



Lisbon School
of Economics
& Management
Universidade de Lisboa

MESTRADO

GESTÃO DE SISTEMAS DE INFORMAÇÃO

TRABALHO FINAL DE MESTRADO

RELATÓRIO DE ESTÁGIO

ESTÁGIO CURRICULAR EM GESTÃO DE PROJETOS NA *CELFOCUS*

ANDRÉ FILIPE DELGADO GOMES DA SILVA

OUTUBRO - 2021



Lisbon School
of Economics
& Management
Universidade de Lisboa

MESTRADO

GESTÃO DE SISTEMAS DE INFORMAÇÃO

TRABALHO FINAL DE MESTRADO

RELATÓRIO DE ESTÁGIO

ESTÁGIO CURRICULAR EM GESTÃO DE PROJETOS NA *CELFOCUS*

ANDRÉ FILIPE DELGADO GOMES DA SILVA

SUPERVISÃO:

JESUALDO FERNANDES

MARGARIDA AFONSO

OUTUBRO - 2021

AGRADECIMENTOS

Quero agradecer à Celfocus pela oportunidade de realizar o estágio curricular e às incríveis pessoas que conheci durante esta aventura, especialmente à Margarida Afonso, pela coordenação e orquestração dos processos internos e externos o que facilitou a formalização deste trabalho.

Ao professor Jesualdo Fernandes, pelo suporte e esforço em guiar-me e corrigir o meu trabalho.

Um agradecimento especial à Liliana, pelas palavras de incentivo e apoio nos momentos mais difíceis.

À minha família pelo apoio incondicional durante toda a minha jornada académica, em especial à minha mãe que tanto me inspira, e motiva-me a ser a melhor versão melhor de mim.

Sem esquecer a minha cadela, Kira, que sempre devolveu amor e carinho e acompanhou-me nas noites longas de trabalho.

GLOSSÁRIO

DoD – Definition of Done

DoR – Definition of Ready

FCFS – First Come First Serve

IoT – Internet of Things

IT - Information Technology

OMI – Order Management & Integration

PM – Project Manager

PMLC – Product Management Life Cycle

PO – Product Owner

PPTx – Power Point

SM – Scrum Master

SPOC – Single Person of Contact

SSOT – Single Source of Truth

SW – Software

Telco – Empresa de telecomunicações

TFM – Trabalho Final de Mestrado

UE – União Europeia

UK – United Kingdom

US – User Story

WO – Work Order

WP – Work Package

RESUMO

O presente trabalho procura explorar os conceitos de gestão de projetos de natureza informática, relacionando-os com o estágio curricular que o candidato participou.

Foram abordadas as tipologias *Waterfall* e *Agile* de forma a escrutinar a sua adaptabilidade nos projetos de *IT*. Foi possível analisar as vantagens e desvantagens de cada metodologia, assim como a sua adequabilidade. Após a análise teórica dos conceitos mencionados, procedeu-se a uma descrição pormenorizada das atividades realizadas pelo candidato durante o seu estágio, o qual dividiu-se em duas fases, a primeira centrada no acompanhamento das tarefas realizadas pelos PM's em projetos *Waterfall* e a segunda focada em projetos *Agile*, na *framework Scrum*.

Procurou-se identificar a aplicabilidade dos conceitos teóricos no contexto organizacional. Concluiu-se que os projetos informáticos se distinguem de projetos de outras áreas, por apresentarem características únicas que exigem novas técnicas de gestão. O contexto onde as organizações de *IT* operam é altamente imprevisível e complexo, exigindo flexibilidade na sua gestão.

Palavras-Chave: *Agile; Waterfall; IT; Estágio; PM; Scrum; Complexo; Flexibilidade; Gestão.*

ABSTRACT

The current work seeks to explore the concepts of IT project management, relating them with the internship in which the candidate participated.

The Waterfall and Agile methodologies were approached to scrutinize their adaptability in IT projects. It was possible to analyze the advantages and disadvantages of each methodology, as well as their suitability. After the theoretical analysis of the mentioned concepts, it followed a detailed description of the activities performed by the candidate during his internship. It was divided into two phases; the first focused on monitoring the tasks performed by PMs in Waterfall projects, and the second focused on Agile projects in the Scrum framework.

It was sought to identify the applicability of the theoretical concepts in the organizational context. It was concluded that IT projects differ from projects in other areas because they present unique characteristics that require new management techniques. The context where IT organizations operate is highly unpredictable and complex, requiring flexibility in their management.

Keywords: Agile; Waterfall; IT; Internship; PM; Scrum; Complex; Flexibility; Management.

ÍNDICE

Agradecimentos.....	iii
Glossário.....	iv
Resumo.....	v
Abstract	vi
Índice.....	vii
Tabela de Figuras	ix
1. Introdução.....	1
2. Revisão da Literatura	2
2.1. O que é um projeto?	2
2.2. Distinção de Projetos e Projetos Informáticos	3
2.3. O PMBOK.....	5
2.4. Pressupostos do Waterfall	8
2.5. Waterfall.....	9
2.6. Agile.....	11
2.6.1 Metodologia Scrum.....	12
2.6.1.1 Scrum	12
2.6.1.2 Scrum Master	13
2.6.1.2 Product Owner.....	13
2.6.1.3 Developers.....	13
2.6.1.4 Artefactos e Eventos do Scrum	13
2.6.1.5 Product backlog.....	13
2.6.1.6 Sprint Backlog.....	14
2.6.1.7 Increment.....	14
2.6.1.8 Sprint Planning.....	14

2.6.1.9 Daily Scrum.....	14
2.6.1.10 Sprint Review	15
2.6.1.11 Retrospective	15
3. Caracterização da Organização	15
3.1 Novabase	15
3.2 Celfocus.....	16
4. Caracterização do Estágio	16
4.1 PROJ-1 – Introdução	16
4.2 PROJ-1 – Tarefas desempenhadas	18
4.3 PROJ-1 – Alocação de Recursos.....	18
4.4 PROJ-1 – Documentação	19
4.5 PROJ-1 – Formação e Desenvolvimento Profissional	19
4.6 PROJ-1 – Fases do projeto	20
4.7 PROJ-2 – Introdução	23
4.8 PROJ-2 – Ferramentas Agile.....	24
4.9 PROJ-2 – Principais atividades	25
4.10 PROJ-2 – Documentação	27
5. Conclusão e Análise Crítica e interpretativa do estágio.....	29
Referências	31
Anexos.....	33

TABELA DE FIGURAS

Figura I Percentagem de sucesso dos projetos em 2000.	3
Figura II Encapsulamento – simplificação de um sistema complexo, através da categorização em parcelas menores.	4
Figura III Fases de um projeto	6

1. INTRODUÇÃO

O presente trabalho, procura descrever as atividades que foram realizadas durante o estágio curricular na *Celfocus*, empresa que se dedica ao desenvolvimento de soluções informáticas na vertente das telecomunicações. O estágio teve início a 7 de setembro de 2020 e término a 7 de junho de 2021. Durante os primeiros seis meses, o candidato participou num projeto de integração de uma ferramenta *middleware*, numa metodologia *waterfall*, e nos últimos três meses num projeto de desenvolvimento evolutivo de um *website* numa metodologia *Agile*. A empresa líder das telecomunicações em Portugal, foi o cliente em ambos projetos. O candidato fez parte do departamento responsável pelas integrações de sistemas, nomeadamente OMI (*Order Management & Integration*) composta por cento e trinta e quatro colaboradores. Devido à situação pandémica, o estágio decorreu maioritariamente em regime remoto.

Os objetivos estabelecidos para o estágio, centram-se na aquisição de conhecimentos, e desenvolvimento de competências na área de gestão de projetos, e gestão de equipas *scrum*, através de:

- Suportar o gestor de projeto na preparação de comités operacionais de gestão e assegurar a sua entrega atempada;
- Manter o repositório de documentação atualizado para promover a integração de novos membros;
- Preparar informação sistemática para o gestor de projeto. Extrair e tratar os dados do *backlog*, e controlar a sua evolução;
- Conhecer e integrar-se no modelo operativo da *Celfocus*.

2. REVISÃO DA LITERATURA

2.1. O que é um projeto?

Primeiramente, no contexto organizacional, o trabalho tende a assumir uma de duas categorias, o trabalho operacional e o trabalho em projetos, sendo que o último é aplicado em todos os níveis da organização (PMBOK, 2017).

Em segundo lugar, a origem dos projetos reside nas necessidades empresariais que se encontram por satisfazer (Wysocki, 2018). O *Project Management Institute* (2018) define projeto, como sendo um esforço temporário empreendido, a fim de criar um serviço ou resultado único. Por projeto, entenda-se uma sequência de atividades interligadas, únicas e complexas que têm um propósito ou objetivo em comum, que deverá ser atingido dentro de um tempo e orçamento definidos (Wysocki, 2018).

