



## Universidades Lusíada

Pereira, Inês Filipa Gomes

### **Aplicação de scrum e técnicas lean na gestão do portefólio de projetos de uma empresa de desenvolvimento de soluções informáticas**

<http://hdl.handle.net/11067/7582>

#### **Metadados**

|                           |   |
|---------------------------|---|
| <b>Data de Publicação</b> | 2023  |
| <b>Resumo</b>             | <p>Esta dissertação foi realizada na ProgramArt, uma empresa de desenvolvimento de soluções informáticas. Nesse contexto, a presente dissertação teve como objetivo a aplicação de metodologias de gestão Agile e melhoria contínua na gestão do portefólio de projetos da empresa, nomeadamente Scrum e técnicas Lean. Pretendeu-se compreender como essas abordagens podem ser integradas em sinergia para melhorar a execução de um portefólio de projetos, aumentar a capacidade de inovação e garantir a entrega de ...</p> <p>This dissertation was carried out at ProgramArt, a computer solutions development company. In this context, this dissertation aimed to apply Agile management methodologies and continuous improvement, namely Scrum and Lean techniques, in the management of the company's project portfolio. The aim was to understand how these methods can be integrated in synergy, in order to improve the portfolio implementation, by increasing the innovation capacity and ensuring the delivery of value to customers. I...</p> |
| <b>Palavras Chave</b>     | Gestão industrial, Software   |
| <b>Tipo</b>               | masterThesis  |
| <b>Revisão de Pares</b>   | no  |
| <b>Coleções</b>           | [ULF-FET] Dissertações  |

Esta página foi gerada automaticamente em 2024-09-27T00:15:04Z com informação proveniente do Repositório



**UNIVERSIDADE LUSÍADA**  
VILA NOVA DE FAMALICÃO

**APLICAÇÃO DE *SCRUM* E TÉCNICAS *LEAN* NA GESTÃO DO  
PORTEFÓLIO DE PROJETOS DE UMA EMPRESA DE  
DESENVOLVIMENTO DE SOLUÇÕES INFORMÁTICAS**

**Inês Filipa Gomes Pereira**

Dissertação para obtenção do Grau de Mestre em Engenharia e Gestão  
Industrial

Vila Nova de Famalicão – setembro 2023



**UNIVERSIDADE LUSÍADA**  
VILA NOVA DE FAMALICÃO

**APLICAÇÃO DE *SCRUM* E TÉCNICAS *LEAN* NA GESTÃO DO  
PORTEFÓLIO DE PROJETOS DE UMA EMPRESA DE  
DESENVOLVIMENTO DE SOLUÇÕES INFORMÁTICAS**

**Inês Filipa Gomes Pereira**

**Orientador:** Professora Doutora Ana Cristina Ferreira

Dissertação para obtenção do Grau de Mestre em Engenharia e Gestão  
Industrial

## **Agradecimentos**

Com a conclusão da dissertação de mestrado, gostava de salientar e manifestar os meus sinceros agradecimentos a todos que intervieram diretamente e indiretamente. A conclusão desta dissertação não teria sido possível sem o apoio e a contribuição de muitas pessoas.

Em primeiro lugar, gostaria de expressar a minha sincera gratidão à minha orientadora Professora Doutora Ana Cristina Ferreira que orientou o meu trabalho com dedicação e expertise, pela sua disponibilidade em todas as ocasiões para discutir o projeto, pela motivação e por todos os conselhos preciosos ao longo do último ano. Esta dissertação não seria possível sem a sua ajuda e contributo.

Agradecer ao grupo fantástico da ProgramArt (Nuno Pereira, Carlos Campos, André Borges e Fernando Vale) por todo o companheirismo, por todas as partilhas de ideias e até mesmo as críticas construtivas ao longo do projeto.

Agradecer ao João Rafael pelo companheirismo, disponibilidade pela paciência e os bons conselhos ao longo destes anos. Por todos os incríveis momentos passados, pelas aventuras e descobertas juntos, por ter sido um pilar nestes últimos anos e pelos ensinamentos e sabedoria que me tem passado.

Meus profundos agradecimentos também vão para a minha família e amigos, pais (Maria Gomes e José Pereira), cunhada (Mara Gonçalves) e em especial ao meu irmão (Nuno Pereira) pela paciência, disponibilidade, companheirismo, pelos valores transmitidos. A eles que me apoiaram incondicionalmente ao longo desta jornada académica. Ao Alexandre por todo o apoio incondicional que me deu ao longo de todo o processo.

## Resumo

Esta dissertação foi realizada na ProgramArt, uma empresa de desenvolvimento de soluções informáticas. Nesse contexto, a presente dissertação teve como objetivo a aplicação de metodologias de gestão *Agile* e melhoria contínua na gestão do portfólio de projetos da empresa, nomeadamente *Scrum* e técnicas *Lean*. Pretendeu-se compreender como essas abordagens podem ser integradas em sinergia para melhorar a execução de um portfólio de projetos, aumentar a capacidade de inovação e garantir a entrega de valor aos clientes. Para tal foram aplicados diferentes princípios subjacentes ao método *Scrum* e técnicas *Lean*, incluindo ferramentas como os 5 *Whys*, a matriz 5W2H e o *Kanban*.

A ProgramArt foca-se sobretudo no desenvolvimento de três tipos de soluções informáticas, sendo elas, a conceção de páginas *web*, a conceção de lojas *online* e o desenvolvimento de *software* para aplicações móveis. Com base na análise dos fluxos de informação e na caracterização das etapas de realização de cada tipo de serviço, foi possível identificar três principais problemas, a falta de conhecimento de *lead time* dos projetos e as inerentes dificuldades de comunicação com os clientes, orçamentação de serviços desajustada e a dificuldade de gestão do portfólio de projetos.

No sentido de resolver estes problemas, foram propostas quatro principais propostas de melhoria: criar sistemáticas e normalizar procedimentos, identificando os tempos estimados de execução das tarefas dos projetos; identificar as etapas necessárias para a definição de um orçamento; e aplicar as ferramentas de gestão *Agile* de projetos, identificando indicadores de desempenho. Com estas propostas foi possível estimar intervalos de *lead time* de projeto. Recorrendo à plataforma *Trello*, foi criado um quadro *Kanban* para cada um dos projetos da ProgramArt. Adicionalmente, foram desenvolvidos *dashboards* com recurso ao *Power BI* para monitorização de projetos e colaboradores. Neste último ponto, identificaram-se como indicadores de desempenho a percentagem relativa do número total tarefas realizadas e a quantidade de *bugs* corrigidos.

Os resultados evidenciam que a adoção de abordagens de gestão *Agile* promove melhorias nos ciclos de desenvolvimento, permitindo aprimorar a eficiência operacional e a excelência na entrega de soluções informáticas. Adicionalmente, permite aprofundar a compreensão das causas raízes de problemas. No entanto, também se destacam os desafios na transição para a adoção do *Scrum*, incluindo resistência à mudança, necessidade de formação e ajustes na estrutura de gestão.

**Palavras-chave:** Gestão *Agile*; Gestão de projetos; *Scrum*; Técnicas *Lean*; *Software*.

## **Abstract**

This dissertation was carried out at ProgramArt, a computer solutions development company. In this context, this dissertation aimed to apply Agile management methodologies and continuous improvement, namely Scrum and Lean techniques, in the management of the company's project portfolio. The aim was to understand how these methods can be integrated in synergy, in order to improve the portfolio implementation, by increasing the innovation capacity and ensuring the delivery of value to customers. In doing so, different principles underlying the Scrum and Lean techniques were applied, including tools such as the 5 Whys, the 5W2H matrix and Kanban.

ProgramArt is focused on the development of three types of IT solutions, namely, the design of web pages, the design of online stores and the development of software for mobile applications. Based on the analysis of information flows and the characterization of the implementation phases in carrying out each type of service, it was possible to identify three main problems, the lack of knowledge regarding the projects lead time and the inherent difficulties in customers communication, the service mismatch budgeting and the difficulty of managing the project portfolio.

In order to solve these problems, four main improvement proposals were proposed: the creation of systematics and standardizing procedures, identifying the estimated cycle times of project tasks; identify the steps necessary to define a budget; and apply Agile project management tools, identifying performance indicators. With these proposals it was possible to estimate project lead time intervals. Using the Trello platform, a Kanban board was created for each of ProgramArt's projects. Additionally, dashboards were developed using Power BI to both monitor projects and employees. In the latter, the relative percentage of the total number of tasks performed and the number of bugs fixed were identified as performance indicators.

The results show that the adoption of Agile management approaches promotes improvements in development cycles, allowing to improve operational efficiency and excellence in the delivery of the software solutions. Additionally, it allows you to deepen your understanding of the root causes of problems. However, challenges in the transition to adopting Scrum also stand out, including resistance to change, the need for training and adjustments in the management structure.

**Keywords:** Agile Management; Project management; Scrum; Lean tools; Software.

# Índice geral

|  |      |
|--|------|
| Resumo .....   | iv   |
| Abstract.....  | v    |
| Índice de figuras .....  | viii |
| Índice de tabelas .....  | ix   |
| Lista de abreviaturas .....  | x    |
| 1. Introdução .....  | 1    |
| 1.1. Enquadramento e motivação.....                                  | 1    |
| 1.2. Objetivos propostos .....                                       | 2    |
| 1.3. Metodologia aplicada.....                                       | 3    |
| 1.4. Organização da dissertação .....                                | 4    |
| 2. Fundamentação teórica .....                                       | 5    |
| 2.1. Integração da melhoria contínua com a gestão <i>Agile</i> ..... | 5    |
| 2.2. Gestão <i>Agile</i> .....                                       | 6    |
| 2.2.1. Metodologia <i>Scrum</i> .....                                | 7    |
| 2.2.2. Aplicações do <i>Scrum</i> .....                              | 7    |
| 2.3. <i>Lean</i> e ferramentas de melhoria contínua .....            | 9    |
| 2.3.1. Gestão visual e quadros <i>Kanban</i> .....                   | 9    |
| 2.3.2. Matriz 5W2H .....   | 11   |
| 2.3.3. Ferramenta 5 <i>Whys</i> .....                                | 12   |
| 2.3.4. Padronização dos processos de trabalho.....                   | 12   |
| 3. Apresentação da empresa .....                                     | 13   |
| 3.1. Caracterização da empresa.....                                  | 13   |
| 3.2. Tipologia de clientes e mercado alvo.....                       | 14   |
| 3.3. Produtos e serviços prestados .....                             | 15   |
| 3.3.1. Construção de páginas <i>web</i> .....                        | 16   |
| 3.3.2. Construção de lojas <i>online</i> .....                       | 16   |

|   |    |
|---|----|
| 3.3.3. Construção de aplicações móveis .....                        | 17 |
| 4. Análise crítica e identificação de problemas .....               | 19 |
| 4.1. Processo de criação de uma página <i>web</i> .....             | 19 |
| 4.2. Processo de criação de uma loja <i>online</i> .....            | 20 |
| 4.3. Processo de criação de uma aplicação móvel .....               | 21 |
| 4.4. Análise comparativa de requisitos dos três serviços.....       | 23 |
| 4.4. Identificação dos principais problemas.....                    | 25 |
| 4.4.1. Falta de conhecimento de <i>lead time</i> dos projetos ..... | 25 |
| 4.4.2. Dificuldade de orçamentação de serviços.....                 | 27 |
| 4.4.3. Dificuldades de gestão do portfólio de projetos .....        | 28 |
| 5. Identificação e implementação de propostas de melhoria .....     | 30 |
| 5.1. Identificação de oportunidades de melhoria .....               | 30 |
| 5.2. Criação de sistemáticas e normalização de procedimentos .....  | 33 |
| 5.2.1. Sistemática para criação de páginas <i>web</i> .....         | 33 |
| 5.2.2. Sistemática para criação de lojas <i>online</i> .....        | 37 |
| 5.2.3. Sistemática para criação de aplicações móveis .....          | 40 |
| 5.3. Criação de ferramenta de orçamentação automatizada.....        | 40 |
| 5.4. Aplicação de ferramentas de gestão <i>Agile</i> .....          | 43 |
| 5.4.1. Implementação do <i>Trello</i> .....                         | 43 |
| 5.4.2. Criação de <i>dashboards</i> .....                           | 44 |
| 5.5. Análise de resultados das propostas de melhoria.....           | 48 |
| 6. Conclusões .....   | 51 |
| 6.1. Principais conclusões .....                                    | 51 |
| 6.2. Propostas de trabalho futuro .....                             | 52 |
| Referências bibliográficas .....                                    | 53 |
| Apêndice 1 – Exemplo de portfólio orçamental .....                  | 57 |



## Índice de figuras

|  |    |
|--|----|
| <b>Figura 1</b> – Logótipo da empresa ProgramArt. ....   | 13 |
| <b>Figura 2</b> – Exemplo de página web desenvolvida pela empresa. ....  | 16 |
| <b>Figura 3</b> – Exemplo de uma loja online desenvolvida pela empresa.....  | 17 |
| <b>Figura 4</b> – Diagrama de tarefas do desenvolvimento de uma página web. ....                                       | 19 |
| <b>Figura 5</b> – Diagrama de tarefas do desenvolvimento de uma loja online.....                                       | 21 |
| <b>Figura 6</b> – Diagrama de tarefas do desenvolvimento de uma aplicação móvel. ....                                  | 22 |
| <b>Figura 7</b> – Comparação do tempo estimado e tempo real de execução de diferentes projetos realizados. ....        | 26 |
| <b>Figura 8</b> – Aplicação da ferramenta 5Whys. ....  | 27 |
| <b>Figura 9</b> – Identificação de subtarefas no desenvolvimento de páginas web. ....                                  | 34 |
| <b>Figura 10</b> – Diagrama atualizado das tarefas de desenvolvimento de páginas web (nova sistemática). ....          | 36 |
| <b>Figura 11</b> – Identificação de subtarefas no desenvolvimento de lojas online. ....                                | 38 |
| <b>Figura 12</b> – Diagrama atualizado das tarefas de desenvolvimento de lojas online (nova sistemática). ....         | 40 |
| <b>Figura 13</b> – Diagrama de etapas de elaboração da ferramenta de orçamentação. ....                                | 41 |
| <b>Figura 14</b> – Componente de custeio da ferramenta de orçamentação - portfólio orçamental da empresa. ....         | 42 |
| <b>Figura 15</b> – Quadro Kanban na plataforma Trello para desenvolvimento de uma loja online. ....                    | 44 |
| <b>Figura 16</b> – Dashboard em Power BI de monitorização de um projeto. ....  | 45 |
| <b>Figura 17</b> – <i>Dashboard</i> em <i>Power BI</i> de análise de desempenho dos colaboradores da ProgramArt. ....  | 47 |
| <b>Figura 18</b> – Estimativas de tempo e tempo real de diferentes projetos depois de implementadas as melhorias. .... | 48 |
| <b>Figura 19</b> – Mockups reais apresentados ao cliente. ....   | 50 |
| <b>Figura 20</b> – Portfólio orçamental. ....  | 57 |

## Índice de tabelas

|   |    |
|---|----|
| <b>Tabela 1</b> – Comparação dos atributos de cada serviço em função das fases de desenvolvimento.....                      | 18 |
| <b>Tabela 2</b> – Análise comparativa de atributos no desenvolvimento de páginas web, lojas <i>online</i> e aplicações..... | 24 |
| <b>Tabela 3</b> – Matriz 5W2H com a definição do plano de ações de melhoria .....   | 32 |
| <b>Tabela 4</b> – Tempos de execução das tarefas para o desenvolvimento de páginas web ....                                 | 35 |
| <b>Tabela 5</b> – Tempos de execução das tarefas para o desenvolvimento de uma loja <i>online</i> ...<br>.....              | 39 |

## Lista de abreviaturas

|                 |   |
|-----------------|---|
| API             | <i>Application Programming Interface</i>      |
| CEO             | <i>Chief Executive Officer</i>                |
| CSS             | <i>Cascading Style Sheets</i>                 |
| GPS             | <i>Global Positioning System</i>              |
| HTML            | <i>Hyper Text Markup Language</i>             |
| PDCA            | <i>Plan, Do, Check, Act</i>                   |
| <i>Power BI</i> | <i>Power Business Intelligence</i>            |
| <i>SCRUM</i>    | <i>Sprint, Cycle, Review, Update, Meeting</i> |
| SEO             | <i>Search Engine Optimization</i>             |
| TI              | Tecnologia da Informação                      |
| URL             | <i>Uniform Resource Locator</i>               |
| WIP             | <i>Work in progress</i>                       |

# 1. Introdução

Neste capítulo são apresentados os motivos que justificam a escolha do tema e o seu enquadramento. Além disso, é apresentada a metodologia de investigação aplicada e definem-se os objetivos que devem ser alcançados. Na última secção, identifica-se a estrutura e organização da dissertação.

