that every agile development team must have by definition. In addition, the architecture of this ACO aims to cover the various levels of management, from project and program up to the portfolio management. This division also reduces the complexity of ACO's implementation process and gives flexibility to rearrange the ACO over time.

Keywords: *Project management office, Agile PMO, Agile governance, Scaling agile*