Existem cinco características que definem um projeto (Villafiorita, 2014):

- **Temporário** – apresenta um princípio, meio e fim. Normalmente, a celebração do contrato – *sign-off* – constitui o início do projeto. Definir o fim torna-se complicado uma vez que após a entrega do *deliverable* final possa existir uma fase de suporte ou trabalho residual por fazer.
- **Output** – Espera-se que no momento final do projeto, seja produzido um entregável, em forma de produto, serviço ou capacidade. Estes terão de ser tangíveis e mensuráveis.
- **Recursos limitados** – O tempo, orçamento e recursos humanos são variáveis escassas num projeto. A consequência, está no facto do gestor de projeto ter de encontrar um equilíbrio entre estas variáveis, tendo em conta as limitações do projeto.
- **Progressivo** - Ao longo do projeto, vão-se tomando decisões que o irão conduzir numa dada direção. No início, as possibilidades de implementação de um projeto são incontáveis. À medida que se vão tomando decisões, o custo de retrocesso aumenta. A liberdade do projeto fica condicionada, até restar uma só possibilidade de implementação do projeto.
- **Exclusividade** – O projeto tem de conferir um ou mais entregáveis exclusivos, algo que proporcione novidade. O que conduz à variável risco,

que se apresenta sob forma de ameaça ou oportunidade. Pelo fator novidade, alguns requisitos poderão não ser compreendidos pela entidade que vai executá-los (Villafiorita, 2014).

2.2. Distinção de Projetos e Projetos Informáticos

A indústria de desenvolvimento de *software* é considerada uma das mais complexas atividades suportadas pelo ser humano (Villafiorita, 2014). Villafiorita (2014) refere a importância que representam para a sociedade, uma vez que estes sistemas são a base de aplicações críticas para a sustentabilidade da sociedade. A nível mundial, anualmente, decorrem milhões de projetos produzindo uma quantia que ronda os 600 mil milhões de dólares americanos (Jalote, 2002). No entanto, grande parte não preenche as expectativas do cliente seja pela qualidade do produto final, seja pela entrega que não correspondeu ao orçamento e o tempo previamente definidos (Wysocki, 2019).

Um estudo realizado em 2000, pela *Standish Group* (2001) – empresa independente de investigação e consultoria em *IT* - concluiu que, apenas 28% dos projetos de *software* são bem-sucedidos e cerca de 23% foram cancelados, enquanto os restantes estavam atrasados, e/ou acima do orçamento, e/ou não preenchiam os requisitos definidos pelo cliente.

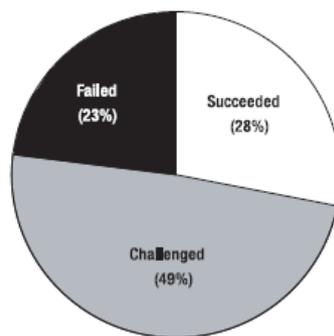


Figura I Percentagem de sucesso dos projetos em 2000.

Fonte: Stepanek (2012) p.4.

Em contraste, os projetos de outras indústrias como por exemplo a construção civil, apresentam taxas de sucesso na ordem dos 94%, segundo a *Engineering News-Record* (2000). Todos os gestores de projetos, partilham os mesmos objetivos no que diz respeito às variáveis tempo, custo e qualidade (Kuehn, 2011).

No entanto dada a natureza do *software*, este apresenta características únicas que irão exigir técnicas específicas de gestão. A falta de adaptação da gestão à realidade tecnológica pode conduzir ao falhanço do projeto (Villafiorita, 2014). Algumas dessas características são:

- **Complexidade** - cada linha de código traduz-se numa instrução, e os sistemas de software compreendem milhares de linhas de código. Para além das instruções, existem interações entre essas instruções. De forma a reduzir a complexidade, os programadores procuram categorizar o código em objetos e componentes - *encapsulation*. (Martin, 2009).

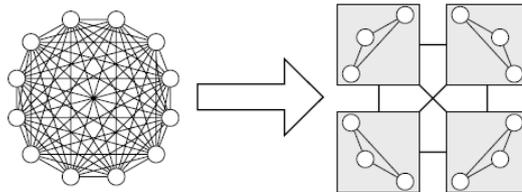


Figura II Encapsulamento – simplificação de um sistema complexo, através da categorização em parcelas menores.
Fonte: Martin (2009) p.11

- **Abstrato** – na programação de software, nada é palpável. Consiste numa junção de comportamentos individuais – linhas de código - tornando-se difícil, para o ser humano, visualizar um elevado número de comportamentos sequenciais e alternativos.

Os projetos de software provêm de necessidades organizacionais, ou seja, necessidades no âmbito do negócio da empresa. Existe um desfasamento no conhecimento entre os programadores e os utilizadores – aqueles que irão utilizar a ferramenta. A complexidade, e abstração no desenvolvimento de *software* impede que os utilizadores consigam definir com precisão os requisitos para o produto, pelo menos até este estar parcialmente concluído (Stepanek, 2012). Para além das características já mencionadas, o facto das tecnologias de desenvolvimento de *software* sofrerem alterações a um ritmo vertiginoso, impossibilita que se chegue a um nível de maturidade na construção de melhores práticas (Stepanek, 2012).

2.3. O PMBOK.

O PMBOK – *Project Management Body of Knowledge Guide* - trata-se de um guia, produto do PMI – *Project Management Institute*. O seu objetivo consiste em colmatar as necessidades das organizações na uniformização das práticas de gestão de projetos, através de guias e melhores práticas (PMBOK, 2017). O seu conteúdo, compreende e sumariza conceitos que foram obtidos de décadas de experiências de vários voluntários que se dedicaram em atualizá-lo regularmente (Ajam, 2018).

A sua primeira publicação oficial foi em 1996, com cerca de 180 páginas, onde refere nove áreas de conhecimento e cinco fases (agrupando um total de 37 processos) num projeto. A cada quatro anos, é lançada uma nova versão do guia. As alterações que foram feitas, estão resumidas na tabela abaixo:

Year	Edition	PG ^a	KA ^b	Processes	Pages ^c
1996	1996	5	9	37	~ 180
2000	2000	5	9	39	~ 210
2004	Third	5	9	44	~ 400
2008	Fourth	5	9	42	~ 460
2013	Fifth	5	10	47	~ 620
2017	Sixth	5	10	49	~ 790

^a Process groups
^b Knowledge areas
^c Cover to cover

Tabela I Mudanças nas edições do PMBOK. Fonte: Ajam (2018), p.12.

As cinco fases descritas no guia, segundo o PMBOK (2017), são as seguintes:

- **Início** – nesta fase, é expectável construir um modelo de negócio, e definir os objetivos, suposições, restrições, orçamento e tempo esperado. Após definidos os pontos anteriores, espera-se que sejam definidos os principais *stakeholders*, assim como os seus respetivos papéis.
- **Planeamento** – a fase de planeamento por norma, consome mais de metade do tempo útil do projeto (Ajam, 2018). O objetivo desta fase consiste na recolha e identificação dos requisitos e especificações técnicas do produto/serviço (PMBOK, 2017). Define-se igualmente, os entregáveis e *milestones* – eventos usados para medir o progresso de um projeto (Pankaj, 2002). É nesta fase que é definido o âmbito do projeto.

- **Execução** – etapa onde a equipa executa o trabalho. A função do gestor de projeto é garantir que permaneça uma eficiente fluidez durante esta fase. É esperado uma constante comunicação com o cliente, com base no plano de comunicação sobre os progressos da equipa.
- **Monitorização** - ocorre em paralelo com a fase anterior, e desta forma é possível garantir que o projeto, e os objetivos do mesmo estão alinhados.
- **Fecho** – corresponde à última etapa do projeto.

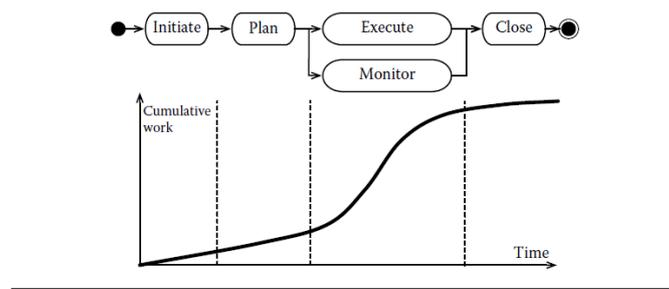


Figura III Fases de um projeto

Fonte: Villafiorita (2014) p.2

Relativamente, às dez áreas de conhecimento:

- **Gestão da Integração** – é nesta fase que se define as seis etapas que constituem o ciclo de vida do projeto: início, planeamento, execução, monitorização e fecho (PMBOK, 2017).
- **Gestão do Âmbito** – compreende o processo de “partir” os requisitos do projeto em tarefas. Para tal, o PMBOK recomenda a ferramenta WBS – *Work Breakdown Structure* – (Stepanek, 2012) diagrama que representa hierarquicamente a decomposição do trabalho a ser executado pela equipa de projeto, com vista a atingir os objetivos definidos, através do alinhamento com os entregáveis. Permite organizar e definir o âmbito do mesmo. Cada nível descendente, representa um maior nível de detalhe no projeto. A estas partições dá-se o nome de *Work Packages* (Norman, Eric S. & Brotherton, Shelly A. & Fried, Robert T, 2008).
- **Gestão do Tempo** – esta área, inclui o planeamento e o controlo. Numa primeira instância, estima-se a duração das tarefas do projeto, assim como o esforço necessário para completá-las. Essa duração, é usada para estimar o

tempo total necessário para a conclusão do projeto. As horas de esforço, permitem também calcular o custo total do projeto (Wysocki, 2019).