## 1.1. Enquadramento e motivação

Nos dias de hoje, para lidar com os desafios complexos e dinâmicos do mercado de tecnologia informática, uma abordagem útil e eficaz é a gestão *Agile* de projetos (Al-Saqqá et al., 2020). No que diz respeito à gestão do portfólio de projetos de uma empresa de desenvolvimento de soluções informáticas, a aplicação do *Scrum* (corresponde às etapas de *Sprint, Cycle, Review, Update, Meeting*) e técnicas *Lean* pode trazer benefícios significativos (Lalmi et al., 2021). A aplicação de técnicas *Lean* conciliadas com uma gestão *Agile*, permite a adoção de práticas que apoiam a criação de valor na organização num contexto *office* e, ao mesmo tempo, reduzem os custos e os desperdícios de tempo.

O *Scrum* é uma metodologia baseada nos princípios de gestão *Agile* e tem como foco dividir o trabalho em pequenas tarefas, designadas por *sprints*, que normalmente têm duração de duas a quatro semanas. O *Scrum* enfatiza a cooperação entre as equipas e os clientes, a transparência do processo e a capacidade de responder rapidamente às mudanças. Uma vez aplicado o *Scrum* na gestão do portfólio de projetos, a empresa tem uma visão clara do estado de cada projeto, priorizar o trabalho com base no valor do projeto e ajustar as prioridades de acordo com as necessidades do cliente (Firdaus et al., 2019).

As técnicas *Lean* concentram-se na eliminação de desperdícios, na melhoria contínua e no fornecimento de valor aos clientes de forma eficiente. Ao aplicar as técnicas *Lean*, a empresa pode identificar e eliminar atividades que não agregam valor, reduzir a burocracia e melhorar o fluxo de trabalho. Os princípios *Lean* ao serem incorporados na gestão do portfólio de projetos, pode reduzir o tempo de entrega, melhorar a qualidade e aumentar a satisfação do cliente (Ikumapayi et al., 2020).

Desta forma, tanto o *Scrum* como as técnicas *Lean* são abordagens ágeis que procuram melhorar a eficiência, a produtividade e a entrega de valor aos clientes. Assim, os contributos das ferramentas *Lean* e das estratégias da gestão *Agile* na resolução de problemas e melhoria de desempenho de empresas na área informática devem ser estudados.

Desta forma, esta dissertação foca-se no uso destas metodologias ágeis, usadas de forma integrada, na gestão de projetos de Tecnologia de Informação (TI) na empresa ProgramArt. A ProgramArt é uma empresa de desenvolvimento de soluções informáticas, tendo como foco conceber, desenvolver, implementar e manter as soluções informáticas dos seus clientes que atuam nas mais diferentes atividades comerciais. A ProgramArt fornece produtos e serviços únicos, oferecendo um vasto leque de soluções, designadamente, páginas *web*, lojas *online*, aplicações para *smartphones*, *software* de gestão, entre outros.

O sucesso das empresas neste setor altamente dinâmico e competitivo depende da qualidade do serviço, da rapidez de entrega e da satisfação do cliente. A gestão de portfólio de projetos é crucial pois permite à empresa alinhar estrategicamente as suas iniciativas, melhorar a utilização de recursos e garantir que os projetos sejam concluídos no prazo e de acordo com o orçamento apresentado ao cliente. Assim, pretende-se analisar a aplicação destas estratégias e do seu potencial contributo para a melhoria de gestão do portfólio de projetos da ProgramArt. A aplicação integrada do *Scrum* e das técnicas *Lean* corresponde à principal motivação ao desenvolvimento da dissertação, de modo a aprimorar a gestão do portfólio de projetos e impulsionar o sucesso das iniciativas de desenvolvimento de soluções informáticas da ProgramArt.

## **1.2. Objetivos propostos**

O principal objetivo desta dissertação é a implementação de propostas de melhoria na gestão dos projetos, através da aplicação de ferramentas colaborativas associadas à abordagem *Scrum* e técnicas de melhoria contínua. Desta forma, pretende-se cruzar os princípios *Lean* com os pressupostos da gestão *Agile*. Para a realização deste trabalho, podem ser identificados os seguintes objetivos específicos:

- Identificar e caracterizar as diferentes soluções informáticas que a empresa ProgramArt desenvolve;
- Identificar e compreender todo o fluxo de informação que é gerado na empresa sobre as diferentes soluções informáticas desenvolvidas, de acordo com o tipo de serviço (lojas *online*, páginas de *design web*, desenvolvimento de *software* como aplicações móveis);
- Identificar os principais problemas do processo de gestão dos projetos da empresa – falta de sistematização de procedimentos; desenvolvimento de soluções

desnecessariamente complexas; replicação de tarefas pela falta de afetação de responsabilidades; comunicação ineficaz; má gestão do *backlog* e atrasos;

- Normalizar e melhorar o processo de recolha de requisitos dos clientes para o desenvolvimento das soluções.
- Ajustar o modelo de funcionamento *Scrum* à realidade da empresa, identificando os papéis na estrutura *Scrum* à equipa de trabalho, os ciclos de trabalho;
- Implementar uma ferramenta colaborativa para gerir a informação de cada projeto em desenvolvimento, incluindo a utilização de sistemas de *Kanban* visual e *dashboards*.

### **1.3. Metodologia aplicada**

A identificação das características das soluções desenvolvidas, a implementação de um plano de propostas de melhoria e o desenvolvimento de ferramentas colaborativas implicaram a adoção da metodologia Investigação-Ação. A metodologia Investigação-Ação desenvolve-se em cinco etapas: identificação dos problemas, definir um plano de ação, implementar o plano de ação, avaliar os resultados e modificar as práticas em função dos resultados de aprendizagem (Saunders et al., 2019). Os pontos fortes desta metodologia centram-se no envolvimento direto do investigador, no foco da mudança e no tempo necessário para diagnosticar, planear, agir e avaliar os problemas da organização (Coutinho et al., 2009).

Na primeira fase, foi efetuado um estudo na empresa de forma a compreender o seu funcionamento e identificar os seus problemas e possíveis oportunidades de melhoria, nomeadamente, as dificuldades de gestão de múltiplos projetos em simultâneo, a dificuldade de orçamentação de cada projeto dado o desconhecimento dos tempos de execução das diferentes etapas de execução. Posto isto, foi definido um plano de ação, nomeadamente a criação de sistemáticas a partir da análise do histórico dos três tipos de projetos, de forma a padronizar os procedimentos.

Incluída no plano de ação, identificou-se a necessidade de desenvolvimento de uma ferramenta colaborativa, o *Trello*, usando os princípios de gestão *Agile*. Por fim, de forma a avaliar os resultados de implementação deste plano de ações, identificaram-se indicadores de desempenho para avaliar o impacto das ferramentas aplicadas.

Não obstante a aplicação da metodologia de Investigação-ação, previamente à sua implementação, o investigador deve centrar a sua atenção no estudo da contextualização e enquadramento teórico de suporte aos domínios científicos envolvidos na exploração do projeto através da pesquisa e compilação bibliográfica. Assim, em primeira análise, foi efetuada uma revisão bibliográfica sobre os conceitos e ferramentas da abordagem *Scrum* e do seu cruzamento com o *Lean* e os ciclos PDCA (*plan, do, check, act*), ambos assentes na melhoria contínua.

#### **1.4. Organização da dissertação**

Esta dissertação está organizada em 5 capítulos. No primeiro capítulo é efetuada a contextualização e enquadramento da dissertação, sendo indicados os objetivos e metodologia de investigação. No segundo capítulo são apresentados os fundamentos que sintetizam o enquadramento teórico de suporte ao desenvolvimento do tema. No terceiro capítulo apresenta-se a empresa onde foi desenvolvida a presente dissertação. No quarto capítulo apresenta-se a análise crítica e identificação de problemas associados ao desenvolvimento das soluções informáticas da empresa. No quinto capítulo apresenta-se o plano de ações de melhoria com vista à integração das metodologias *Lean* e de gestão *Agile*.

Por fim, no sexto capítulo apresentam-se as principais conclusões e propostas de trabalho futuro.

## 2. Fundamentação teórica

Neste capítulo é apresentada a base teórica de apoio ao desenvolvimento da dissertação. Considerando o tópico em investigação, é analisada a integração da melhoria contínua com a gestão *Agile* de projetos, sendo posteriormente discutido o *Lean* e as ferramentas, como a gestão visual, *Kanban*, matriz 5W2H e a ferramenta dos *5Whys*. Neste capítulo são também abordadas e caracterizadas as técnicas de gestão *Agile* e *Scrum*.

### 2.1. Integração da melhoria contínua com a gestão *Agile*

Nos ambientes corporativos modernos, a eficiência, a agilidade e a capacidade de adaptação são requisitos essenciais para o sucesso empresarial. Posto isto, a gestão do portfólio de projetos torna-se um elemento estratégico, fazendo com que as organizações conquistem os seus objetivos de negócio de forma eficaz e competitiva (Shirokova et al., 2020)

Na gestão tradicional de projetos, frequentemente, as organizações deparam-se com desafios relacionados com os atrasos na entrega, baixa flexibilidade para mudanças e dificuldade em lidar com exigências dinâmicas do mercado. Consequentemente, surgiram abordagens ágeis, como o *Scrum*, e conceitos de melhoria contínua inspirados no *Lean Thinking* que procuraram revolucionar a forma como os projetos são geridos (Tam et al., 2020). O *Scrum* é uma metodologia ágil que se concentra em equipas auto-organizadas, iterações curtas de trabalho (*sprints*) e *feedback* contínuo dos *stakeholders*. Já as técnicas *Lean*, originalmente aplicadas em processos industriais, têm como objetivo eliminar desperdícios, aumentar a eficiência e melhorar a qualidade dos produtos e serviços entregues (Kothapalli & Nanda, 2019).

O sucesso de integração destas técnicas e ferramentas depende da cultura organizacional, da estruturação das equipas, do alinhamento estratégico das empresas e do sentido de compromisso (*engagement*) dos líderes e profissionais envolvidos na gestão de portfólio de projetos das empresas. Os resultados da integração da melhoria contínua com a gestão *Agile* podem contribuir significativamente para aprimorar a gestão de projetos, oferecendo diretrizes para uma adoção mais efetiva e bem-sucedida dessas abordagens em organizações similares. As empresas estão em constante evolução, enfrentando desafios técnicos e requisitos do mercado que exigem abordagens inovadoras na gestão de portfólio de projetos.



## 2.2. Gestão *Agile*

A gestão *Agile* oferece uma abordagem flexível e adaptativa que permite às empresas responder rapidamente às mudanças do mercado. No entanto, implementar uma gestão *Agile* no portfólio de projetos apresenta desafios específicos (Dyba & Dingsoyr, 2009). Um dos desafios consiste na mudança cultural pois, a adoção da gestão *Agile*, envolve uma mudança de responsabilidades, da mentalidade de colaboração, dos níveis de autonomia das equipas e das dinâmicas de trabalho corporativas. Muitas vezes, as organizações estão familiarizadas com uma estrutura hierárquica e processos rígidos, o que pode dificultar a transição para uma abordagem *Agile*. É essencial educar e envolver toda a equipa para que estes entendam os princípios ágeis e estejam dispostos a adotá-los (Stettina & Hörz, 2015).

Quando uma empresa faz a gestão de um portfólio de projetos, é crucial garantir a coordenação e a sincronização entre as equipas sendo que isto pode ser um desafio pois diferentes equipas podem ter ritmos de trabalho e prioridades diferentes. É necessário estabelecer mecanismos eficazes de comunicação e colaboração para garantir que todas as equipas estejam alinhadas e a trabalhar na mesma direção.

A gestão *Agile* coloca um forte foco na entrega de valor incremental ao cliente e na adaptação às suas necessidades em constante mudança. Contudo, isso pode ser um desafio quando os clientes estão acostumados a um modelo de entrega de projetos tradicional, com âmbito fixo e datas de entrega definidas com antecedência. A gestão *Agile* requer uma comunicação clara e constante com os clientes, para que eles compreendam os benefícios e as limitações dessa abordagem e estejam dispostos a colaborar de forma iterativa (Stoica et al., 2013). Este tipo de gestão valoriza a transparência e a aprendizagem contínua. No entanto, pode ser um desafio estabelecer métricas adequadas para medir o progresso e o desempenho dos projetos em contexto *Agile*. É importante desenvolver métricas ágeis e estabelecer um sistema de monitorização eficaz para que a empresa possa avaliar o desempenho do portfólio de projetos.

No contexto de empresas de desenvolvimento de soluções informáticas, a gestão *Agile* exige que as equipas tenham autonomia para definir prioridades e tomar decisões sobre a alocação de recursos, sobretudo porque decorrem diversos projetos em simultâneo. No entanto, a alocação de recursos de forma eficiente e equilibrada entre os projetos pode ser um desafio, especialmente quando há dependências ou restrições orçamentais. É necessário estabelecer critérios claros de priorização e um processo transparente para alocar recursos adequadamente (Senabre Hidalgo, n.d.).

### **2.2.1. Metodologia Scrum**

*Scrum* é uma metodologia *Agile* de gestão de projetos que foi desenvolvida para ajudar as equipas a serem mais flexíveis e adaptáveis, especialmente quando são projetos complexos, de longa duração ou que necessitem da intervenção de um elevado número de recursos. É uma metodologia amplamente utilizada em projetos de desenvolvimento de *software*, mas também pode ser aplicado em outras áreas. O *Scrum* baseia-se num ciclo de trabalho chamado *sprint* que geralmente dura de 2 a 4 semanas. Durante cada *sprint*, a equipa faz o planeamento, desenvolve e testa uma versão funcional do produto. Ao final de cada *sprint*, cada elemento demonstra o progresso de execução das suas tarefas e responsabilidades para o resto da equipa e faz o planeamento do próximo *sprint*.

O *Scrum* também inclui 3 papéis principais: o *Product Owner*, que representa os interesses dos *stakeholder* e prioriza as histórias do utilizador; o *Scrum Master* que é responsável por garantir que o *Scrum* é seguido corretamente; e a equipa de desenvolvimento, que é responsável por realizar o trabalho e entregar o produto (Oomen et al., 2017).

O *Scrum* é conhecido por ser uma metodologia *Agile* e interativa, que permite que as equipas sejam flexíveis e capazes de se adaptarem a mudanças no âmbito e nos requisitos do projeto. Isso é importante para projetos complexos, onde é difícil prever todos os requisitos no início do projeto. Esta metodologia representa uma estrutura de gestão *Agile* que tem sido amplamente aplicada, uma vez que a sua versatilidade permite que ela seja usada em diversas áreas e setores onde a gestão *Agile* de projetos é benéfica. De forma a compreender a versatilidade do *Scrum*, apresentadas algumas aplicações comuns do *Scrum* em diferentes contextos na secção 2.2.2, além de explorar suas principais vantagens e desvantagens (Dingsøyr et al., 2023).

### **2.2.2. Aplicações do Scrum**

O *Scrum* estabeleceu-se como uma abordagem paradigmática, permitindo que as equipas trabalhem de forma colaborativa, iterativa e com entregas em *sprints* curtos. Essa estrutura incentiva a adaptação contínua às mudanças dos requisitos, melhorando a transparência e garantindo que o produto ou serviço final tenha um valor agregado sólido (Verwijns & Russo, 2023). A sua aplicação pode ser estendida a diversos setores, como *marketing* e publicidade, onde a agilidade e a capacidade de adaptação rápida são essenciais. Equipas de *marketing* podem aproveitar a natureza iterativa do *Scrum* para executar campanhas eficazes (MacHado et al., 2020).

A abordagem do *Scrum* apresenta diversas vantagens e desvantagens em diferentes setores. A flexibilidade é uma das principais vantagens do *Scrum*. Essa abordagem permite que os projetos se ajustem de forma ágil às mudanças de requisitos, mantendo sempre o foco na entrega de valor. Além disso, o *Scrum* enfatiza a transparência e a colaboração entre as equipes. Estes aspectos promovem uma compreensão compartilhada dos objetivos e minimizam desentendimentos, contribuindo para um ambiente de trabalho mais harmonioso e produtivo. A entrega de valor contínua é uma característica central destas abordagens.

A natureza iterativa do *Scrum* permite que a empresa apresente resultados tangíveis em curtos intervalos de tempo. Isso possibilita obter *feedback* constante dos *stakeholders* e ajustar as prioridades de acordo com os requisitos do projeto, garantindo que o produto final ou serviço prestado esteja alinhado com as necessidades do mercado (Mccutcheon & Bunch, 2021). No entanto, a implementação bem-sucedida do *Scrum* não está isenta de desafios. Uma das desvantagens é a necessidade de ter equipes formadas e experientes no uso destas abordagens. Esse requisito pode ser um obstáculo em empresas que ainda não possuam conhecimento em *Scrum*. A capacitação e o desenvolvimento da equipe são essenciais para maximizar os benefícios destas abordagens. Outra desvantagem surge quando a abordagem é aplicada em projetos de grande escala. A gestão *Agile* é altamente eficaz em projetos de menor dimensão, sendo mais exigente a sua aplicação em projetos mais extensos ou que envolvam várias equipes. Isto pode obrigar a adaptações e soluções mais avançadas para coordenar efetivamente os esforços e garantir a coesão do projeto como um todo (Verwijns & Russo, 2023).

Além disso, o sucesso do *Scrum* depende do comprometimento ativo das partes interessadas. A falta de envolvimento por parte dos *stakeholders* pode prejudicar o progresso do projeto e afetar os resultados obtidos. O compromisso contínuo das partes interessadas é fundamental para garantir que as abordagens ágeis alcancem o seu potencial máximo (Berezutskyi et al., 2023).

Em suma, as vantagens da abordagem do *Scrum* incluem flexibilidade, transparência, colaboração e entrega de valor contínua. No entanto, é importante saber as desvantagens, nomeadamente, a necessidade de equipes especializadas, desafios em projetos complexos e a importância do compromisso das partes interessadas. A compreensão desses aspectos ajuda as empresas a aproveitar ao máximo essas abordagens em diversos setores.

### **2.3. *Lean* e ferramentas de melhoria contínua**

Para melhorar a eficiência e a qualidade dos processos, a normalização do trabalho pode ser usada como parte das ferramentas e técnicas *Lean*. Um fluxo de trabalho padronizado ajuda a reduzir a variabilidade e os erros do projeto. Posto isto, a normalização do trabalho envolve a criação de procedimentos padronizados para realizar atividades específicas. A padronização garante que todos os membros da equipa sigam os mesmos procedimentos e padrões, o que aumenta a consistência e a qualidade dos resultados (Feng & Ballard, 2008). Ao usar diagramas de processo e sistemáticas, a equipa economiza tempo e garante que as informações importantes não são negligenciadas. Para garantir que os produtos desenvolvidos cumpram com os padrões e requisitos definidos, a equipa tem um ponto de referência definido por critérios claros de qualidade (Middleton, 2001). Por fim, a normalização do trabalho pode incluir revisões e auditorias regulares dos resultados e processos de desenvolvimento de novo *know-how*. Estas revisões fornecem o *feedback* sobre como melhorar os processos, garantem a conformidade com os padrões estabelecidos e ajudam na identificação de oportunidades de melhoria (Dombrowski & Zahn, 2011).

Ao usar a normalização do trabalho no desenvolvimento informático, é importante ter em consideração que os processos e padrões devem ser flexíveis o suficiente para cumprir com a natureza ágil e iterativa do desenvolvimento de *software*. Esta deve ser vista como uma base sólida que pode ser alterada e melhorada de forma contínua, à medida que a equipa aprende com a experiência (Dyba & Dingsoyr, 2015).

Neste contexto, algumas ferramentas *Lean* que podem ajudar a reduzir e eliminar desperdícios, identificar problemas e melhorar desempenhos são a gestão visual, o *Kanban*, a metodologia *Kaizen* e ferramentas como a matriz 5W2H e 5Whys (Nicholas, 2018).

#### **2.3.1. Gestão visual e quadros *Kanban***

A integração da gestão visual e do método *Kanban* na gestão do portfólio de projetos de uma empresa pode ser uma estratégia eficaz para aprimorar a visibilidade, a eficiência e o controlo sobre o fluxo de trabalho. O *Kanban*, uma abordagem derivada dos princípios *Lean*, concentra-se em visualizar o trabalho, limitar o *Work in Process* (WIP) e melhorar continuamente os processos. Neste contexto, a aplicação do *Kanban* pode ser benéfica em diversas etapas de execução de projetos (Ahmad et al., 2013). É fundamental identificar todas as exigências que fazem parte do portfólio de projetos. Cada exigência, seja um projeto integral ou uma tarefa específica, deve ser representada por um cartão *Kanban*.

Quando se utiliza como uma ferramenta de gestão de projetos, estes cartões podem ser físicos, como *post-its* num quadro, ou *Kanban* digital. Devem-se criar colunas que representem as várias etapas do fluxo de trabalho no contexto da gestão de portfólio. Essas colunas podem incluir etapas como "Solicitações em espera", "Análise de prioridade", "Em execução", "Revisão" e "Concluído". A personalização das colunas é importante para alinhar o processo com as necessidades específicas da empresa. A visualização do fluxo de trabalho é uma das características centrais do *Kanban*. Colocar os cartões *Kanban* nas colunas apropriadas reflete o estado atual de cada exigência. Essa representação visual oferece uma visão clara e imediata de onde cada projeto ou tarefa se encontra no processo (Damij & Damij, 2021). Para evitar a sobrecarga da equipa e manter o foco nas exigências mais importantes, é essencial estabelecer limites para a quantidade de cartões que podem estar em cada coluna simultaneamente. Esta limitação do WIP promove a eficiência (Anderson et al., 2011).

A priorização contínua também desempenha um papel fundamental no *Kanban*. Através de sistemas visuais de sinalização, como cores, as exigências mais críticas ou urgentes podem ser destacadas, facilitando a designação de recursos e a tomada de decisões informadas. Reuniões de revisão regulares do *Kanban* são essenciais para discutir o progresso das exigências, identificar possíveis bloqueios e ajustar as prioridades conforme necessário. Essas reuniões proporcionam uma oportunidade para alinhar os objetivos da equipa e manter todos informados sobre o estado do portfólio de projetos (Oza et al., 2013).

A filosofia do *Kanban* também enfatiza a melhoria contínua. Encorajar a equipa a sugerir melhorias no processo com base em observações e *feedback* é crucial para aprimorar constantemente a eficiência e a qualidade do trabalho. É importante garantir que a equipa compreende os princípios do *Kanban* e saiba como aplicá-los efetivamente. A formação e a educação são investimentos cruciais para maximizar os benefícios dessa metodologia (Damij & Damij, 2021).

Em suma, a implementação do *Kanban* na gestão do portfólio de projetos de soluções informáticas oferece uma abordagem estruturada e visual para melhorar o fluxo de trabalho, promover a eficiência operacional e melhorar o controlo sobre as exigências. Adaptar o método *Kanban* ao contexto específico da empresa pode ser um passo significativo em direção a uma gestão de projetos mais eficaz e alinhada com as metas organizacionais.

### 2.3.2. Matriz 5W2H

A implementação de práticas *Agile* como *Scrum* e estratégias *Lean*, juntamente com o uso da matriz 5W2H, fornece uma abordagem abrangente e eficaz para identificar, avaliar e executar projetos de forma eficiente (Lisboa & Godoy, 2012). A matriz 5W2H surge nesse contexto como uma técnica de análise completa, projetada para identificar e examinar minuciosamente todos os aspetos importantes de um projeto. A matriz 5W2H é composta por sete componentes essenciais que guiam a análise e o planeamento detalhado, aumentando a eficiência e reduzindo os riscos.

A matriz 5W2H oferece um *framework* sistemático para o planeamento de projetos ao explorar sete dimensões cruciais: o quê (*what*), porquê (*why*), quem (*who*), onde (*where*), quando (*when*), como (*how*) e quanto (*how much*) (Klock et al., 2016). O elemento “*what*” define o intuito da tarefa ou projeto de forma concisa e específica, alinhando-se com a definição de metas claras. O “*why*” justifica a iniciativa, garantindo que a motivação está alinhada com os objetivos gerais da empresa e dos clientes. O aspeto “*who*” corresponde à distribuição de papéis e responsabilidades, uma componente crucial para o sucesso de projetos ágeis, onde a colaboração interdisciplinar é essencial. O “*where*” considera a localização da tarefa, um ponto importante para a componente logística dos projetos, que se alinha aos princípios *Lean* de melhoria de fluxos de trabalho. O “*when*” da matriz 5W2H está alinhado com a necessidade de estabelecer prazos específicos. A abordagem *Lean* valoriza a gestão de tempo eficiente. O “*how*” reflete o plano de ação detalhado e a execução. Finalmente, o elemento “*how much*” da matriz 5W2H centra-se na identificação dos custos com a implementação das estratégias *Lean* ou os custos de alocação de recursos para evitar desperdícios. Este aspeto é importante para garantir a viabilidade financeira do projeto (Klock et al., 2016).

A adoção da matriz 5W2H oferece uma série de vantagens tangíveis. Ao considerar cada elemento, as organizações podem garantir que todos os aspetos relevantes sejam abordados de forma abrangente. Isso, por sua vez, promove uma comunicação clara e eficaz entre os membros da equipa, minimiza riscos, aumenta a eficiência da execução e reduz a probabilidade de atrasos. A abordagem sistemática assegura que todas as partes envolvidas estão alinhadas, contribuindo para um resultado bem-sucedido (Klock et al., 2016).

No contexto da gestão de projetos, a matriz 5W2H surge como uma abordagem com foco na análise detalhada e abrangente de cada aspeto relevante, torna-a numa aliada valiosa na condução de iniciativas de sucesso.

Com esta ferramenta, as organizações podem promover a eficiência, minimizar riscos e atingir metas com maior eficácia. Ao adotar esta metodologia, as organizações estão preparadas para enfrentar os desafios dinâmicos do cenário empresarial atual, garantindo resultados sólidos e bem planejados (Klock et al., 2016).

### **2.3.3. Ferramenta 5Whys**

A inclusão da ferramenta dos 5 *Whys* enriquece a abordagem de gestão de projetos. Esta ferramenta é útil para descobrir e explorar as causas raízes de problemas. Começando pela identificação do problema, a análise prossegue com uma série de perguntas (por quê), cada uma baseada na resposta anterior. Essa análise profunda leva à descoberta da causa raiz subjacente, permitindo que a equipa adote medidas eficazes para resolver o problema e evitar recorrências futuras (Pinto, 2014).

### **2.3.4. Padronização dos processos de trabalho**

O trabalho normalizado ou *standard work*, corresponde a um método para definir os melhores procedimentos e a respetiva sequência, de forma a garantir que cada operador segue estritamente essa padronização. Ou seja, *standard work* consiste num conjunto de medidas e ações que conduzem à padronização e à aplicação da melhoria contínua das empresas na execução de processos.

Com a normalização dos processos, é criado o *framework* de base para a implementação do *Kaizen* e da melhoria contínua. O objetivo da normalização consiste na minimização de erros, na criação de sistemáticas e na adoção de práticas que evitem retrabalho. A padronização é essencial para descrever cada etapa, de acordo com o padrão estipulado. A sua implementação determina as ações a serem desempenhadas no presente, mas trata-se de uma metodologia que nunca está terminada e carece de correções e ajustes. A melhoria no padrão existente num dado momento, passa a ser a base para melhorias futuras (Pinto, 2014).

### 3. Apresentação da empresa

A dissertação foi realizada numa empresa de desenvolvimento informático, a ProgramArt. Neste capítulo é efetuada a caracterização da empresa, dos seus objetivos, missão e valores, bem como, a tipologia de clientes, o mercado alvo e os serviços por ela prestados.

#### 3.1. Caracterização da empresa

Fundada em 2015, a empresa ProgramArt (Figura 1) concebe, desenvolve, implementa e mantém soluções informáticas para as diferentes atividades comerciais. O principal objetivo é melhorar as vendas e a produtividade dos clientes, oferecendo um vasto leque de soluções que se adequam às mais diferentes necessidades.



Figura 1 – Logótipo da empresa ProgramArt.

A ProgramArt, com sede em Vila Nova de Famalicão, é composta por uma equipa multidisciplinar que para além de programadores conta com *designers*, *marketers*, *copyrighters* e fotógrafos, colmatando as várias necessidades das soluções que são desenvolvidas para cada um dos clientes. Atualmente, quando a empresa é contactada por um cliente, quem recebe esse contacto e faz a análise dos requisitos é um dos dois representantes da empresa, ou o *Chief Executive Officer* (CEO) ou o *designer* responsável. Posto isto, é efetuada a distribuição das tarefas, sendo da responsabilidade do CEO a verificação do desenvolvimento dos projetos, conjuntamente com mais dois programadores. Qualquer tipo de contacto com o cliente é feito através do CEO.



Com mais de 100 projetos já desenvolvidos, a ProgramaArt conta com um portfólio de clientes nacionais e internacionais que vão desde *startups* a empresas multinacionais. Esta assume-se como parceira dos seus clientes executando, na maioria dos casos, mais do que um projeto por cliente, o que demonstra satisfação com o trabalho desenvolvido.

A missão da empresa é criar soluções tecnológicas inovadoras, que transformem desafios em oportunidades para os clientes que impulsionem o sucesso dos seus projetos, ajudando-os a alcançar os seus objetivos de negócios de forma eficaz.

A ProgramaArt tem como visão a necessidade de ser reconhecida como uma líder no seu setor de atividade, pela excelência técnica, criatividade e compromisso. A equipa da ProgramaArt procura constantemente inovar e aprimorar as habilidades para enfrentar os desafios tecnológicos do futuro.

A ProgramaArt baseia-se em valores como a excelência, sendo que a empresa procura manter os elevados padrões de qualidade nos seus processos e serviços prestados. Outro dos seus valores é a inovação. A ProgramaArt estimula a criatividade e a inovação no seu trabalho, procurando constantemente formas de melhorar e encontrar soluções únicas para problemas complexos. Por último, a integridade é outro dos valores da empresa, sendo que esta mantém altos padrões de ética e integridade em todas as suas interações, agindo com transparência e honestidade em todos os aspetos.

### **3.2. Tipologia de clientes e mercado alvo**

A ProgramaArt é uma empresa no setor de programação, com um portfólio diversificado que abrange uma ampla gama de clientes, tanto a nível nacional como internacional. A sua reputação é construída sobre a base sólida de serviços excecionais em três áreas principais: desenvolvimento de páginas *web*, lojas *online* e aplicações móveis, criando soluções de *software* à medida de gestão.