- **Gestão do Custo** – envolve o orçamento do projeto, o que significa que é necessário ter ferramentas para estimar os custos do mesmo (Westland, 2019). Desta forma, é possível perceber se os fundos disponíveis são suficientes para cobrir os custos do projeto (Villafiorita, 2014). O custo total do projeto, assim como a *deadline*, são definidos no início do mesmo, não havendo alterações durante o seu ciclo de vida (Wysocki, 2019).
- **Gestão da Qualidade** – um projeto que seja concluído dentro do tempo e orçamento estabelecidos, mas que não preencha os requisitos de qualidade do cliente, é considerado um projeto mal sucedido (Westland, 2019). Compreende três processos:
 - **planeamento** – define-se os *standards* que o produto deverá estar em conformidade;
 - **controle** – inclui ferramentas que permitem monitorizar essa conformidade;
 - **garantia de qualidade** – atividades que garantem que os requisitos de qualidade estão alinhados (Wysocki, 2019).
- **Gestão dos Recursos Humanos** – considerado o recurso mais importante, torna-se crucial manter as pessoas ligadas ao projeto, motivadas e satisfeitas (Weiss, 2014). Adicionalmente, é necessário fazer o rastreio da performance para garantir que o projeto vai de acordo com o que foi planeado (Westland, 2019). Mentoria, liderança e gestão de conflitos, são competências esperadas do gestor de projetos (PMBOK, 2017)
- **Gestão da Comunicação** – a comunicação é reconhecida como fator crítico do projeto (Hermarij, 2016). Deste modo torna-se necessário formular um plano de comunicação com a equipa e os principais *stakeholders*. Neste plano, deverá constar como é comunicada a informação, a quem e com que frequência. (Westland, 2019)
- **Gestão do Risco** – contempla a elaboração de um plano gestão de risco através da problematização, categorização e priorização dos riscos (Hermarij, 2016).

- **Gestão de Stakeholders** – inclui os procedimentos que identificam as pessoas que possam impactar ou ser afetadas pelo projeto (PMBOK, 2017). Torna-se necessário analisar as suas expectativas e grau de envolvimento no mesmo (Weiss, 2014).
- **Gestão da Aquisição** – como gestor de projetos, é comum haver necessidade de adquirir *hardware*, *software* e/ou serviços de outras entidades (Wysocki, 2019). O processo começa por identificar as eventuais necessidades de aquisição no projeto. Inclui os processos relacionados com a negociação e compra dos produtos, assim como serviços a fornecedores externos e subcontratados (PMBOK, 2017).

2.4. Pressupostos do Waterfall

Segundo Stepanek (2012), a metodologia *Waterfall* sugere linhas orientadoras para uma gestão de projetos com sucesso, para uma vasta gama de indústrias. Pelo facto da sua abrangência ser vasta, pode ser necessário reajustar essas mesmas guias orientadoras para contextos específicos, como é o caso de projetos de *IT*.

Com base nas áreas de conhecimento que foram mencionadas anteriormente, a metodologia *Waterfall* parte do pressuposto que o âmbito do projeto possa ser completamente definido, e que seja possível fazê-lo antes do início do projeto. Dada a complexidade, e abstração do *software*, é esperado que na fase de execução sejam encontradas falhas ou incompatibilidades nos requisitos que foram anteriormente definidos (Stepanek, 2012). Numa fase mais avançada, a alteração dos requisitos leva a que o trabalho que foi desenvolvido até então, seja inadequado e inaproveitado. Esta reformulação irá aumentar os custos do projeto.

Assume ainda que as atividades de software possam ser sequenciais, e que é possível alocá-las individualmente aos membros da equipa. Stepanek (2012) acredita que a falta de paralelismo entre as atividades de recolha, *design*, programação e testes limita a comunicação entre os membros da equipa. Por exemplo, na fase de recolha de requisitos não existe espaço para *feedback* ou questões. Durante a fase de desenvolvimento, os programadores não conseguem discutir com os *designers* sobre a arquitetura do *software* em questão, nem os *testers* conseguem refutar os requisitos com o analista de negócio.

Num cenário em que as atividades estão sobrepostas, o aparecimento de defeitos no *software*, por exemplo, são detetados atempadamente, e a reação é mais precisa (Stepanek, 2012).

Outro pressuposto da metodologia *Waterfall*, consiste em assumir que existe uma fórmula exata para obter as estimativas de forma precisa. A estimativa do projeto foi elaborada antes de terem sido identificados os membros da equipa (Stepanek, 2012). Num cenário em que uma equipa nunca trabalhou junta antes, não existe forma de verificar as estimativas em relação aos resultados dos projetos anteriores. Para além de que o PMBOK assume que todos os programadores partilham as mesmas competências de programação.

2.5. *Waterfall*

A metodologia *Waterfall*, ou também conhecida como modelo tradicional, foi adotada pelo setor informático inspirada por outras indústrias, nomeadamente a de construção civil (Muslihat, 2019). Consiste num *breakdown* das atividades relacionadas com o projeto, em iterações lineares e sequenciais com *outputs* definidos para as fases de análise, *design*, desenvolvimento e testes (Davis, B. & Radford, D. 2014). A fase do projeto será formal e oficialmente finalizada, quando os critérios de entrada forem cumpridos.

Esta metodologia prevê um total de cinco fases (Olic, 2017). Sejam elas:

- **Recolha de requisitos** – análise das necessidades de negócio e documentação das especificações do *software*;
- **Design** – criação de diagramas do comportamento do produto com o utilizador, a sua arquitetura e quais as versões a serem utilizadas;
- **Desenvolvimento** – tradução das necessidades de negócio através de código informático;
- **Testes** – realização de testes unitários à fase anterior para aumentar a qualidade do produto final;
- **Implementação** – lançamento do produto final no ambiente produtivo do cliente.

Esta abordagem apresenta várias vantagens, pois a sua natureza estruturada e sequencial, permite uma fácil gestão e execução do projeto (Zinchenko, n.d). Torna-se mais fácil orçamentar e ter a visibilidade do cronograma do projeto na íntegra, e ainda, pela quantidade de documentação que esta metodologia exige é mais fácil fazer o ponto de situação para novos integrantes da equipa.

No entanto, esta abordagem apresenta algumas desvantagens. A natureza lógica, sequencial, formal e rígida poderá dificultar o desenvolvimento, o que limita a sua aplicabilidade para todos os projetos (Davis, B. & Radford, D. 2014) nomeadamente projetos que sejam imprevisíveis, pouco estáveis, e que estejam inseridos num contexto de mudança constante.

Adicionalmente, esta abordagem assume que é possível concluir as fases, que a compõem, a 100% - ou seja não será necessário voltar a rever a fase antecedente. Com base nesta premissa, acredita-se ser possível reduzir o risco, e incertezas relativamente à entrega. A entrega sequencial, apesar de parecer perfeitamente lógica, dá uma falsa sensação de confiança e controlo ao gestor de projeto (Davis, B. & Radford, D. 2014). Neste cenário, o nível de risco aumenta, devido a erros e omissões, tal como a incerteza.

Como já foi referido anteriormente, esta metodologia teve origem na indústria da construção civil (Asmo, 2019). Podemos caracterizar este contexto como sendo uma indústria altamente estruturada, onde pequenos ajustes no plano inicial, ou na fase anterior poderão traduzir-se em custos exorbitantes, inviabilizando a sustentabilidade do projeto (Asmo, 2019). Por esse motivo, a metodologia *waterfall* encaixa perfeitamente nesse contexto. A imensa robustez e inflexibilidade da abordagem, conduz a questões sobre a sua adaptabilidade ao desenvolvimento de sistemas informáticos (Stepanek, 2012).

Os requisitos de aplicações informáticas são alvos de constante revisão e redefinição, por inúmeros motivos. Entre eles destaca-se a complexidade do projeto (Villanova University, 2017). Sabendo que a fase de recolha de requisitos de um projeto é considerada a mais crítica (visto que servirá de base para as fases seguintes), coloca-se a questão se fará sentido a sua aplicação em projetos de natureza informática.