A ProgramaArt abrange uma variedade de clientes na área de lojas *online* e páginas *web*. Desde *startups* em ascensão que procuram uma presença *online* marcante para impulsionar as suas vendas e alcance, até grandes multinacionais que precisam de páginas *web* complexas e lojas *online* de alto desempenho para atender a uma base global de clientes. O mercado-alvo inclui empresas de comércio eletrónico, organizações sem fins lucrativos, empresas de serviços, entre outros. A ProgramaArt adapta as suas soluções de *web design* e desenvolvimento de acordo com as necessidades específicas de cada cliente, garantindo que estes se destaquem num ambiente *online* competitivo.

No mercado de aplicações móveis, a ProgramaArt tem um portfólio de clientes diversificado. *Startups* que desejam lançar aplicações inovadoras e atraentes para dispositivos móveis podem contar com o desempenho da empresa para tornar sua visão numa realidade. Além disso, são várias as empresas multinacionais que confiam na ProgramaArt para desenvolver aplicações informáticas que melhoram a experiência do utilizador e aumentem a fidelidade à marca. O mercado-alvo inclui também empresas de tecnologia, empresas de comunicação, empresas do setor de saúde e muitos outros.

A ProgramaArt destaca-se na criação de *software* personalizado para gestão, especialmente para empresas que operam em ambientes *Windows*®. Os seus clientes incluem empresas de diversos setores, desde pequenas empresas que procuram melhorar os seus processos internos até grandes multinacionais que precisam de soluções avançadas de gestão. O mercado-alvo inclui empresas de logística, finanças, saúde, entre outros, que procuram aumentar a eficiência operacional e a produtividade por meio de soluções de *software* sob medida.

Em suma, a ProgramaArt é uma empresa versátil e inovadora que atende a uma ampla variedade de clientes, desde *startups* com visões ambiciosas até multinacionais que procuram soluções de alta qualidade. A sua experiência abrange as áreas de desenvolvimento *web*, aplicações móveis e *software* de gestão, permitindo-lhes satisfazer as necessidades de um mercado-alvo diversificado e em constante evolução. A ProgramaArt é conhecida pela sua capacidade de traduzir as ideias dos clientes em soluções de *software* excepcionais, contribuindo assim para o sucesso e crescimento de seus clientes num mundo cada vez mais digital.

### **3.3. Produtos e serviços prestados**

A ProgramaArt é uma empresa de programação que se destaca no mundo da tecnologia pela criação de soluções digitais inovadoras. Com uma equipa de especialistas apaixonados pela programação e *design*, a ProgramaArt compromete-se a transformar as ideias dos seus clientes em realidade digital. Os seus serviços abrangem uma ampla gama de necessidades tecnológicas, incluindo: páginas *web*, lojas *online* e aplicações móveis para *smartphones*.

### 3.3.1. Construção de páginas web

Uma página web é a porta de entrada para o mundo *online* de qualquer empresa. A ProgramArt entende a importância de ter uma presença digital forte e, portanto, oferece serviços de desenvolvimento web de alta qualidade. A equipa trabalha incansavelmente para criar páginas webs atraentes, funcionais e responsivas que refletem a identidade da marca. Independentemente de o cliente precisar de uma página web corporativa, um *blog* pessoal ou uma plataforma de comércio eletrónico, a ProgramArt está pronta para transformar as ideias em numa página web de destaque. A Figura 2 é um exemplo de uma página web desenvolvida pela ProgramArt para um dos seus clientes. Este produto pode consultado através da hiperligação: <https://aventauros.pt/>.

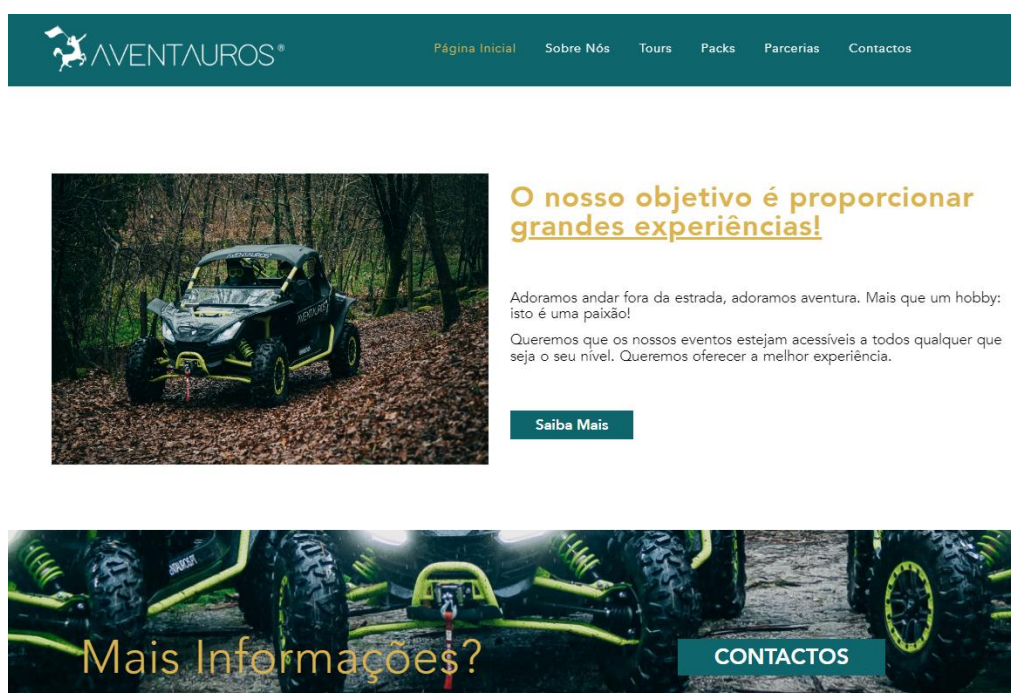
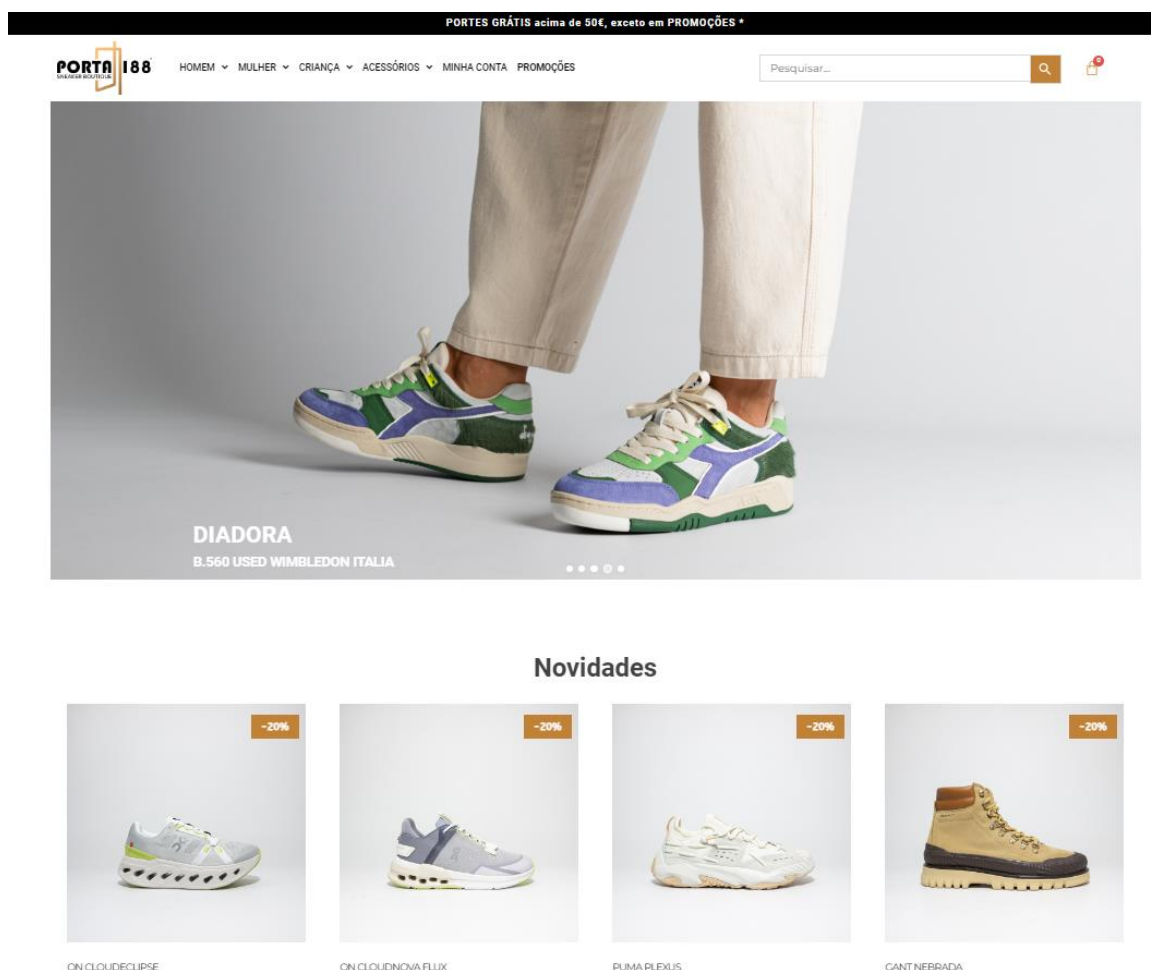


Figura 2 – Exemplo de página web desenvolvida pela empresa.

### 3.3.2. Construção de lojas online

O comércio eletrónico está em constante crescimento, e a ProgramArt quer ajudar os seus clientes a tirar o máximo proveito desse mercado em expansão. A equipa é especializada na criação de lojas *online*, fáceis de usar, que fornecem uma experiência de compra excepcional aos seus clientes. Desde a configuração inicial até à implementação de recursos avançados de comércio eletrónico, a equipa está comprometida em ajudar os seus clientes a prosperar no mundo digital.

A Figura 3 é um exemplo de uma loja *online* desenvolvida pela ProgramArt. Este produto pode consultado através da hiperligação: <https://porta188.com/> .



**Figura 3** – Exemplo de uma loja *online* desenvolvida pela empresa.

Na ProgramArt, o compromisso vai além do desenvolvimento tecnológico. A equipa está empenhada em fornecer suporte contínuo, manutenção e atualizações para garantir que as soluções digitais permaneçam relevantes e eficazes ao longo do tempo.

### **3.3.3. Construção de aplicações móveis**

A empresa desenvolve aplicações móveis personalizadas que atendem às exigências exclusivas dos seus clientes. Seja para *iOS* ou *Android*, a equipa trabalha em colaboração com o cliente para criar aplicações que oferecem uma experiência de utilizador excepcional. Desde aplicações empresariais até jogos e aplicações, a ProgramArt tem a experiência necessária para transformar as ideias dos seus clientes em soluções informáticas de sucesso.

A Tabela 1 identifica os principais atributos que a empresa tem de analisar nas diferentes fases de concepção e desenvolvimento dos serviços prestados. Os principais atributos incluem a definição do âmbito e objetivos dos serviços, a análise dos requisitos do cliente, a seleção da plataforma e tecnologia, a maximização *Search Engine Optimization* (SEO), as iterações de desenvolvimento e programação, a publicação, fase de testes e a manutenção e suporte das soluções desenvolvidas durante a fase de utilização.

**Tabela 1** – Comparação dos atributos de cada serviço em função das fases de desenvolvimento

| <b>Fases de desenvolvimento dos projetos</b> | <b>Página <i>web</i></b> | <b>Loja <i>online</i></b> | <b>Aplicação móvel</b> |
|--|--------------------------|---------------------------|------------------------|
| Definição do âmbito e objetivos              | <b>X</b>                 | <b>X</b>                  | <b>X</b>               |
| Análise dos Requisitos                       | <b>X</b>                 | <b>X</b>                  | <b>X</b>               |
| Escolha da plataforma e tecnologias          |                          |                           | <b>X</b>               |
| Maximização SEO                              | <b>X</b>                 | <b>X</b>                  |                        |
| Desenvolvimento                              | <b>X</b>                 | <b>X</b>                  | <b>X</b>               |
| <i>Design</i> e arquitetura                  | <b>X</b>                 | <b>X</b>                  | <b>X</b>               |
| Publicação                                   | <b>X</b>                 | <b>X</b>                  | <b>X</b>               |
| Testes                                       | <b>X</b>                 | <b>X</b>                  | <b>X</b>               |
| Manutenção e suporte                         | <b>X</b>                 | <b>X</b>                  | <b>X</b>               |

## 4. Análise crítica e identificação de problemas

A dissertação foi desenvolvida na ProgramArt, a qual presta três tipos de serviços distintos sendo eles as páginas *web*, lojas *online* e as aplicações móveis para *smartphones*. Neste capítulo serão abordados cada um dos serviços de forma pormenorizada, explicando como cada um deles é desenvolvido. Por fim, é efetuada uma breve análise comparativa dos três serviços prestados, sendo identificados os principais problemas da empresa.

### 4.1. Processo de criação de uma página *web*

Antes de começar o desenvolvimento de páginas *web*, é crucial definir o objetivo e o âmbito do projeto, entendendo o propósito e as funcionalidades pretendidas. De seguida, prossegue-se com a etapa de *design* e arquitetura, onde a *interface* gráfica do utilizador é projetada juntamente com a arquitetura de página *web*. Isto inclui definir as funcionalidades e integrações com outros sistemas. A atual sequência do processo de criação de uma página *web* é apresentada na Figura 4.

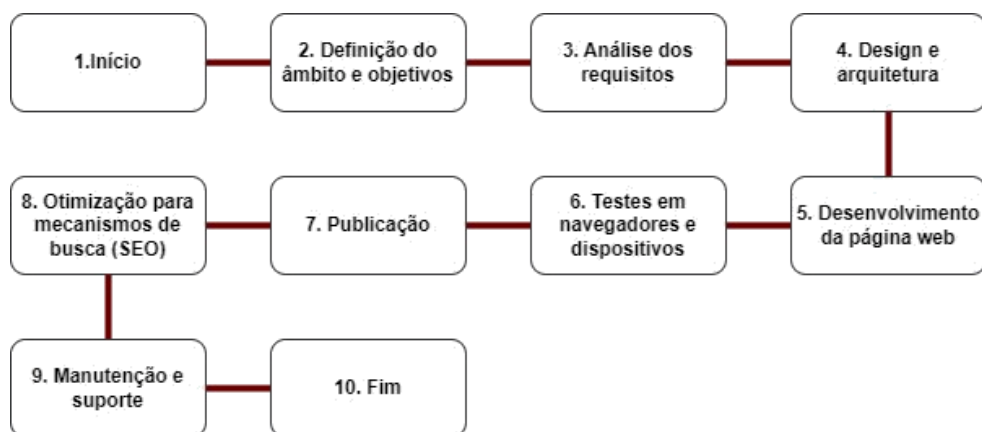


Figura 4 – Diagrama de tarefas do desenvolvimento de uma página *web*.

O desenvolvimento é o passo seguinte, onde o código é escrito para o tornar funcional, seguindo o planeamento e *design* estabelecidos anteriormente. Nessa fase, podem ocorrer o desenvolvimento de novas funcionalidades e testes para garantir que tudo funciona corretamente. É fundamental realizar testes exaustivos em diferentes navegadores e dispositivos para garantir a qualidade e a compatibilidade de páginas *web*.

Após o desenvolvimento e testes concluídos, escolhe-se um fornecedor de alojamento e realiza-se a publicação das páginas *web* no servidor. Para atrair tráfego e visibilidade, a página *web* precisa ser adaptada para mecanismos de procura e estar em conformidade com as melhores práticas de SEO.

Por fim, após o lançamento, é essencial fornecer manutenção e suporte contínuos. Isso inclui corrigir *bugs*, melhorar o desempenho e adicionar novas funcionalidades e recursos conforme necessário. É importante lembrar que a construção de páginas *web* é uma tarefa contínua, onde é preciso estar sempre a atualizar e a melhorar para garantir que a página *web* esteja a funcionar corretamente e adapta-se às necessidades dos utilizadores.