2.6. Agile

A necessidade de mudança, surge da rapidez com que a economia se desenvolve nos dias de hoje. O contexto caótico exige que os produtos sejam entregues ao mercado antes da concorrência, com mais qualidade e mais funcionalidades, de forma a conquistar a quota de mercado (Barbee, 2013). Hoje em dia, desenvolve-se projetos com o pressuposto de uma estabilidade que não existe (Strpanek, 2012). Desta forma, surge a abordagem *Agile*, a fim de dar resposta às lacunas relativas à flexibilidade. É uma abordagem que não advém da teoria, mas sim do processo empírico e experiência de projetos de sucesso (Strpanek, 2012). Trata-se de flexibilidade, habilidade para responder à mudança e construção de uma cultura empresarial, onde os colaboradores são incentivados e motivados a trabalhar e a entregar mais valor (Kotter, J.P. & Heskett, J.L. 2011).

Agile traduz-se, não numa metodologia, mas sim num conjunto de princípios, e alberga várias metodologias como *kanban*, *scrum* e *extreme programming* (Manifesto Agile, 2001). Fundamentalmente, destaca-se por permitir uma maior articulação na gestão do projeto, facilitando as alterações recorrentes. Os princípios do *Agile*, são os que diferenciam da metodologia tradicional (Villafiorita, 2014):

- *Individuals and interactions over processes and tools*
- *Working software over comprehensive documentation*
- *Customer collaboration over contract negotiation*
- *Responding to change over following a plan.*

Caracteriza-se por ser um processo de iterações cíclicas, que no final de cada ciclo – *sprints* - normalmente entre 1 a 4 semanas, é produzido um *deliverable* para o cliente, com o objetivo de obter *feedback* do mesmo. (Barbee, 2013). Estas iterações são incrementais, é entregue uma parte funcional do *software* de forma faseada. As especificações do produto, serão aprimoradas na próxima iteração, conforme as novas exigências do cliente. Assistimos assim, a um *feedback* constante de forma a que as expectativas do cliente fiquem o mais próximo possível daquilo que será o produto final, conduzindo a uma boa execução do projeto (Rubin, 2012).

Resumindo, o forte envolvimento do cliente durante a fase de desenvolvimento do produto desencadeia inúmeras vantagens (Barbee, 2013). Centram-se principalmente

na flexibilidade em reformular os requisitos do produto e na rapidez em que os erros são detetados e corrigidos, reduzindo os custos para o projeto.

2.6.1 Metodologia Scrum

2.6.1.1 Scrum

O *scrum* consiste numa metodologia que se enquadra na *framework Agile* e que surgiu no início dos anos 90, inspirado num ensaio de *rugby* (Corry, 2019). A analogia à modalidade desportiva, está no facto da equipa perseguir um objetivo em conjunto, estabelecendo um ritmo e adaptando-se rapidamente às variações no contexto das jogadas (Villafiorita, 2014). O *scrum* tem como pilares, o conhecimento empírico e o *lean thinking* – traduz-se no foco na entrega de valor, procurando a máxima eficiência na produção, despendendo do mínimo de recursos possíveis (Womack & Jones, 1996).

O *Scrum* oferece um conjunto de práticas e dá espaço para possíveis ajustes (Schwaber, 2004). As suas raízes estão no método de ensino japonês: *Shu Ha Ri. Scrum*, é algo que se aprende praticando o corpo a mente e o espírito, alinham-se através da prática constante.

- **Shu** – o aprendiz está a par de todas as regras. Deve seguir à risca, todas as normas até ser absorvido pelo subconsciente.
- **Ha** – neste estado as regras já são dominadas pelo aprendiz e há espaço para inovar e adaptar.
- **Ri** – o estado em que o aprendiz descarta as regras, e é capaz de criar e desenvolver métodos alternativos para atingir o seu objetivo. As suas ações vão transmitir a essência do *Scrum*, porque já tem esse conhecimento embebido (Sutherland, 2014)

Hoje em dia, o *Scrum* é provavelmente a metodologia *Agile*, mais utilizada. Segundo a VersionOne (2013) o *Scrum* é aplicado em 72% dos projetos *Agile*. As equipas *Scrum*, caracterizam-se por serem independentes, auto-organizadas e reduzidas, distribuindo-se nos seguintes papéis: *Scrum Master*, *Product Owner* e *Developers* (Scrum Guide,2020) não prevendo qualquer tipo hierarquia.

2.6.1.2 Scrum Master

O *Scrum Master*, tem a responsabilidade em garantir que os processos de *Scrum* estão a ser seguidos pela equipa, assim como a remoção de obstáculos que interfiram com a eficácia da mesma (Malsam, 2018). O *Scrum Master*, deverá facilitar todas as reuniões, garantido a transparência e auxiliar a equipa na melhoria de eficiência do trabalho. Deve guiar a equipa a um melhoramento contínuo (Sutherland, 2014). Não compete ao *Scrum Master* delegar tarefas, nem ser responsável pelo planeamento ou pela qualidade do produto final. No entanto, é da sua responsabilidade que os objetivos sejam cumpridos num dado período.

2.6.1.2 Product Owner

O papel do *Product Owner*, é maximizar o valor do produto, através da recolha de novas funcionalidades (Scrum Org, 2021). É a pessoa com maior visibilidade sobre o projeto e o negócio e representa outros *stakeholders* do projeto. É ainda responsável pela população e priorização das tarefas que estão no *Product Backlog* (Schwaber, 2004).

2.6.1.3 Developers

Os *Developers*, ou *Creators*, são uma equipa normalmente constituída por programadores, *testers* e analistas funcionais, responsáveis por criar os entregáveis do projeto (Schwaber, Sutherland, 2020). É esperado que percebam os requisitos de negócio propostos pelo *Product Owner*, assim como fornecer estimativas para cada um desses requisitos de forma a quantificar o trabalho e esforço exigido.

2.6.1.4 Artefactos e Eventos do Scrum

A dinâmica *scrum*, desenvolve-se em iterações que se dá o nome de *sprints*. Estes *sprints*, albergam os artefactos (*Product Backlog*, *Sprint Backlog* e *Increment*) e os eventos (*Sprint Planning*, *Daily Scrum*, *Sprint Review*, *Retrospective*) (Drumon, 2018).

2.6.1.5 Product backlog

Este artefacto do *scrum*, permite delinear o âmbito do projeto. Nesta lista estão compilados os requisitos do produto (Radigan, n.d) organizados por ordem de prioridade.

É nesta lista, que o *Product Owner* insere as *user stories* – explicação de uma especificidade de *software* escrita sob a perspetiva do utilizador final (Cohn, 2004) – acrescentando valor ao produto. Existe ainda um *subset* desta lista dedicada ao *sprint* – dando-lhe o nome de *Sprint Backlog*.

2.6.1.6 *Sprint Backlog*

O *Sprint Backlog* consiste numa lista de tarefas relativas à implementação de *user stories* de forma a atingir o *Sprint Goal* – objetivo definido para o *sprint*. No início do *sprint* são selecionadas as *user stories* que constituem o *Product Backlog*, para serem desenvolvidas durante o *sprint* em questão (Schawber, 2004).

2.6.1.7 *Increment*

No final de cada *sprint* espera-se que seja entregue uma funcionalidade do produto. Esta funcionalidade terá de ser testada e bem estruturada, uma vez que poderá ser logo integrada na solução final desejada pelo *Product Owner* (Schawber, 2004).

2.6.1.8 *Sprint Planning*

No início do *sprint*, ocorre o *sprint planning* que tem como objetivo planear quais as *user stories* que o irão fazer parte do *Sprint Backlog*. Esta cerimónia prevê que estejam presentes o *Scrum Master*, *Product Owner* e a equipa de *Developers* (Scrum Guide, 2020).

2.6.1.9 *Daily Scrum*

Segundo o Scrum Guide (2020), a *Daily Scrum*, consiste numa sessão de quinze minutos com a *Scrum Team* (*Product Owner*, *Developers* e *Scrum Master*). O objetivo da reunião é perceber o estado atual das tarefas, planear ações para o dia de forma a progredir em direção ao *Sprint Goal*. É uma sessão dedicada a melhorar a comunicação, identificar impedimentos e remover necessidade de estabelecer outras reuniões.

2.6.1.10 *Sprint Review*

Aquando do término do *sprint*, ocorre o *Sprint Review* cujo objetivo é demonstrar aos principais *stakeholders*, o trabalho que foi realizado (Scrum Guide, 2020). É apresentado o progresso do produto respeitando o valor da transparência de informação da metodologia *scrum* (Drumon, 2018). Espera-se que exista *feedback* do cliente no final desta sessão, possibilitando o ajuste do *product backlog*.

2.6.1.11 *Retrospective*

Por último, o *sprint retrospective* é a sessão que dá espaço à equipa um momento de introspeção e análise ao *sprint* que decorreu de forma a perceber quais os pontos fortes e os de melhoria (Alfonso, 2021). Este evento deve ser um espaço seguro para que as pessoas partilhem opiniões sinceras sobre o projeto. Normalmente esta sessão é conduzida pelo *scrum master*, e todos os elementos da equipa devem intervir. No final da reunião, é esperado uma documentação sobre pontos de mudança que foram discutidos (Scrum Guide, 2020)

3. CARACTERIZAÇÃO DA ORGANIZAÇÃO

3.1 *Novabase*

A *Novabase* é uma empresa que se dedica ao desenvolvimento de software “à medida”. Pioneira, em 1994, no mercado de desenvolvimento de aplicações informáticas em Portugal. A sua estratégia passou por posicionar-se como integrador de sistemas. Com o passar dos anos, foi aumentando a abrangência da sua oferta, construindo uma rede de empresas especializadas, como a *Celfocus*, no setor das telecomunicações, e outras dedicadas aos setores de energia, serviços financeiros, governo, e por fim, transportes.

Passados 28 anos da sua criação, a *Novabase* torna-se líder português em Tecnologias de Informação, estando cotada na *Euronext Lisbon* desde 2000.

Estão presentes, desde 2011, na Europa, Ásia, África e América. Constituída por um total de dois mil colaboradores, de dezassete nacionalidades. A visão da *Novabase* é tornar a vida das pessoas e das empresas mais simples e mais feliz, através da utilização da tecnologia.

3.2 Celfocus

Fundada em 2000, resultante de uma *joint venture*, entre a *Novabase* e a uma empresa líder no setor das telecomunicações em Portugal. Inicialmente, focou-se no desenvolvimento de soluções *high-tech* e produtos dedicados ao setor das telecomunicações. Nos dias que decorrem, a *Celfocus* tem projetos no setor da banca e infraestruturas.

Com projetos em mais de 25 países, a *Celfocus* espalhou os seus escritórios um pouco por toda a Europa, e Médio Oriente.

4. CARACTERIZAÇÃO DO ESTÁGIO

4.1 PROJ-1 – Introdução

O candidato foi incluído no *PROJ-1* – plataforma *middleware* integrada nos sistemas da Empresa de Telecomunicações, que faz a gestão e monitorização entre os mesmos – que consistia em fazer um melhoramento da ferramenta, com o objetivo de automatizar/simplificar o processo de instalação e uniformizar a solução entre as diferentes *OpCos* - termo utilizado para referir-se à Empresa de Telecomunicações local - nomeadamente nos mercados de Espanha, Itália e Grécia. Esta iniciativa, foi motivada pelas crescentes entropias que a ferramenta proporcionava até à data, gerando custos inesperados para o cliente.

O projeto foi dividido em várias *workstreams*, e cada uma composta por uma equipa de *developers*, um arquiteto e um *Project Manager*. Dentro de cada *workstream*, existem sete *workpackages* que correspondem a uma etapa no projeto apresentando critérios de entrada e de aceitação. Importa ainda referir que este desenho e esquematização de trabalho foi feito em parceria com Empresa de Telecomunicações.

A primeira *workpackage* (*WP#0*) corresponde à fase que é transversal a todo *PROJ-1*. Consiste num projeto piloto em parceria com a Empresa de Telecomunicações de Portugal, que ocorreu num momento anterior à data de entrada do candidato na *Celfocus*. Nesta fase, foi definido uma arquitetura *standard*/base da solução e preparou-se toda a documentação que suporta as especificações do produto. Esta *WP* surgiu como

solução base para as restantes *OpCo's*. O processo demorou mais tempo do que o previsto, o que levou a que estivessem a decorrer duas *WP* em simultâneo.

Numa etapa seguinte, surge a *WP#1*, que consiste no *Due Diligence* – investigação e revisão dos requisitos - fase do projeto em que os *Project Managers* da *Celfocus* e os *SPOC's* da Empresa de Telecomunicações com as suas respetivas equipas, discutem as customizações que serão desenvolvidas no produto do *PROJ-1*. As *workpackages* subsequentes, dizem respeito ao estabelecimento de conectividades, infraestrutura, testes, implementação, aceitação e por fim suporte, correspondente à ultima fase do projeto. Existem *entry criterias* assim como *exit criterias*, que por vezes estão associados a um conjunto de *deliverables*, e só quando estas encontram-se concluídas, é que é permitido avançar para a fase seguinte, refletindo a tipologia *waterfall*.

A entrada do candidato, coincidiu com o início de uma *workstream* para a Empresa de Telecomunicações sediada em *UK*, e a sua primeira tarefa consistiu em elaborar uma agenda para o *kick-off* do projeto e definir *ground rules* para o desenvolvimento das reuniões posteriores. No dia 14 de setembro, a data oficial de arranque do projeto, foram agendadas três sessões de *workshop*, de forma a dar a conhecer o produto ao cliente incentivando a discussão de forma a levantar novos requisitos. É da responsabilidade dos *Project Managers*, anotar todas as subjeções apresentadas pelo cliente, desde novas funcionalidades do produto a compatibilidades do sistema. Estas subjeções, foram apresentadas à equipa para serem debatidas, com o propósito de ajustar a solução do *PROJ-1* às necessidades do cliente.

Após as sessões de *workshops*, foram agendadas reuniões diárias com o cliente de forma a clarificar os requisitos do mesmo, proporcionando também um acompanhamento mais próximo. Foram debatidos assuntos relacionados com a gestão do projeto, evitando entrar ao detalhe sobre questões técnicas. Estas reuniões foram divididas entre a *WP#0* e a *WP#1*, visto que as etapas estavam a decorrer paralelamente e existiam dependências entre as mesmas. As questões técnicas foram endereçadas para as reuniões semanais com a equipa técnica do cliente. Para além das reuniões com o cliente, existiu a necessidade de agendar reuniões internas com o objetivo de monitorizar as tarefas de cada membro da equipa, sendo uma prática frequente em projetos *Agile*.

4.2 PROJ-1 – Tarefas desempenhadas

Embora o modelo de entrega deste projeto seja uma tipologia *Waterfall*, foram adotadas algumas práticas de *Agile*. Foi utilizado o quadro *kanban*, que permite o rastreamento das tarefas através da sua repartição em três colunas : “to do”; “in progress”; “done” , através da ferramenta *Jira* – *software* dedicado à gestão de projetos.

- **To do** – estão listadas todas as tarefas inerentes ao projeto;
- **In progress** – a pessoa a quem foi atribuída tarefa deve movê-la do “*To do*” para este campo, aquando a execução da mesma;
- **Done** – As tarefas são movidas para *done* quando concluídas.

Neste sentido, o candidato ficou responsável por garantir o bom funcionamento da plataforma *Jira*. Este problematizava as tarefas, que provinham das reuniões com o cliente, e listava-as no campo “*To-do*”. Alocava ainda as tarefas aos membros da equipa e junto dos mesmos tentava definir datas de conclusão.

4.3 PROJ-1 – Alocação de Recursos

Durante o projeto, a equipa foi surpreendida com a ausência de um elemento sénior. Este tinha marcado férias, numa altura crítica do projeto e a notícia foi transmitida apenas com uma semana de antecedência. Surgiu assim, a necessidade de criar um mapa de alocação de pessoas com regras. Na eventualidade de marcação de férias, as mesmas deverão ser informadas com trinta dias de antecedência. Aplicou-se também FCFS – *First come, First serves* - uma vez que se procurou evitar que mais do que um membro da equipa tirasse férias em simultâneo. Em caso de conflito de agenda deveria ser solucionado entre os intervenientes seguindo as diretrizes. Não sendo possível, seria escalado ao órgão de gestão *PMs* e *Programme Manager*. Periodicamente, foi realizado uma projeção a sessenta dias com a alocação das pessoas, sendo endereçado ao *Programe Manager*, pois este tinha uma maior visibilidade das inúmeras iniciativas do programa do *PROJ-1*. Desta forma, o candidato participou na elaboração do mapa, mantendo-o atualizado, assim como agendar reuniões periódicas para discutir a alocação das pessoas.

4.4 PROJ-1 – Documentação

De forma a garantir o correto versionamento da documentação, assim como evitar o conflito das mesmas o candidato coordenou com os *PM's* dos projetos de *WP#0* e *WP#1* um conjunto de práticas e métodos de gestão adequados aos respetivos documentos. Foi criada uma folha de *excel* que permitiu controlar a atribuição da nomenclatura e respetiva versão. Estes conflitos eram constantes visto que todas as *workstreams* produziam a respetiva documentação com base na documentação base.

Durante a fase de *Due diligence*, surgiram pedidos para além do que foi estabelecido como âmbito do projeto, outro fator que influenciou o seu atraso. O cliente solicitou adicionar ao produto do *PROJ-1* uma ferramenta que fizesse uma auditoria entre as duas base de dados, de forma a garantir a integridade dos dados. Para conservar a boa relação com o cliente, procurou-se internamente se a *Celfocus* já tinha aplicado algo similar em outros projetos. O candidato, encarregou-se de fazer essa procura: consultou os *PMs* que tinham gerido os projetos em Itália e Espanha, e após reunida toda a informação, encontrou-se uma solução que satisfiz a necessidade do cliente.