## **4.2. Processo de criação de uma loja *online***

A criação de uma loja *online* envolve diversos passos para garantir que a plataforma esteja totalmente funcional e ofereça uma experiência satisfatória aos clientes. Além das etapas tradicionais de desenvolvimento de páginas *web*, como a definição do âmbito, *design*, arquitetura, desenvolvimento, testes, hospedagem, SEO e manutenção, uma loja *online* requer etapas adicionais para o correto funcionamento do comércio eletrónico. É necessário realizar a integração com sistemas de pagamento, possibilitando que os clientes finalizem suas compras de forma segura e conveniente. Outro aspeto essencial é a integração com sistemas de gestão de *stock*, garantindo que o estado do *stock* esteja atualizado, para os clientes verificarem a disponibilidade do produto antes de concluir a compra.

A configuração do fluxo do carrinho de compras é um passo crucial, englobando a criação de páginas de *check-out*, confirmação de pedidos e páginas de pagamento, assegurando uma jornada de compra eficiente. Além disso, a integração com sistemas de envio e rastreio é indispensável, permitindo que os clientes acompanhem o estado das suas entregas e garantindo uma experiência transparente de toda a cadeia logística do processo.

A personalização da página de detalhes do produto também é essencial, incorporando imagens, descrições detalhadas, opções de cores, tamanhos e outros detalhes para fornecer informações completas ao cliente. Para facilitar a procura dos produtos específicos por parte dos clientes, é importante configurar opções de filtragem e classificação de produtos, tornando a navegação mais amigável e eficaz. Opções adicionais de personalização do carrinho de compras devem ser configuradas, como a possibilidade de ajustar quantidades, remover itens e outros recursos para aprimorar a experiência do utilizador.

Por fim, é necessário implementar opções de avaliação de produtos, permitindo que os clientes compartilhem as suas opiniões e comentários, o que pode influenciar outros consumidores e fortalecer a confiança na loja. Com a conclusão dessas etapas adicionais, a loja *online* está pronta para operar e proporcionar uma experiência positiva aos clientes, incentivando o crescimento e o sucesso do empreendimento no ambiente competitivo do comércio eletrônico. Salienta-se que podem haver outras necessidades, que são específicas para cada projeto. A atual sequência do processo de criação de uma aplicação de uma loja *online* é apresentada na Figura 5.

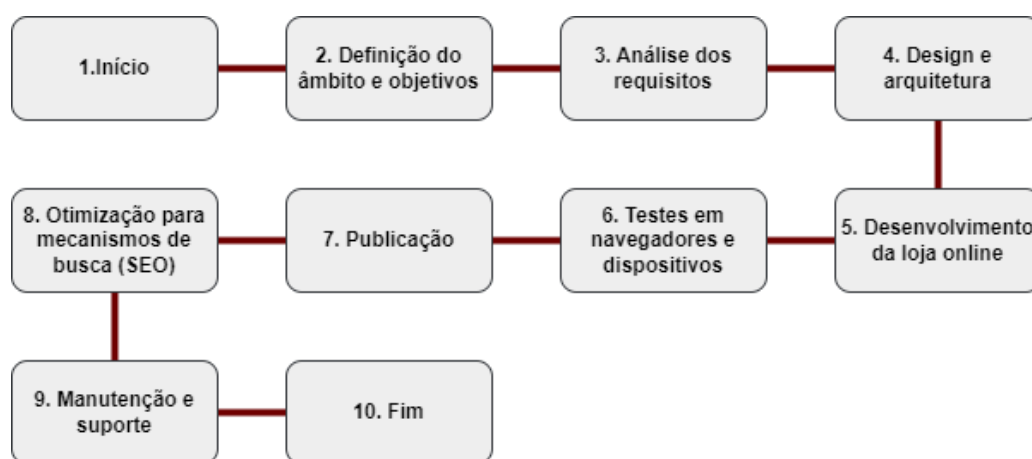


Figura 5 – Diagrama de tarefas do desenvolvimento de uma loja *online*.

### 4.3. Processo de criação de uma aplicação móvel

A criação de uma aplicação de *software* é um processo demorado e complexo, o que implica a união de esforços de várias equipas. É necessário definir corretamente os requisitos da aplicação assim como o seu *design* e estrutura. Este ponto é crucial para evitar avanços e recuos na fase de desenvolvimento. A codificação e a melhoria do código são partes essenciais do processo, a fim de garantir que a aplicação funciona sem problemas e com boa performance. Nesse sentido, é necessário efetuar testes em vários dispositivos, garantir a segurança dos dados e, muitas vezes, lidar com desafios imprevistos ao longo do desenvolvimento.

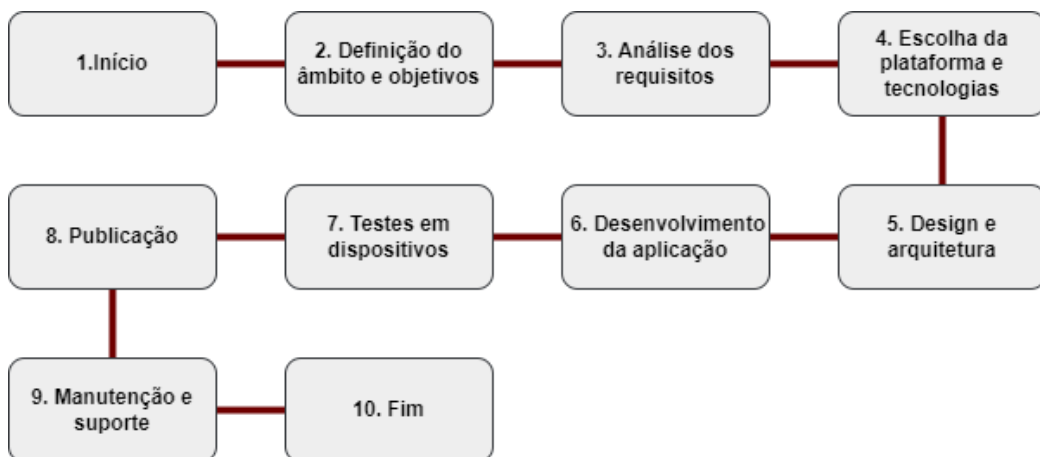
Além disso, a *interface* deve ser cuidadosamente projetada para proporcionar uma experiência positiva e única aos seus utilizadores. O desenvolvimento de uma aplicação requer uma abordagem estruturada, uma equipa qualificada e um plano de projeto bem definido para alcançar o sucesso.



O projeto começa com uma reunião com o cliente para entender as suas necessidades e requisitos. Neste ponto, é definida de forma clara o âmbito do projeto, incluindo funcionalidades, prazos e orçamentos. Em seguida, procede-se à análise de requisitos para identificar os elementos funcionais e não funcionais da aplicação. Isso engloba as necessidades do utilizador, as restrições do dispositivo e as limitações do sistema. A terceira etapa envolve a escolha da plataforma, decidindo se a aplicação será desenvolvida para *iOS*, *Android* ou ambas. Além disso, são selecionadas as ferramentas e tecnologias adequadas para o desenvolvimento.

Com a base sólida estabelecida, inicia-se o desenvolvimento da estrutura da aplicação, garantindo que esta está de acordo com as necessidades e permite o crescimento e manutenção futura. A fase seguinte aborda o desenvolvimento da *interface* do utilizador, onde é criada uma experiência intuitiva e atrativa, seguindo as melhores práticas de *design* de aplicações móveis. Na etapa de desenvolvimento e teste da aplicação, escreve-se o código para implementar as funcionalidades, sendo testado em vários dispositivos e sistemas operacionais.

A última etapa é o lançamento e manutenção da aplicação. O lançamento é efetuado nas lojas de aplicações, sendo fornecido o suporte contínuo para corrigir *bugs*, adicionar novas funcionalidades e prestar assistência. Isto implica resolver problemas, aprimorar o desempenho e incorporar novas funcionalidades e recursos conforme as necessidades. A atual sequência do processo de criação de uma aplicação móvel é apresentada na Figura 6.



**Figura 6** – Diagrama de tarefas do desenvolvimento de uma aplicação móvel.

#### 4.4. Análise comparativa de requisitos dos três serviços

É fundamental compreender as especificações técnicas que diferenciam o desenvolvimento de uma página *web*, de uma loja *online* e de uma aplicação móvel. Uma das principais diferenças entre o desenvolvimento de uma aplicação móvel e uma *página web* está nas plataformas em que são executados. Enquanto uma aplicação móvel é projetada para funcionar em dispositivos móveis, uma *página web* pode ser também acessada através de um navegador *web*, abrangendo tanto computadores quanto dispositivos móveis. Essa distinção é vital ao avaliar a relevância das metodologias ágeis como o *Scrum*, que têm de ser adaptadas às particularidades de cada plataforma.

As linguagens de programação desempenham um papel significativo. Aplicações móveis geralmente são desenvolvidas utilizando linguagens específicas para cada plataforma, como *Swift* para *iOS* ou *Kotlin* para *Android*. Em contraste, páginas *web* são geralmente construídas utilizando *HyperText Markup Language* (HTML), *Cascading Style Sheets* (CSS) e *JavaScript*, aproveitando sistemas de gestão de conteúdo de código aberto como o *WordPress*. A escolha das linguagens impacta diretamente na estratégia de desenvolvimento e na seleção de ferramentas ágeis, como o *Scrum*, que precisam se ajustar às tecnologias adotadas.

A interação com os dispositivos é outro ponto de divergência. Aplicações móveis têm a vantagem de acessar aos recursos do dispositivo, como câmara, *Global Positioning System* (GPS) e sensores, enriquecendo a experiência do utilizador. Por outro lado, as páginas *web* geralmente não têm esse acesso a recursos de *hardware*. Essa diferença na capacidade de interação influencia as prioridades e a dinâmica das equipas na implementação das funcionalidades.

O *design* desempenha um papel crucial na experiência do utilizador. A *interface* gráfica de uma aplicação móvel é projetada para se adequar ao contexto e às dimensões de um dispositivo móvel, enquanto a *interface* de páginas *web* deve ser adaptável tanto a computadores quanto a dispositivos móveis. Considerar essas peculiaridades na gestão do portfólio é vital para a alocação eficaz de recursos e o alinhamento dos projetos com os objetivos da empresa.

Por fim, a distribuição dos produtos é uma distinção marcante. Aplicações móveis precisam ser publicadas em lojas de aplicações, como a *App Store* da *Apple* ou a *Play Store* do *Google*, enquanto as páginas *web* são acessíveis através de um endereço *Uniform Resource Locator* (URL) sendo publicados num alojamento próprio.

A Tabela 2 apresenta uma análise comparativa de atributos no desenvolvimento de páginas *web*, lojas *online* e aplicações móveis.

**Tabela 2** – Análise comparativa de atributos no desenvolvimento de páginas *web*, lojas *online* e aplicações

| <b>Características</b>                       | <b>Páginas <i>web</i></b>                  | <b>Loja <i>Online</i></b>                  | <b>Aplicação</b>                       |
|--|--|--|--|
| <b>Dispositivos</b>                          | Dispositivos com browser/acesso à internet | Dispositivos com browser/acesso à internet | <i>Smartphones, tablets, wearables</i> |
| <b>Plataforma de execução</b>                | Navegador <i>web</i>                       | Navegador <i>web</i>                       | <i>iOS /Android</i>                    |
| <b>Linguagens de programação</b>             | Maioritariamente HTMLML, CSS, JS           | Maioritariamente HTMLML, CSS, J            | Maioritariamente <i>Swift, Kotlin</i>  |
| <b>Acesso e Recursos de dispositivo</b>      | Limitado                                   | Limitado                                   | Câmara, GPS, Sensores...               |
| <b>Design Responsivo</b>                     | Sim  | Sim  | Sim                                    |
| <b>Distribuição</b>                          | Através de URL                             | Através de URL                             | Lojas de aplicações                    |
| <b>Integração com sistemas de pagamentos</b> | Não  | Sim  | Opcional                               |
| <b>Integração com gestão de <i>stock</i></b> | Não  | Sim  | Opcional                               |
| <b>Configuração do carrinho de compras</b>   | Não  | Sim  | Opcional                               |
| <b>Integração com sistemas de envio</b>      | Não  | Sim  | Opcional                               |
| <b>Detalhes do Produto</b>                   | Não  | Sim  | Opcional                               |
| <b>Filtragem e classificação</b>             | Não  | Sim  | Opcional                               |
| <b>Personalização do carrinho</b>            | Não  | Sim  | Opcional                               |
| <b>Avaliação de produtos</b>                 | não  | Sim  | Opcional                               |
| <b>SEO</b>                                   | Sim  | Sim  | Sim                                    |
| <b>Atualizações e Manutenção</b>             | URL  | URL  | Lojas de aplicações                    |

Algumas características são comuns às três abordagens, enquanto outras são exclusivas de cada uma delas. A compreensão dessas diferenças é fundamental para a gestão de portfólio de projetos, pois permite uma adaptação eficaz das metodologias ágeis, como o *Scrum* e das práticas *Lean*, de acordo com os requisitos únicos de cada projeto, como é pretendido pela empresa.

Essa diferença tem implicações nas estratégias de lançamento, testes e recolha de *feedback*, que podem ser melhoradas por meio de abordagens ágeis. Integrar a compreensão dessas especificações técnicas com a aplicação de *Scrum* e técnicas *Lean* na gestão do portfólio de projetos de uma empresa de desenvolvimento de soluções informáticas pode permitir uma adaptação mais precisa das metodologias aos requisitos e características únicas de cada projeto.

#### **4.4. Identificação dos principais problemas**

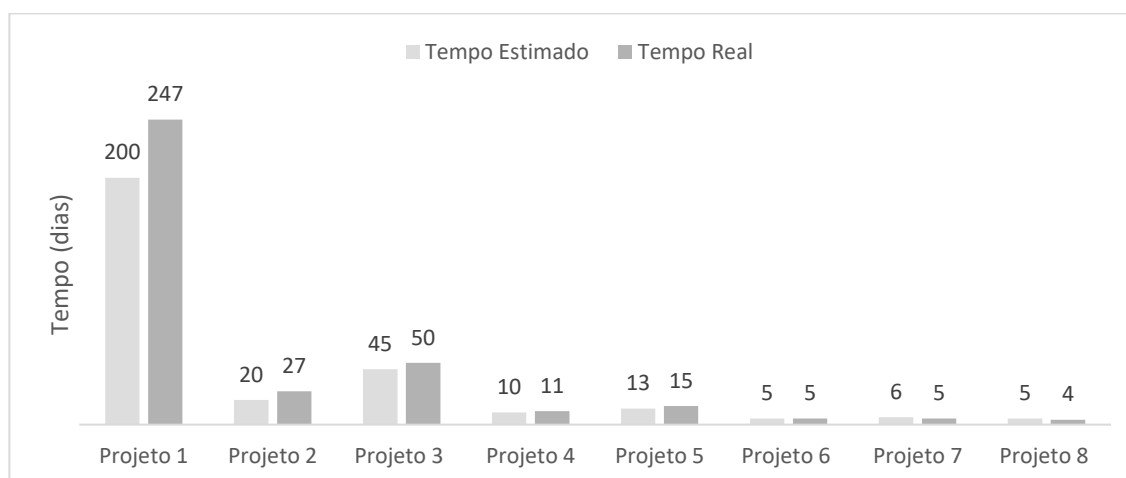
Na ProgramArt, embora muitos dos princípios de melhoria de trabalho sejam aplicados, alguns desafios recorrentes afetam a gestão do portfólio de projetos e as estimativas de tempo de conclusão e de orçamentação dos projetos. Esses desafios são cruciais para entender a complexidade e a gestão eficaz de projetos de *software* na empresa. Com base na observação e acompanhamento de alguns dos projetos da empresa desenvolvidos entre outubro de 2022 e julho de 2023, foram identificadas algumas das dificuldades operacionais da empresa. Nesse sentido, nesta secção são identificados os principais problemas que foram observados durante a realização da presente dissertação.