4.5 PROJ-1 – Formação e Desenvolvimento Profissional

Paralelamente ao projeto, foram agendadas reuniões de *feedback* semanais com o gestor de projeto. Estas sessões tinham como objetivo expor as dúvidas do candidato, assim como os seus objetivos, quer fossem de natureza profissional ou pessoal. A *Celfocus* desenhou ainda um programa de crescimento contínuo e *feedback* constante entre *PT's* – colaboradores *Celfocus* com estatuto sénior na empresa - e *Talenter's* – colaboradores com perfil júnior - que consistia em sessões mensais sobre a *performance* do candidato. Ainda no mesmo contexto, existiam reuniões periódicas com a orientadora de estágio, com o intuito de acompanhar o progresso do candidato. Por fim, existia também um fórum de *Center of Excellence*, onde mensalmente eram debatidas questões relacionadas com as melhores práticas na gestão de projetos.

O candidato foi inscrito em inúmeras formações no âmbito de gestão de projetos, tais como:

- **Scope Management** – a formação focou-se nas melhores práticas da gestão do âmbito do projeto, em contextos reais;

- **Budget Control** – gerir e adquirir conhecimento sobre o manuseamento da folha de *excel Celfocus*, que retrata a estimativa de esforço e a elaboração de propostas;
- **Stakeholder Management** – identificar todas as partes interessadas do projeto e gerir e planear a comunicação com os mesmos;
- **Project Management with MS Project** - formação lecionada por uma entidade especializada e certificada em formação de *software Microsoft Project*.

Adicionalmente, o candidato foi exposto a formações de *soft skills*, como:

- **Effective Communication** - onde os formandos são submetidos a um teste comportamental – *DISC* – que consciencializa sobre o discurso a adaptar dependendo do tipo de perfil de pessoa a quem nos dirigimos.

Para aprofundar os seus conhecimentos sobre a metodologia *Agile*, o candidato foi inscrito em formações como:

- **SAFe Foundations** – consiste na escalação da metodologia *Agile* em toda a organização;
- **Agile & DevOps Foundations** – esta formação procurou explorar todas as metodologias dentro do *Agile* com especial foco no *Scrum*.

Por fim, o candidato participou numa formação de **Set Your Goals**, que auxilia na definição dos objetivos utilizando a metodologia *S.M.A.R.T* – *Specific Measureable Achievable Relevant Time-Bound*.

4.6 PROJ-1 – Fases do projeto

Durante a primeira fase de *Due Diligence* do *PROJ-1*, algo que causou constrangimento foi o desajuste de expectativas entre o cliente e o arquiteto *Celfocus*. Este último, considerava que o produto que se estava a desenvolver para o mercado de *UK* seria bastante similar ao que foi realizado no projeto piloto. O desfasamento destas expectativas foi uma das causas que levou à incorreta definição dos requisitos, conduzindo a um esforço adicional não planeado. Coube assim ao candidato, juntamente com o seu *PM*, agendar reuniões com o Arquiteto com intuito de perceber o esforço que

esta mudança iria requerer, e qual a melhor estratégia a definir, permitindo poupar esforços e mitigar os riscos.

Dada a complexidade do projeto, e uma vez que este já excedia o tempo esperado de conclusão, surgiu a necessidade de centralizar toda a informação, e garantir que todos os intervenientes do projeto estavam alinhados. O candidato foi desafiado a elaborar um documento com base no princípio *MECE* (*Mutually Exclusive and Collectively Exhaustive*) - consiste em agrupar toda a informação num único documento - *SSOT* (*Single Source of Truth*). Este documento, compreendia todos os requisitos e especificações, parametrizações, compatibilidades, versões de software, condições de suporte, assim como o desenho arquitetónico do produto do *PROJ-1*, sendo apresentado regularmente ao cliente.

Um dos *deliverables* que encerra a fase de *Due Dilligence* é o *Project Plan*, baseado num plano previamente elaborado com o cliente durante a *WP#0*. Este precisou de ajustes, dado que o projeto sofreu algumas alterações incentivadas pelas especificidades do mercado. Foi utilizado o *Microsoft Project* para esquematizar as tarefas a executar. Deste modo, o candidato procurou redefinir datas, ordem cronológica das tarefas e a alocação das pessoas às mesmas. Para além da integração do sistema do *PROJ-1*, a *OpCo* de *UK* solicitou à *Celfocus* que procedesse à criação de serviços adicionais, os quais não estavam contemplados no plano do projeto.

Uma vez esquematizadas as tarefas a ser realizadas durante a fase de desenho e implementação, foi necessário reunir a equipa técnica para se discutir o esforço para cada uma das tarefas. Este esforço foi medido através da métrica *man-days* – quantidade de trabalho realizado, em dias, por uma pessoa durante o período laboral – que permitiu calcular o custo total do projeto.

Com o arranque do *WP#1* de Portugal, decorriam três *workstreams* em paralelo. Os recursos encontravam-se sobre alocados, devido a complicações no projeto da *WP#0*. Neste contexto, o *Programme Manager*, sugeriu colocar em *stand-by* o projeto de *WP#1* de *UK*, para que se canalizasse todos os esforços afim de solucionar os problemas que estavam a surgir no *WP#0*. Foi delegado ao candidato elaborar um num plano de contingência, afim de mitigar os riscos e reduzir o impacto no mesmo.

Dado o cenário de incerteza sobre a continuidade do projeto e a saída do *PM*, tornou-se necessário elaborar um plano de comunicação ao cliente, preservando a confiança e a saúde do projeto. O processo de passagem para o novo *PM*, consistiu em compilar toda a informação disponível do projeto, formular uma lista dos principais *stakeholders*, divulgar o documento *SSOT* com as características do produto e apresentar os elementos da equipa. Agendou-se sessões diárias, com o novo *PM*, de forma a que a transição não prejudicasse o desenvolvimento do projeto.

O candidato e o novo *PM* prepararam e apresentaram a proposta final ao cliente. As estimativas de esforço foram realizadas pela equipa de desenvolvimento, categorizadas por requisito do cliente e por perfil profissional. A *Celfocus*, disponibiliza um ficheiro de *excel* que auxilia na elaboração da proposta, no qual se preenche informações sobre o projeto, tais como: número de departamentos e entidades externas envolvidas, modo de entrega (*turnkey* – âmbito fechado - ou *time & materials* – âmbito em aberto), antiguidade do cliente e duração do projeto. A folha de *excel*, calcula o risco do projeto que representará um valor de reserva na proposta final. Existem outros campos que são preenchidos nomeadamente despesas extras, como viagens, estadias ou licenças de software e a margem lucrativa do projeto. Atribui-se a cada perfil um esforço estimado o que permite calcular o valor comercial do projeto. Por ter sido estipulado um valor com o cliente na fase contratual do projeto, mesmo que informalmente, e a folha de *excel* ter excedido em milhares de euros esse montante, surgiu a necessidade de reajustar os parâmetros da ferramenta.

Numa fase posterior, foi proposto ao candidato fazer a gestão de um *change request* – pedido de alteração formal a um produto ou sistema. A Empresa de Telecomunicações sediada na Grécia propôs desenvolver serviços adicionais na plataforma, serviços estes que visam mudar a forma como os sistemas comunicam entre si, através de novos protocolos. A equipa era composta pelo candidato, como gestor de projeto, e quatro *developers*. Numa primeira instância, o candidato encarregou-se em alinhar o esforço necessário para a tarefa em questão e formalizar a *work-order* – documento formal, em que o cliente expressa a vontade de solicitar um serviço ao vendedor. Por existir uma resposta rápida às suas estimativas e orçamento disponível, antes do encerramento do ano fiscal, o cliente procedeu ao pedido de implementação de dois serviços extra. Assim, o candidato simplificou a gestão dos três pedidos e

uniformizou-os numa única *work-order*. A preocupação inicial foi estimar o esforço em *man-days* para conseguir apresentar o valor real do projeto e estabelecer com o cliente o valor que iria consistir como *buffer* – margem para cobrir o risco e suporte no caso de eventuais *bugs*. A vantagem do projeto, é não existir *burn-rate* de recursos, uma vez que não havia pessoas dedicadas única e exclusivamente ao mesmo.

4.7 PROJ-2 – Introdução

No sétimo mês de estágio, iniciou-se o período de formações no âmbito do *Agile*, tais como:

- ***Achieving an Agile Mindset*** – esta formação teve como base o documento *Agile Manifesto* o qual procura entender os princípios e valores Ágeis e interligá-los com as boas práticas de gestão de projetos.
- ***Become Familiar With Scrum*** – dentro do *Agile*, existe uma metodologia denominada de *Scrum* e o foco nesta formação, através do *Scrum Guide*, foi perceber qual o seu propósito, a sua definição, os seus valores e quais os papéis existentes.
- ***Scrum Master Skills*** - esta formação centra-se nas principais tarefas de um *SM* dentro de uma equipa *Scrum*, oferecendo ferramentas de melhoria da eficiência, liderança e comunicação da equipa. O último módulo da formação refere como escalar o *Agile* para múltiplas equipas através de uma metodologia *SAFe*.