##### **4.4.1. Falta de conhecimento de *lead time* dos projetos**

Um problema frequente na ProgramArt estava relacionado com a falta de clareza nos requisitos fornecidos pelos clientes. Isso frequentemente levava a que o tempo necessário para o desenvolvimento do projeto fosse subestimado. É crucial que os requisitos sejam detalhados e bem compreendidos desde o início, para permitir estimativas precisas.

Além disso, muitas vezes, a equipa não considerava possíveis problemas/imprevistos ou atrasos, como solicitações de recursos adicionais dos proprietários do projeto. A falta de previsão desses fatores impactava significativamente no cronograma do projeto e resultava na necessidade de estender os prazos previamente definidos com os clientes. Em relação a esta questão temporal, o principal problema estava relacionado com a inadequada identificação de tempo para testes, revisões e correções de *bugs*.

Essas etapas são essenciais para garantir a qualidade do produto final e devem ser planejadas de forma adequada para evitar atrasos e custos adicionais. De forma a clarificar este problema, no decorrer da realização da dissertação foram acompanhados oito projetos que incluem os três tipos de serviço prestados pela empresa, comparando o tempo estimado no momento de orçamentação e o tempo real de concretização e desenvolvimento de cada um dos projetos. Como se pode verificar pela Figura 7, existem discrepâncias entre o tempo estimado e o tempo real de execução dos projetos concluídos que foram analisados do histórico da empresa. No caso do Projeto 1, que corresponde ao desenvolvimento de uma aplicação móvel, a discrepância entre tempo estimado e tempo real de execução do projeto corresponde a uma diferente de 37%. No caso dos restantes projetos de desenvolvimento de páginas *web* e lojas *online* (com a exceção do Projeto 2), existem diferenças entre os tempos orçamentados e os tempos de execução, mas estes são inferiores a 11,1%.



**Figura 7** – Comparação do tempo estimado e tempo real de execução de diferentes projetos realizados.

De forma a identificar a origem das dificuldades nas estimativas de tempo, foi adotada a técnica dos *5Whys*. Esta técnica permitiu explorar as causas subjacentes aos problemas que estavam a afetar o processo de desenvolvimento de páginas *web*. Na Figura 8 está presente a aplicação da ferramenta dos *5Whys*. Com base nas conversas informais com os colaboradores da empresa e após a apreciação efetuada com base no levantamento dos dados e da observação do trabalho, foram definidas 5 iterações interrogativas para determinar a causa raiz do problema.

A principal causa que afetava o processo de estimativas de tempo é a falta de um processo de recolha de requisitos dos projetos mais robusto, identificando a necessidade de incluir reuniões detalhadas com os clientes no início de cada projeto.

Além disso foi evidenciada a necessidade em desenvolver uma estrutura normalizada para documentar e avaliar as complexidades específicas de cada projeto e trabalhar com a equipa a importância da comunicação eficaz com o cliente na análise de requisitos.



Figura 8 – Aplicação da ferramenta 5Whys.

#### 4.4.2. Dificuldade de orçamentação de serviços

Na ProgramArt, a elaboração de orçamentos baseava-se na comparação com projetos semelhantes, realizados anteriormente, permitindo estimar custos com base em experiências passadas sem qualquer base de custo padrão. Em alguns casos, as mudanças no âmbito do projeto não eram devidamente consideradas no orçamento inicial. A solicitação de recursos adicionais ou modificações nos requisitos resultavam muitas vezes em desvios financeiros ao longo do projeto. A falta de consideração de possíveis problemas ou atrasos também contribuía para imprecisões nos orçamentos, estando desta forma este problema também relacionado com a ineficaz estimativa de duração dos projetos. Eventos imprevistos, como problemas técnicos ou dificuldades operacionais, não eram adequadamente previstos, levando a despesas não planeadas.

Outro ponto crítico era a simplificação excessiva no processo de orçamentação. Em alguns casos, a ProgramArt não avaliava completamente todas as tarefas inerentes a um projeto, resultando em orçamentos subestimados.

A falta de identificação de requisitos explícitos por parte dos clientes na fase negocial, a subestimação de possíveis problemas ou atrasos, e a não inclusão adequada de tempo para testes, revisões e correção de *bugs* eram aspetos que afetavam as estimativas de tempo. Da mesma forma, a omissão do impacto de mudanças, a falta de consideração de problemas e a falta de planificação dos projetos eram pontos críticos que influenciavam as estimativas de orçamentos.

Para melhorar a precisão das estimativas e garantir o sucesso dos projetos, a ProgramArt precisava abordar essas questões de forma mais estratégica e proativa. Isso envolveria uma maior comunicação com os clientes para obter requisitos claros, a inclusão de margens de tempo para acomodar mudanças e problemas, bem como uma análise mais detalhada da complexidade do projeto no processo de orçamentação. Ao enfrentar esses desafios, a empresa poderia aprimorar os seus processos e fornecer resultados mais eficientes e alinhados com as expectativas dos clientes.

A elaboração de estimativas mais realistas levando em conta as contingências operacionais e a realização de análises de impacto sempre que ocorrerem mudanças no âmbito do projeto devem ser contabilizadas na fase de orçamentação. A simplificação do processo de orçamentação dos projetos da empresa pode também estar associada às fragilidades formativas da equipa nesse âmbito. Foram percecionadas carências da equipa no que diz respeito à capacidade de gerir projetos de forma eficiente, dado que não eram considerados todos os aspetos, desde a conceção até a entrega dos projetos. A aprendizagem contínua e a adaptação são cruciais em um ambiente de desenvolvimento de *software* em constante evolução.

#### **4.4.3. Dificuldades de gestão do portfólio de projetos**

Um dos problemas identificados na empresa prende-se com a falta de uso de ferramentas ágeis na gestão de projetos. A não utilização deste tipo de ferramentas acaba por estar relacionada com os dois problemas anteriormente identificados na empresa, ou seja, as estimativas de tempo imprecisas, os problemas de comunicação e de troca de informação com os clientes sobre os requisitos. Estas problemáticas conduzem a outras dificuldades, tais como, os constrangimentos em priorizar tarefas e até mesmo a falta de visibilidade do progresso do desenvolvimento dos projetos.

Numa empresa de desenvolvimento de *software* que enfrenta problemas como estimativas de tempo incorretas e falhas de comunicação, a implementação de metodologias ágeis e ferramentas de gestão como *Scrum*, *Kanban* e *Trello* pode melhorar a gestão de projetos. Por exemplo, ao usar o *Scrum*, a equipa pode beneficiar da estrutura de *sprints*, reuniões diárias de acompanhamento e retrospectivas, que ajudam a definir metas claras a manter e em linha com os objetivos do projeto.

As dificuldades de gestão de projetos também estão relacionadas com as dificuldades de comunicação interna e a falta de ferramentas de consulta que permitam identificar os estados dos projetos, as tarefas já concluídas, as que estão a ser realizadas e as que ainda têm de ser iniciadas, além da identificação das respetivas responsabilidades. Nesse sentido, faltam ferramentas de gestão visual, sobretudo digitais, tendo em conta a área de atuação da ProgramaArt. Durante a realização da dissertação, verificou-se a falta de sistemas *Kanban* ou outras ferramentas que informassem acerca do fluxo de trabalho, nomeadamente, o que estava em desenvolvimento e o que estava pronto para revisão ou entrega.



## **5. Identificação e implementação de propostas de melhoria**

Considerando o levantamento dos principais problemas da empresa, apresenta-se neste capítulo a identificação e implementação das principais propostas de melhoria desenvolvidas com base no plano de ações que foi definido por intermédio da matriz 5W2H.

### **5.1. Identificação de oportunidades de melhoria**

A empresa em questão não utilizava nenhuma metodologia de trabalho padronizada, sendo que o seu método de trabalho era desempenhado de acordo com o projeto. Até à data de início de realização da presente dissertação não existiam registos detalhados de procedimentos ou de tempos de cada tarefa dos projetos passados. Para colmatar este problema, é importante a identificação de todas as tarefas e efetuar um levantamento dos tempos de execução de cada uma dessas tarefas.

Uma parte destes procedimentos normalizados deve integrar a definição de reuniões com os clientes, através da implementação de eventos como reuniões de *kickoff* e videoconferências devidamente programadas para identificação de todos os requisitos no âmbito dos projetos. Além da comunicação com o cliente, devem ser fomentadas reuniões da equipa de desenvolvimento. Garantir que a equipa comunica de forma eficaz, seja através de reuniões diárias, *chats* ou outras ferramentas de comunicação, ajuda a evitar deficientes interpretações do estado dos projetos e das respetivas tarefas, permitindo adicionalmente alinhar expectativas e resolver problemas de forma rápida.

As estimativas de tempo imprecisas são um problema da empresa, sendo que este problema também se relaciona com a comunicação ineficiente, falta de documentação de procedimentos e falta de gestão visual. Alguns atrasos que decorrem nos projetos devem-se ao desconhecimento do estado de desenvolvimento dos diferentes projetos. Criar sistemáticas através da identificação das etapas de desenvolvimento dos serviços prestados pela empresa e efetuar um levantamento de tempos padrão para cada uma das tarefas pode evitar atrasos, levar a orçamentos mais precisos e garantir que a entrega é realizada dentro das expectativas dos clientes.

Em relação à orçamentação, este processo também deve ser refinado, através do estabelecimento de um processo sequencial mais preciso para orçamentar projetos, tendo em conta possíveis mudanças de âmbito e contingências, acautelando para derrapagens temporais que acarretam maiores custos operacionais.

Uma das principais oportunidades de melhoria para a ProgramArt é a adoção de gestão *Agile* de projetos. A gestão eficiente de projetos numa empresa de desenvolvimento de soluções informáticas é essencial para alcançar o sucesso no mercado altamente competitivo da tecnologia. Neste contexto, a aplicação de metodologias ágeis, como *Scrum* e *Kanban*, torna-se uma abordagem fundamental para melhorar o trabalho e garantir que os projetos sejam entregues de forma eficaz e com alta qualidade.

Desta forma, uma das principais propostas de melhoria desta dissertação é a adoção do *Scrum*, enquanto abordagem que permite à equipa manter-se organizada, priorizar tarefas e garantir que o trabalho esteja sempre em progresso. Através de *sprints* e reuniões diárias, o *Scrum* facilita o acompanhamento do progresso do projeto, a identificação de obstáculos e a adaptação contínua. Os quadros *Kanban*, por sua vez, oferecem uma gestão visual do fluxo de trabalho, permitindo que a equipa execute as tarefas de forma eficiente. Esta metodologia é particularmente útil para empresas que lidam com um grande volume de exigências em constante mudança, tal como a ProgramArt. Uma plataforma a partir da qual é possível conciliar as potencialidades do *Scrum* e da gestão visual é o *Trello*. Esta ferramenta é muito útil para auxiliar na organização das tarefas, atribuição de responsabilidades e acompanhamento do progresso. Além disso, permitem a criação de quadros e listas personalizadas que se adaptam às necessidades de cada projeto, resultando numa maior visibilidade, controlo e eficiência.

Na sequência da implementação destas ferramentas, propõe-se a definição de indicadores de desempenho (*Key Performance Indicators (KPI)*), adequados à realidade da empresa, de forma a medir e avaliar o progresso da equipa e dos respetivos projetos. Indicadores como tempo de entrega, qualidade do código, número de *bugs* corrigidos e satisfação do cliente são essenciais para identificar áreas e oportunidades de melhoria.

Este plano de ações de melhoria foi definido com recurso à ferramenta 5W2H, conforme apresentado na Tabela 3. Esta ferramenta fornece uma visão clara de quais ações devem ser realizadas, por que são importantes, quem é responsável, onde serão implementadas, quando ocorrerão, como serão executadas e qual será o investimento necessário para a sua implementação. Desta forma, a ProgramArt pode abordar os desafios identificados e melhorar a gestão de projetos de desenvolvimento de soluções informáticas.

**Tabela 3** – Matriz 5W2H com a definição do plano de ações de melhoria

| <i>What</i>   | <i>Why</i>  | <i>Who</i>                                      | <i>Where</i>                               | <i>When</i>            | <i>How</i>  | <i>How much</i> |
|---|---|---|--|------------------------|---|-----------------|
| <b>Criar sistemáticas e normalizar procedimentos</b>      | Melhorar a comunicação é essencial para alcançar uma gestão de projetos bem-sucedida e manter clientes satisfeitos                                    | Equipa de gestão de projetos e equipa de vendas | Departamento de desenvolvimento            | Out.22<br>–<br>Jan. 23 | Criar um protocolo de comunicação, designar responsabilidades, definir pontos-chave a serem discutidos, programar reuniões de <i>kickoff</i> , videoconferências e uso de salas de <i>chat</i> para a comunicação entre elementos | 0 €             |
|   | Melhorar a precisão das estimativas de tempo ajuda a evitar atrasos, orçamento desajustados e garante a entregas dentro das expectativas dos clientes | Equipa de gestão de projetos                    | Departamento de desenvolvimento            | Out.22<br>–<br>Fev.23  | Criar sistemáticas através da identificação das etapas de desenvolvimento dos serviços prestados pela empresa e efetuar um levantamento de tempos padrão para cada uma das tarefas  | 0 €             |
| <b>Refinar estimativas de orçamento</b>                   | Melhorar a precisão das estimativas de custo para evitar exceder orçamentos e assegurar que os projetos sejam financeiramente controlados             | Equipa de gestão de projetos e financeira       | Departamento de desenvolvimento            | Out.22<br>–<br>Jan.23  | Estabelecer um processo mais preciso para orçamentar projetos, tendo em conta possíveis mudanças de âmbito e contingências, acautelando para derrapagens temporais que acarretam maiores custos operacionais                      | 0 €             |
| <b>Aplicação de ferramentas de gestão Agile</b>           | Utilizar metodologias ágeis e ferramentas de gestão ajuda a melhorar a eficiência e a organização na gestão de projetos                               | Equipa de gestão de projetos                    | Departamento de desenvolvimento            | Jan.23<br>–<br>Jul.23  | Treinar a equipa, estabelecer processos ágeis, adotar ferramentas de gestão. Implementar o <i>Scrum</i> , <i>Kanban</i> e ferramentas de gestão como o <i>Trello</i>  | 0 €             |
| <b>Definir indicadores de monitorização de desempenho</b> | Estabelecer KPI é fundamental para medir o desempenho, identificar áreas de melhoria e alcançar os objetivos  | Equipa de gestão de projetos                    | Gestão;<br>Departamento de desenvolvimento | Jan. 23<br>–<br>Jul.23 | Identificar KPI relevantes, definir metas, criar sistemas de monitorização através de <i>dashboards</i>   | 0 €             |

## 5.2. Criação de sistemáticas e normalização de procedimentos

Para abordar os desafios identificados na gestão de projetos na ProgramArt, é fundamental estabelecer novas sistemáticas e criar ferramentas específicas. Estas ações visam melhorar a precisão das estimativas de tempo, melhorar o planeamento do orçamento e automatizar a monitorização dos processos. Nesta secção, são detalhadas as principais iniciativas em implementação na empresa.

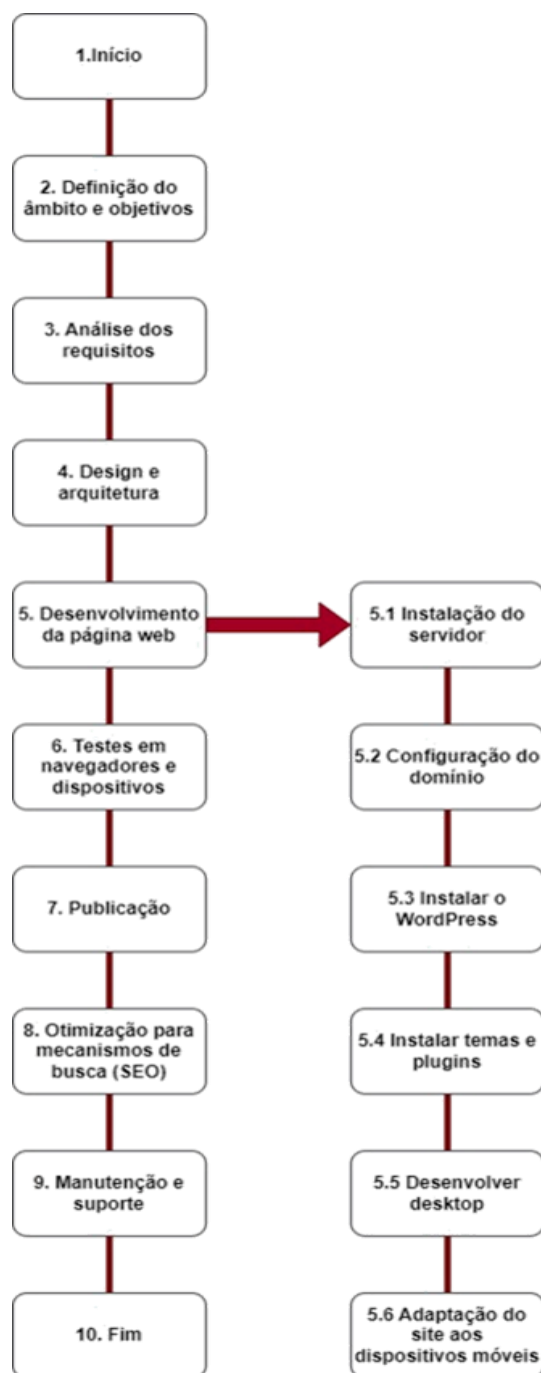
### 5.2.1. Sistemática para criação de páginas *web*

A gestão de tempos desempenha um papel fundamental no contexto da aplicação de metodologias ágeis, como o *Scrum* ou o *Lean*, e na gestão do portfólio de projetos de uma empresa. Esta subsecção centra-se na importância de identificar, padronizar e quantificar a duração das diferentes tarefas envolvidas nos projetos de criação de páginas *web*, lojas *online* e aplicações móveis, tendo como objetivo melhorar a eficiência e a qualidade das entregas.

Uma vez que na empresa não havia registos de tempos por tarefas, a primeira etapa foi entender e subdividir os três serviços – páginas *web*, lojas *online* e aplicações móveis – em tarefas mais específicas, de forma a conseguir observar e analisar a sequência e o tempo de realização de cada tarefa. Posto isto, foram acompanhados diferentes projetos com tarefas similares até à sua conclusão. Na Figura 9 estão descritas as etapas necessárias para o desenvolvimento de páginas *web* que ainda não tinham sido padronizadas, especificamente em relação à etapa (5), o desenvolvimento da página *web* em si. Desta forma foram identificadas as tarefas de instalação do servidor, a configuração do domínio, instalar temas e *plugins*, desenvolver o desktop e, por fim, adaptar a página desenvolvida também a dispositivos móveis. Estas tarefas são personalizáveis e podem variar de projeto para projeto.

Com a identificação das tarefas, o passo seguinte passou pela identificação e estimativa dos respetivos tempos de execução. De forma a evitar estimativas desajustadas e a melhorar a distribuição do tempo para testes e revisões, é crucial estabelecer tempos padronizados realistas. Isso requer a medição e análise detalhada dos tempos em cada fase do ciclo de vida do projeto, desde a conceção até à implantação e manutenção. Ao fazer o registo do tempo gasto em atividades específicas, é possível identificar tarefas onde o processo enfrenta atrasos ou desperdícios de recursos, possibilitando melhorias.

Além disso, a padronização dos tempos é um passo crucial no desenvolvimento de páginas *web*. Este processo oferece benefícios significativos, como estimativas de prazos mais precisas, identificação ágil de desvios e, conseqüentemente, um aumento geral na produtividade. É importante notar que a padronização dos tempos não deve ser rígida e inflexível, pois os projetos de desenvolvimento de *software* têm sua própria complexidade variável. Portanto, manter uma abordagem flexível e adaptável é essencial.



**Figura 9** – Identificação de subtarefas no desenvolvimento de páginas *web*.

Na Tabela 4 apresentam-se os tempos médios de execução das tarefas para o desenvolvimento de páginas *web*. Os tempos identificados resultam dos projetos acompanhados ao longo da realização da dissertação (tempos dos projetos da Figura 7). Analisando criticamente os dados recolhidos, foi possível constatar que algumas tarefas têm tempos que são padronizáveis, enquanto que outras tarefas têm uma enorme variabilidade, dependendo de projeto para projeto. Assim, para as tarefas consideradas como padronizáveis, foi determinado um tempo de execução com base na média dos tempos retirados. Nas restantes tarefas, optou-se por identificar o intervalo de tempos de execução. No desenvolvimento de *páginas web*, as tarefas de análise dos requisitos, instalação do servidor, configuração do domínio, instalação do *WordPress*, temas e *plugins* assim como a publicação, podem ser assumidas como tarefas de tempo padronizável. No total, estas representam um tempo estimado de 7 horas.

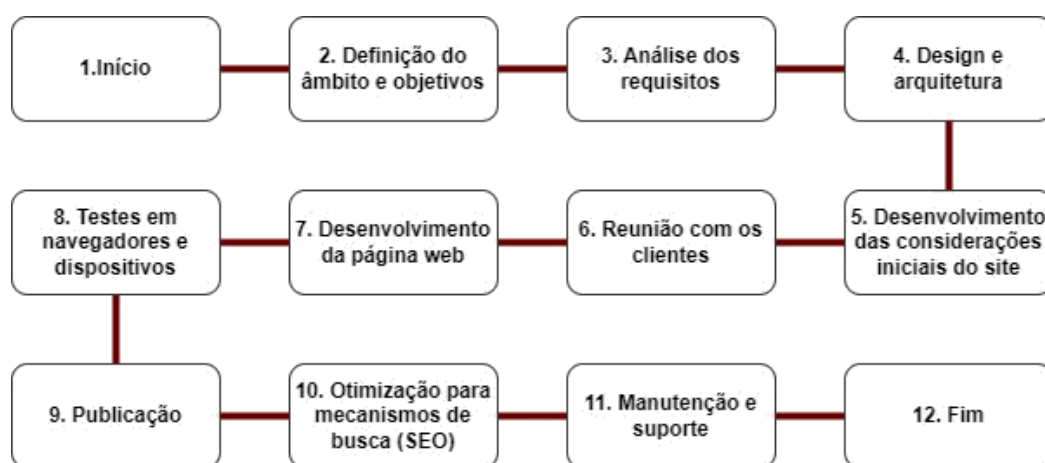
A definição do âmbito e objetivos, *design* e arquitetura, adaptação da página *web* a dispositivos móveis, melhorias para mecanismo de busca e manutenção e suporte são tarefas de duração variável, dependendo da complexidade dos requisitos de cada projeto. Considerando os tempos das tarefas variáveis, foi considerado que o *lead time* de execução pode variar, no limite, entre 29 horas para os projetos de menor complexidade e 166 horas para os projetos de maior complexidade.

**Tabela 4** – Tempos de execução das tarefas para o desenvolvimento de páginas *web*

| <b>Identificação das tarefas</b>                            | <b>Tempo (horas)</b> |
|---|----------------------|
| 2 Definição do âmbito e objetivos                           | [2; 5]               |
| 3 Análise dos requisitos                                    | 1                    |
| 4 <i>Design</i> e arquitetura                               | [8; 24]              |
| 5.1 Instalação do servidor                                  | 2                    |
| 5.2 Configuração do domínio                                 | 0,5                  |
| 5.3 Instalar o <i>WordPress</i>                             | 0,5                  |
| 5.4 Instalar temas e <i>plugins</i>                         | 1                    |
| 5.5 Desenvolver <i>desktop</i>                              | [8; 40]              |
| 5.6 Adaptação de páginas <i>web</i> aos dispositivos móveis | [2; 20]              |
| 6 Testes  | -                    |
| 7 Publicação  | 2                    |
| 8 Melhoria para mecanismos de busca                         | [1; 40]              |
| 9 Manutenção e suporte                                      | [1; 30]              |

Não é fácil definir um *lead time* específico, pois existem muitas etapas que são variáveis e dependem das especificações de cada projeto. Não foi identificado um valor para tempo de testes devido ao facto de na empresa não haver tempo estipulado para tal.

Com base na revisão do processo de desenvolvimento de página *web*, além de melhorar as estimativas de tempo de execução, redefiniu-se e padronizou-se um novo diagrama de tarefas do processo de desenvolvimento de páginas *web*, conforme apresentado na Figura 10. Assim, o desenvolvimento de uma página *web* começa pela definição do âmbito e objetivos (ponto 2). É neste ponto em que o cliente deve reunir com a equipa de desenvolvimento, comunicando todos os requisitos que deseja para o seu projeto. Por vezes, este pode ter dificuldades em visualizar como será o projeto final e, existem ainda outros casos, em que o cliente não tem uma ideia do que deseja e coloca essa responsabilidade na empresa, sendo que a equipa faz então um levantamento mais abrangente de requisitos. É no decorrer desta tarefa que devem ser definidos os pontos-chave a serem desenvolvidos, sendo necessário programar reuniões de *kickoff*, videoconferências e trocas de informação que fomentem uma melhor comunicação. Para colmatar estes desafios, foi definida uma prática que consistia em dedicar uma hora para realizar uma reunião com o cliente para fazer o levantamento de todos estes requisitos.



**Figura 10** – Diagrama atualizado das tarefas de desenvolvimento de páginas *web* (nova sistemática).

Na análise de requisitos (ponto 3) inicia-se o desenvolvimento do projeto. Assim que a fase inicial do projeto está concluída, deve ser agendada uma nova reunião com o cliente para que, desta forma, as expectativas sejam alinhadas e sejam efetuados os ajustes necessários diminuindo a probabilidade de atrasos devido a mudanças tardias e repentinas no projeto. Isto é uma prática valiosa pois permite que o cliente sinta que faz parte do

desenvolvimento do projeto. De forma a melhorar a comunicação, a empresa tem por prática já adotada, o fornecimento de uma hiperligação de acesso à página *web* para que o cliente possa desta forma acompanhar o seu desenvolvimento. Após validação do desenvolvimento do projeto nessa reunião, o projeto prossegue.

Outro ponto crítico identificado diz respeito aos testes. Este era um ponto subestimado pela empresa que assumia que o desenvolvimento ocorreria sem problemas. No entanto, a experiência evidenciou que essa suposição não correspondia à realidade e que, muitas vezes surgiam *bugs* e outros problemas. Posto isto, foi definido incluir-se a estimativa de um tempo específico, de acordo com a dimensão do projeto (geralmente 20% do tempo total estimado), para realizar testes de qualidade para diminuir possíveis riscos de atrasos devido às correções. Esta percentagem foi assumida com base no estudo de Rizwan & Iqbal (2011).

Outros dois fatores cuja quantificação de tempo é complexa, dada a sua variabilidade de projeto para projeto, é o desenvolvimento desktop (ponto 5.5), a melhoria para mecanismos de busca (ponto 8) e a manutenção e suporte da página *web* (ponto 9), podendo necessitar de 30 a 40 horas de trabalho (Figura 9).

### **5.2.2. Sistemática para criação de lojas *online***

O desenvolvimento de uma loja *online* é semelhante ao desenvolvimento de páginas *web* convencional, com a distinção que uma loja *online* proporciona a capacidade de realizar transações comerciais. Posto isto, há a necessidade de abordar tarefas específicas inerentes a essa etapa específica, como podemos observar na Figura 11.

Desta forma, também foram identificadas subtarefas necessárias para o desenvolvimento de lojas *online* que ainda não tinham sido padronizadas, especificamente em relação à etapa (5), o desenvolvimento propriamente dito. Desta forma foram identificadas as tarefas de integração com sistemas de pagamento, a gestão de *stocks* e de envio, a configuração do carrinho de compras, a personalização da página de detalhes do produto, a *interface* para avaliação dos produtos e a própria configuração da conta do utilizador.

De igual forma ao que foi analisado em relação ao desenvolvimento de páginas *web*, também na construção de lojas *online* foram identificadas algumas tarefas têm tempos padronizáveis e outras tarefas que têm uma enorme variabilidade, dependendo do projeto. Assim sendo, foi aplicada a mesma metodologia para a estimativa dos tempos de execução das tarefas para o desenvolvimento lojas *online*.





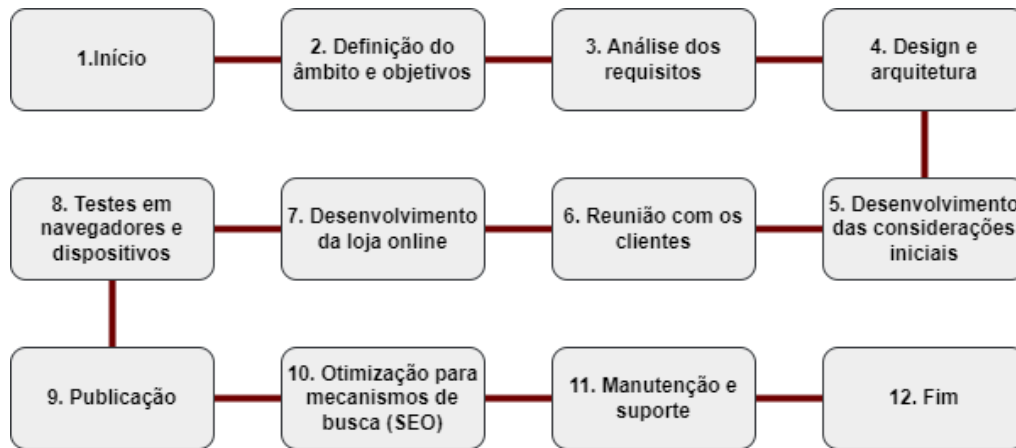
**Figura 11** – Identificação de subtarefas no desenvolvimento de lojas *online*.

Na Tabela 5 apresentam-se os tempos de execução das tarefas para o desenvolvimento de lojas *online*. No desenvolvimento de lojas *online*, as tarefas de análise dos requisitos, a gestão de *stocks* e de envio, a configuração do carrinho de compras, a personalização de detalhes e avaliação do produto, a configuração da conta do utilizador e a publicação podem ser consideradas tarefas padronizáveis. No total, estas representam um tempo estimado de 14 horas. As restantes tarefas dependem da complexidade dos requisitos de cada projeto. Considerando os tempos das tarefas variáveis, foi considerado que o *lead time* de execução pode variar, no limite, entre 30 horas para os projetos de menor complexidade e 129 horas para os projetos de maior complexidade.

**Tabela 5** – Tempos de execução das tarefas para o desenvolvimento de uma loja *online*

| <b>Identificação das tarefas</b>                    | <b>Tempo (horas)</b> |
|---|----------------------|
| 2 Definição do âmbito e objetivos                   | [2; 5]               |
| 3 Análise dos requisitos                            | 1                    |
| 4 <i>Design</i> e arquitetura                       | [8; 24]              |
| 5.1 Integração com sistemas de pagamentos           | [4; 16]              |
| 5.2 Gestão de <i>stock</i>                          | 1                    |
| 5.3 Gestão de envio                                 | 1                    |
| 5.4 Configuração do carrinho de compras             | 2                    |
| 5.5 Personalização da página de detalhes do produto | 4                    |
| 5.6 Configuração da avaliação de produtos           | 1                    |
| 5.7 Configuração da “minha conta”                   | 2                    |
| 6 Testes  | -                    |
| 7 Publicação  | 2                    |
| 8 Melhoria para mecanismos de busca                 | [1; 40]              |
| 9 Manutenção e suporte                              | [1; 30]              |

Verifica-se assim que as tarefas de duração variável e que se tornam uma problemática nas estimativas de *lead time* são as mesmas que foram identificadas no desenvolvimento de páginas *web*, pelo que os métodos e processos implementados em páginas *web*, devem ser os mesmos no desenvolvimento de lojas *online*. A Figura 12, apresenta um diagrama atualizado do processo de desenvolvimento de uma loja *online*.