O candidato cedeu o projeto que esteve a gerir para integrar um novo desafio. O *PROJ-2* foi alvo de inúmeros elogios por parte do cliente, e foi eleito como equipa exemplo em relação à aplicação do modelo *Scrum*. A equipa do *PROJ-2*, é composta por 16 elementos distribuídos por:

- **1 *Scrum Master*;**
- **1 *System Architect*;**
- **1 *UX/UI Design*;**
- **2 QA's – *Quality Assurance*;**
- **2 *Business Analyst*;**
- **9 *Developers*.**

O projeto teve início em Abril de 2019, com o propósito de melhorar e acompanhar o desenvolvimento do *website* de *IoT* da Empresa de Telecomunicações.

O *PROJ-2* surgiu da fusão dos projetos *PROJ-2.1* e *PROJ-2.2*. O primeiro focava-se na gestão do *front-end* – interface do *website* - utilizando a ferramenta *AEM* – *Adobe Experience Manager* - e a equipa do *PROJ-2.2* estava focada no desenvolvimento do *back-end*, a qual utilizava a ferramenta *Magento*, que consistia numa plataforma intermodal que centralizava a gestão do *website* num único catálogo. O *website* variava consoante o país, visto que cada mercado tinha os seus próprios produtos e preços. O âmbito do projeto estava limitado ao *website* de *UK*, e devido ao sucesso do *PROJ-2*, este foi gradualmente conquistando responsabilidades e áreas de ação. As responsabilidades iniciais incidiam sobre o *Homepage*, e com o decorrer do projeto assumiram o *PDP* (*Product detail page*) - consiste na página do produto que contém toda a informação do mesmo: o tamanho, cor, preço, avaliações e outro tipo de informação que seja relevante para os consumidores. A entrada do candidato para o *PROJ-2*, coincidiu com o momento em que a equipa herdou a componente do *checkout* do *website* - a última fase do processo de compra em *e-commerce*. Por influência do *Brexit*, o *delivery* também estava a sofrer alterações. A empresa encarregue do transporte dos produtos da Empresa de Telecomunicações, para casa dos clientes em *UK* deixou de ser viável economicamente, tendo de ser substituída. Deste modo, o *PROJ-2* realizou a migração e integração dos sistemas entre o *website* e a nova transportadora.

4.8 *PROJ-2* – Ferramentas Agile

Nas primeiras semanas deste projeto, o objetivo foi conhecer as pessoas que faziam parte do mesmo e quais as suas funções. Era importante estar informado da sequência das entregas, assim como dos procedimentos que tinham de ser tomados em cada uma das fases. Era utilizada a ferramenta *Jira* para manter a organização do trabalho, no entanto a mesma estava sediada no ambiente do cliente, não existindo liberdade de customizá-la. O *Jira* estava organizado da seguinte forma:

- **To do** – neste campo estão listadas tarefas do *sprint backlog*, divididas em *User Stories* e/ou *Spikes* - trabalho de investigação - e *bugs* que possam surgir durante a fase de testes.

- **Blocked** – este campo é reservado para tarefas bloqueadas por dependências externas.
- **In development** – alberga todas as tarefas em fase de desenvolvimento
- **Code-review** – os desenvolvimentos inseridos no “*in development*” são revistos pela restante equipa de desenvolvimento, de forma a garantir a qualidade do código.
- **QA testing** – neste campo as tarefas estão em ambiente de desenvolvimento e são utilizadas para realizar os testes funcionais do *website* antes deste ir para o ambiente produtivo - *website* que está disponível para o utilizador final.
- **Ready to deploy on pre-prod** – Após a fase de testes no ambiente de desenvolvimento, estas tarefas encontram-se em espera para implementação no ambiente produtivo do cliente.
- **Pre-prod testing** – Após a implementação, o *website* encontra-se em modo “freeze”, durante o qual é realizado um reforço dos testes pelos QA’s da Empresa de Telecomunicações. Existe duas fases de testes, realizadas pela *Celfocus* e pela Empresa de Telecomunicações, de forma a partilhar responsabilidades caso o *website* final apresente *bugs*.
- **Done** – Quando as tarefas preenchem todos os requisitos definidos no *DoD – Definition of Done* – a área de intervenção da *Celfocus* termina, uma vez que o responsável por fazer a implementação para o ambiente produção é o cliente.

4.9 PROJ-2 – Principais atividades

O candidato esteve encarregue do processo de implementação no ambiente *pre-prod*, tendo autonomia para gerir com o cliente a melhor altura para fazê-lo. Após a confirmação do cliente, o candidato alinou com a equipa os preparativos para a implementação, e garantia a atualização do *Jira*. Adicionalmente, criou as *Releases Notes* e respetivo versionamento, que consiste num comprovativo das novas *features* ou *fixes* que foram implementadas

Gradualmente o candidato foi assumindo outras responsabilidades no projeto. Conduziu as *daily scrums*, com o objetivo de facilitar a reunião, criando um espaço para partilha de dificuldades e/ou obstáculos. Planeava igualmente as tarefas diárias a ser

executadas. Por vezes, era útil falar individualmente com os elementos da equipa para perceber se os constrangimentos que mencionaram na reunião já se encontravam resolvidos.

Foi ainda desafiado a liderar as reuniões de retrospectiva. A cada duas semanas, no final do *sprint*, preparou a sessão através da ferramenta *Trello*, que permitia criar *boards* para cada sessão de trabalho, e personalizá-lo com colunas. Procurou manter a seguinte organização:

- ***Previous Actions Items*** - era questionado o estado das tarefas que foram identificadas de sprints anteriores. Caso não estivessem concluídos, permaneciam nesta coluna para a próxima reunião de retrospectiva.
- ***What could be better*** - foi reservada esta coluna para os membros da equipa partilharem as suas preocupações relativas ao *sprint* que decorreu. Este é o ponto que origina mais discussão e que conduz a novos *action itens*. Apesar de enriquecedora a presença do cliente, a mesma condicionou a discussão de conflitos internos, tendo sido necessário a criação de outra sessão de retrospectiva restrita à equipa *Celfocus*.
- ***What I loved*** – essencial para amenizar o ambiente entre a equipa. As pessoas são parabenizadas pela sua performance durante o *sprint*, e são referidos momentos positivos da equipa.
- ***Action Items*** – são registados os pontos de melhoria e ações que deverão ser aplicadas durante o(s) próximo(s) *sprint(s)*.

De forma a complementar a sessão, o candidato preparou questionários de satisfação da equipa e do cliente. O questionário direcionado para a equipa, tinha as seguintes questões:

- *How happy are you with the last sprint teamwork?*
- *How happy are you with the current Client/Customer?*
- *How happy are you with working & learning at Celfocus?*
- *In my team, I feel fit and strong*
- *I am proud of the work that I do for my team*
- *I am enthusiastic about the work that I do for my team*
- *I find the work that I do for my team of meaning and purpose*

- *What would increase your happiness?*
- *Any recommendations/suggestions?*

Cada elemento da equipa teve de classificar as questões numa escala de 0 a 10, em que “0” correspondia a “Hell no!” – Nada satisfeito – e “10” “*Super Happy!*” – Muito Satisfeito. A ferramenta permitia detetar sinais de insatisfação da equipa em geral e a título individual. Inicialmente o questionário foi realizado num ficheiro *Excel* partilhado, no entanto, o candidato acreditou que a falta de privacidade nas classificações iria influenciar as respostas da equipa. Nesse sentido, utilizou-se o *Microsoft Forms*. Ao identificar-se os sinais de insatisfação da equipa, foi possível trabalhar nesses pontos antes que compromettesse a qualidade de entrega.

O formulário direcionado ao cliente, apresentava a seguinte estrutura:

- *How happy are you with the speed & efficiency at which we are able to respond to your requests?*
- *How happy are you with our attention to detail and thoroughness?*
- *How happy are you with the extent to which we help you learn and provide recommendations that are in the best interest of your organization?*
- *Does our service live up to your expectations during the past iteration?*
- *How does our service during the past iteration compare to your ‘ideal’ service?*
- *How likely would you be to recommend our company’s services to a friend or a colleague? (based on current sprint)*
- *How happy are you with how collaborative and proactive we are in the way that we work with your organization?*
- *Overall, how satisfied are you with our company and services so far?*

A escala de classificação foi a mesma utilizada no questionário para a equipa. As respostas ao formulário refletiram a satisfação do cliente e o sucesso da equipa *Celfocus*.

4.10 PROJ-2 – Documentação

A equipa foi desafiada pelo cliente a preparar um plano de implementação do *website*, semelhante ao que foi desenvolvido para *UK*, para a Hungria e a Roménia.

O candidato recolheu, junto da equipa, dados relativos às atividades a desenvolver e respetivo esforço. Com esta informação construiu uma *timeline* e *milestones* importantes e desenvolveu um *pptx*, de forma a apresentar os resultados ao cliente.

A boa performance da equipa *PROJ-2*, e a validação por parte do cliente, levou a que fosse submetida a processos de auditoria interna, servindo de exemplo para outras equipas *Celfocus*. Dessa auditoria, surgiram *KPIs* – *Key Performance Index* - que deveriam ser cumpridos. Um deles, consistia na formalização do processo de implementação, e uma vez que o candidato era o responsável pelo mesmo, encarregou-se de garantir que a equipa seria bem sucedida nesse ponto. Redigiu um documento que contemplava informações sobre as datas previstas das implementações, a estratégia a ser seguida, a metodologia, a tecnologia utilizada, os testes necessários e por fim os obstáculos conhecidos. O documento estava esquematizado da seguinte forma:

- ***Deployment Schedule and Resources*** – O processo de implementação começa no ambiente *pre-prod*. Após a aprovação e validação das *US* foi realizada uma nova implementação no ambiente produtivo (estado final). Este último evento foi realizado pelo cliente, com o auxílio técnico da *Celfocus*. Idealmente é realizado a cada dois dias.
- ***Deployment Risks*** – Verificou-se alguns riscos durante o processo de implementação. A mesma foi erradamente feita para o ambiente produtivo, não sendo sujeita a testes, o que conduziu à indisponibilidade do *website* e ao condicionamento das faturas naquele dia. Elaborou-se um plano de mitigação de risco, que consistiu em adicionar passos extra antes do início da implementação, sempre sujeita a aprovação do cliente.
- ***Services Impacted*** – O objetivo da implementação é introduzir novas funcionalidades e especificações no *website* de *IoT* da Empresa de Telecomunicações nos sete mercados disponíveis.
- ***Possible Issues and Conflicts*** – Foram identificados os seguintes problemas, com os respetivos planos de resolução:
 - Quando existem implementações de *US* de *AEM* e de *Magento* de forma isolada, é necessário limpar a *cache*. Quando se procede à implementação no ambiente produtivo, é necessário instruir a *Adobe* (empresa responsável pela implementação em produção) dos passos a seguir para limpar a *cache*.

- Durante o processo de implementação pode existir um *crash* no ambiente. Neste caso, é necessário formalizar um *ticket* de urgência à Adobe.
- Há erros que apenas são detetados após a implementação. Para tal existe o método de *Rollback*, que permite restaurar a versão anterior do ambiente.

5. CONCLUSÃO E ANÁLISE CRÍTICA E INTERPRETATIVA DO ESTÁGIO

Tendo em consideração a revisão da literatura, e a experiência que o candidato adquiriu durante o estágio curricular na *Celfocus*, é possível fazer uma ponte entre conceitos, nomeadamente a aplicação das metodologias de gestão de projetos informáticos.

No *PROJ-1*, esteve presente o modelo *waterfall* detetável através da esquematização da entrega da plataforma *middleware* em fases lineares e sequenciais. O projeto reservou momentos distintos para a recolha de requisitos, estabelecimento de conectividades, desenvolvimento, testes e aceitação. A definição de critérios de entrada e saída, também constituiu uma prática comum deste modelo, evitando que exista sobreposição de fases. Apesar do projeto apresentar características claras do modelo *waterfall*, o candidato verificou práticas *Agile* a serem aplicadas, tais como reuniões internas diárias, com o uso do quadro *kanban*.

O modelo *waterfall* sugere uma fase dedicada à recolha de requisitos junto do cliente, não sendo possível redefini-los noutras fases do projeto. Essa inflexibilidade, conduziu a complicações no mesmo. Assumiu-se erradamente ser possível recolher todos os requisitos do cliente antes de começar o desenvolvimento do mesmo. O prazo destinado a esta fase foi excedido, atrasando o início da fase seguinte e comprometendo todo o projeto.

Julgou-se que a *WP#0* estava terminada, dando início à fase seguinte. No entanto, no decorrer da *WP#1* surgiram complicações relacionadas com a fase anterior. O modelo *waterfall*, assume que as fases concluídas não são suscetíveis a alterações. Este constrangimento levou à ausência de um membro sénior em auxílio da *WP#0*, o que condicionou a progressão do *WP#1*, salientando a fragilidade de uma equipa não autossuficiente. O modelo *waterfall* assume que todos os membros da equipa têm o mesmo nível de experiência.

No *PROJ-2* foi adotada uma metodologia *Agile* para gerir o desenvolvimento do *website*. As entregas das funcionalidades do mesmo foram realizadas em iterações cíclicas, a cada duas semanas. Procurou-se maximizar o valor dessas entregas através de um *feedback* constante do cliente, e também com foco na motivação da equipa.

O candidato verificou como a envolvimento do cliente foi crítica para o sucesso do projeto. Este esteve presente em todas as reuniões previstas do *scrum*, estando a par do trabalho que ia sendo desenvolvido pela equipa, dando linhas orientadoras. Assim foi possível garantir que as entregas estavam mais próximas das expectativas do cliente.

A motivação da equipa teve um papel crucial para o sucesso do projeto. O *Agile* dá especial foco às pessoas. No *PROJ-2*, a equipa mantinha um ambiente de trabalho relaxado e divertido, o que facilitava a comunicação e consequentemente a resolução e identificação de constrangimentos, obstáculos e conflitos.

A natureza do projeto exigia uma flexibilidade constante da gestão, uma vez que novas tecnologias e ferramentas de desenvolvimento iam emergindo no mercado. Foi necessário adaptar o projeto a essas mesmas exigências.

Conclui-se que a adaptabilidade das metodologias de gestão de projetos terão de ser ajustadas à realidade dos mesmos, não existindo uma fórmula que garanta o seu sucesso. Comprova-se que a flexibilidade é uma característica exigida num mercado de cariz informático, que está em constante mudança e evolução.

REFERÊNCIAS

- ActiveCollab (n.d) Disponível em:<https://activecollab.com/blog/project-management/waterfall-project-management-methodology> [Acesso em: 23/06/2021]
- Ajam (2018) *Project Management Beyond Waterfall and Agile*, 1ª ed. London: CRC Press
- Atlassia Agile Coach (2018). Disponível em: <https://www.atlassian.com/agile/scrum> [Acesso em: 01/09/2021]
- Atlassian (n.d) Disponível em:<https://www.atlassian.com/agile/scrum/backlogs> [Acesso em: 27/06/2021]
- Barbee, D. (2013) *Agile Practices for Waterfall Projects*, 1ª ed. FL: J.Ross.
- Cohn, M. (2004) *User Stories Applied: For Agile Software Development*. 1ª ed. Boston: Addison Wesley.
- Davis, B. & Radford, D. (2014) *Going Beyond the Waterfall: Managing Scope Effectively Across the Project Life Cycle*, 1ª ed. FL: J.Ross.
- Digital Project Manager (2021) Disponível em: <https://thedigitalprojectmanager.com/how-run-sprint-retrospective/#what-is> [Acesso em: 27/06/2021]
- Jalote, P (2002) *Software Project Management in Practice*, 1ªed. India: Pearson.
- Kotter, J. P., & Heskett, J. L. (2011) *Corporate Culture and Performance*. 1ª ed. New York: Free Press.
- Kuehn, U. (2011) *Integrated Cost and Schedule Control in Project Management*, 2ª ed. Virginia: Management Concepts.
- Medium (2019) Disponível em: <https://medium.com/serious-scrum/scrums-connection-to-rugby-597405fed5ec> [Acesso em: 23/06/2021]
- Norman, Eric S. & Brotherton, Shelly A. & Fried, Robert T. (2008) *Work Breakdown Structures*, 1ª ed. New Jersey: John Wiley & Sons.
- Project Manager (2019) Disponível em: <https://www.projectmanager.com/blog/10-project-management-knowledge-areas> [Acesso em: 22/09/2021].

- Project Management Institute (2017) *Project Management Body of Knowledge*, 6ª ed. Pennsylvania: PMI.
- Project Management Institute. 2004. *Practice Standard for Work Breakdown Structures*. 2nd ed. Newtown Square, PA: Project Management Institute.
- Project Manager (2018) Disponível em: <https://www.projectmanager.com/blog/what-is-a-scrum-master-everything-you-need> [Acesso em: 27/06/2021]
- Research Gate (2018) Disponível em: https://www.researchgate.net/publication/323939022_Scrum_Culture_Definition [Acesso em: 25/09/2021]
- Schwaber, K. (2009) *Agile Project Management with Scrum*, 1ª ed. Washington, DC: Microsoft Press.
- Stepanek, G (2012) *Software Project Secrets: Why Software Projects Fail*, 1ª ed. New York: Apress
- Sutherland, J. (2014) *Scrum: The art of doing twice the work in half the time*, 1ª ed. New York: Crown Business.
- Villafiorita, A. (2014) *Introduction to Software Project Management*, 1ª ed. Florida: CRC Press.
- Weiss, J. (2014) *Business Ethics: A Stakeholder and Issues Management Approach*, 6ª ed. San Francisco: Berrett-Koehler.
- Womack, J. & Jones, D. (1996) *Lean Thinking: Banish Waste and Create Wealth in Your Corporation*, 2ª ed. New York: Free Press.
- Zenkit Blog (2019) Disponível em: <https://zenkit.com/en/blog/waterfall-project-management-an-overview/> [Acesso em: 23/06/2021]

ANEXOS