**Figura 12** – Diagrama atualizado das tarefas de desenvolvimento de lojas *online* (nova sistemática).

### 5.2.3. Sistemática para criação de aplicações móveis

As tarefas de criação de aplicações móveis são muito singulares em cada projeto, pelo que é difícil subdividir em tarefas e criar uma estrutura genérica por meio de um diagrama como os realizados na criação de páginas *web* e nas lojas *online*.

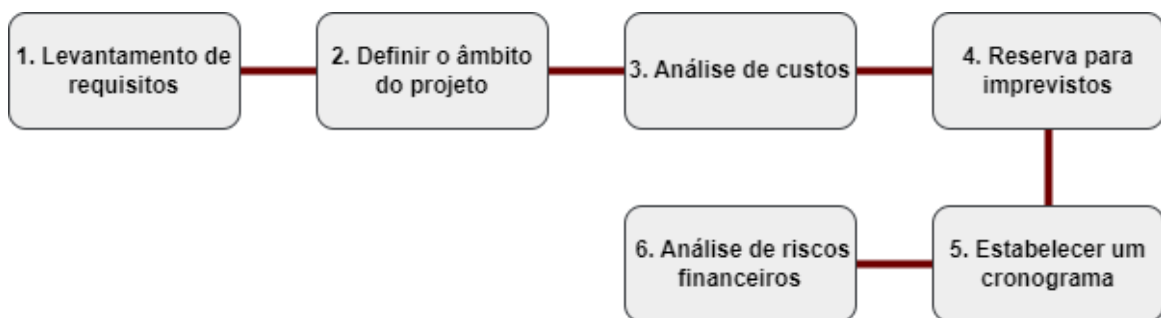
Embora muitas das técnicas e abordagens sugeridas para o desenvolvimento de aplicações sejam comparáveis às usadas para criar páginas *web* ou loja *online*, existe uma distinção significativa no processo de comunicação com o cliente. Distintivamente do desenvolvimento de páginas *web* ou lojas *online*, nos quais os clientes podem acompanhar o progresso do projeto *online* por meio de uma hiperligação, no desenvolvimento de uma aplicação, essa opção não está disponível. Devido a esta limitação, é necessário haver um número maior de reuniões com o cliente para que ele possa acompanhar de perto o desenvolvimento da aplicação. Essas reuniões desempenham um papel fundamental na comunicação eficaz entre a equipa de desenvolvimento e os clientes, pois permite que estes se mantenham atualizados sobre o progresso, realizem avaliações intermediárias e recebam *feedback* importante. Acresce que o *lead time* de execução de uma aplicação pode ser muito superior ao de desenvolvimento de páginas ou lojas *online*. Assim, optou-se por manter o diagrama de tarefas já apresentado no capítulo 4, na Figura 6.

### 5.3. Criação de ferramenta de orçamentação automatizada

No contexto da gestão do portfólio de projetos de uma empresa de desenvolvimento de soluções informáticas, a definição de orçamentos desempenha um papel crucial. A alocação adequada de recursos é essencial para assegurar o sucesso de projetos para manter o equilíbrio entre custo, prazo e qualidade do serviço.

Para alcançar esse equilíbrio, é necessário padronizar a elaboração dos orçamentos. Nesse sentido, foi definido um diagrama das fases que devem ser analisadas na elaboração de qualquer orçamento que a empresa venha a apresentar a um potencial cliente, conforme apresentado na Figura 13. Inicialmente, é crucial realizar um levantamento de requisitos detalhados, ou seja, antes de iniciar qualquer projeto, é essencial realizar um levantamento de requisitos detalhados em colaboração com o cliente. Isso implica uma compreensão abrangente dos objetivos do projeto, funcionalidades desejadas, requisitos técnicos e quaisquer expectativas específicas em relação ao orçamento. Com base no levantamento desta informação, é possível identificar os recursos que são necessários alocar aos projetos e, analisando os diagramas de tarefas, pode-se determinar os tempos de afetação e respectivos custos de mão-de-obra de concepção e desenvolvimento do projeto.

Outro passo crítico na análise de custos na definição do orçamento requer a identificação de todos os custos associados à aquisição de *software* ou *hardware*, licenças, custos operacionais e quaisquer outras despesas adicionais associadas ao desenvolvimento do projeto. Deve-se também alocar uma margem de custo no orçamento para contingências e imprevistos que possam surgir durante o projeto. Isso ajuda a evitar que desvios do orçamento inicial causem problemas significativos e resultem em prejuízos para a empresa ou necessidade de correção extrema de valor.



**Figura 13** – Diagrama de etapas de elaboração da ferramenta de orçamentação.

As estimativas de tempo desempenham um papel crucial na determinação do orçamento e, portanto, devem ser realistas. É essencial estabelecer um cronograma para determinar os custos associados com a mão-de-obra e para planejar o uso eficiente dos recursos financeiros. Por fim, fazer uma análise de riscos financeiros é muito importante para identificar possíveis ameaças ao orçamento. Um dos aspectos mais importantes da proposta de revisão do processo de orçamentação é a inclusão de uma margem financeira que permita amortizar os efeitos dos potenciais imprevistos, nomeadamente, os tempos de teste e a resolução de *bugs*.

Em relação à questão orçamental, devem ser realizadas reuniões com os clientes, com o objetivo de levantamento de requisitos sobre o projeto, permitindo que a empresa elabore um orçamento personalizado. A realização dessas reuniões facilita a incorporação de ajustes no projeto antes da elaboração do orçamento final. Isso evita a inclusão de elementos não essenciais nos orçamentos e garante que sejam competitivos. A Figura 14 ilustra a componente de custeio que foi desenvolvida como parte do portfólio orçamental da empresa. O portfólio auxilia na definição e envio de orçamentos, contendo informações sobre a empresa, exemplos de projetos semelhantes concluídos, detalhes dos serviços incluídos no orçamento e explicação sobre cada serviço, prazo de entrega e contacto. No Apêndice 1 apresenta-se um exemplo de portfólio orçamental da empresa. Este portfólio permite ao cliente conhecer e comparar trabalhos passados da empresa, fornecendo ao cliente informação sobre os potenciais custos de certos requisitos de desenvolvimento.

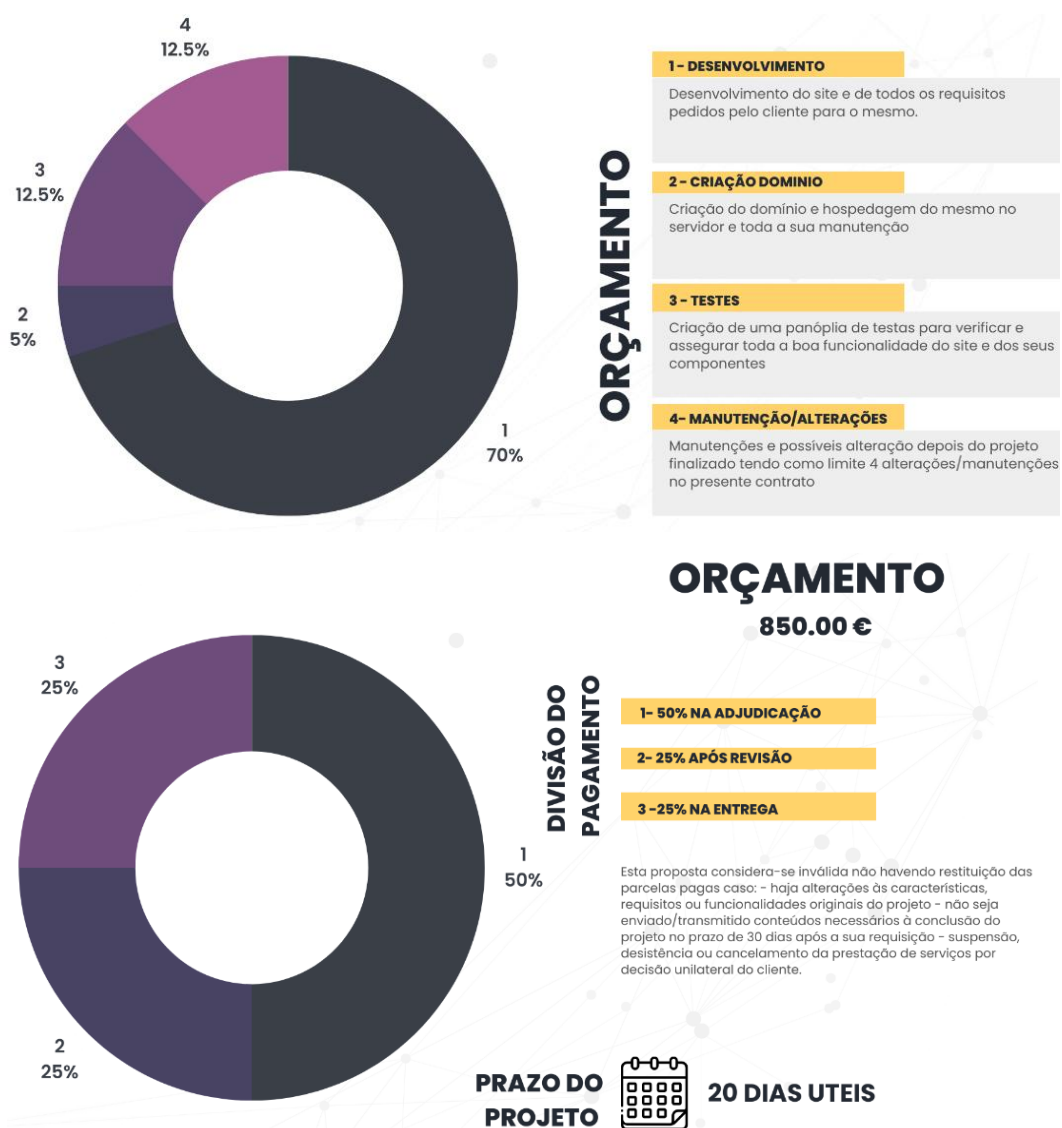


Figura 14 – Exemplo de uma componente de custeio da ferramenta de orçamentação de uma página web - portfólio orçamental da empresa.

## 5.4. Aplicação de ferramentas de gestão *Agile*

O *Scrum* é uma metodologia *Agile* amplamente adotada que se baseia na colaboração da equipa, na comunicação transparente e na flexibilidade para se adaptar às mudanças, sendo proposta a sua implementação. Assim, nesta secção exploram-se os passos essenciais para implementar o *Scrum* através do uso da plataforma *Trello*. Com esta abordagem pretende-se responder a algumas das ações de melhoria referidas na secção 5.1, nomeadamente, a implementação de ferramentas ágeis que simultaneamente permitam registar dados de tempos de execução dos projetos, criando uma base de dados que melhore a gestão das equipas, a atribuição de responsabilidades das diferentes tarefas dos projetos em desenvolvimento. Ao registar tempos e tarefas de forma rigorosa, é possível estimar com mais rigor tempos futuros e, por sua vez, implica um orçamento mais rigoroso.

### 5.4.1. Implementação do *Trello*

A ferramenta escolhida para implementação de gestão *Agile* foi o *Trello* que é uma plataforma de gestão de projetos e tarefas baseada na *web* que utiliza um sistema de quadro *Kanban* para organizar e acompanhar o trabalho. Esta é uma ferramenta amplamente utilizada por equipas e indivíduos para realizar a gestão de tarefas, projetos e compreender os fluxos de trabalho de forma visual e colaborativa. Esta é uma ferramenta altamente flexível e versátil que pode ser usada em diversos contextos, desde a gestão de projetos de desenvolvimento de *software* até à organização de tarefas individuais. O *Trello* tem uma interface simples e intuitiva, o que o torna acessível mesmo para pessoas que não têm experiência prévia em gestão de projetos.

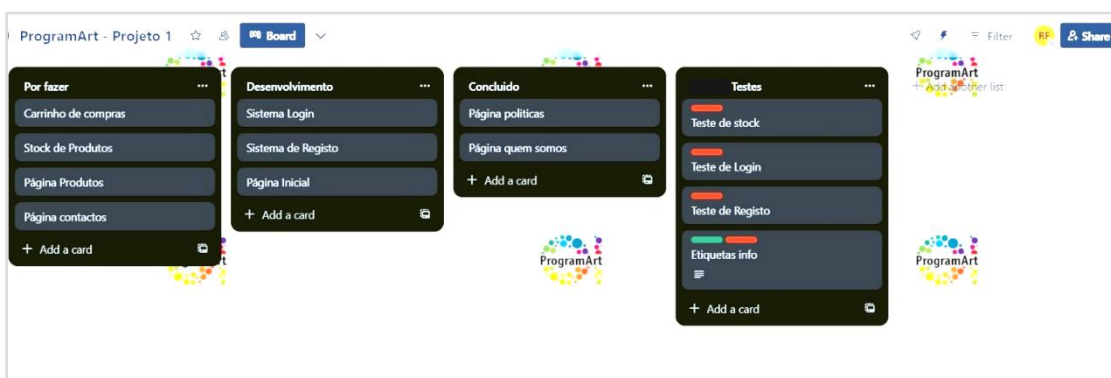
Importa salientar que, uma vez que a equipa da ProgramArt não tinha conhecimento sobre métodos ágeis e as suas características, nomeadamente, a implementações dos diferentes papéis associados ao *Scrum*, foram definidas ações de formação para que a equipa aprendesse estes conceitos. As formações foram realizadas *online* através do *Linkedin learning*.

Para cada projeto em desenvolvimento foi criado um quadro *Kanban* na plataforma *Trello*, dividido em quatro tabelas de informação.

A primeira tabela sintetiza as tarefas do respetivo projeto cujo estado corresponda a “Por fazer”; a segunda tabela corresponde à informação sobre as tarefas em “Desenvolvimento”; enquanto que a terceira tabela identifica as tarefas já “Concluídas”. A última tabela identifica os “Testes” que estão a ser efetuados, identificando com uma

codificação de cores o respetivo estado de validação desses testes. Esta ferramenta está disponível *online* para atualizações de informação em tempo real, de acordo com as responsabilidades atribuídas a cada elemento da equipa. A Figura 15, apresenta um *screenshot* de um exemplo de um quadro *Kanban* na plataforma *Trello* de um projeto da *ProgramArt*.

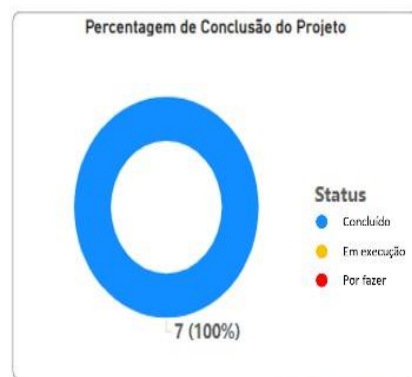
Foram então implementadas reuniões a cada 15 dias para ser planeado o que deveria ser efetuado nos 15 dias seguintes e rever o que se fez nos 15 dias passados. Foi também implementado reuniões diárias para atualizar o resto da equipa do estado do projeto. Apesar da equipa ser relativamente pequena, a implementação de uma ferramenta de gestão deste tipo mostrou-se muito útil para estimativas de tempo mais precisas o que implica orçamentos mais precisos.



**Figura 15** – Quadro *Kanban* na plataforma *Trello* para desenvolvimento de uma loja online.

#### **5.4.2. Criação de *dashboards***

Com os dados que foram registados no *Trello* foi desenvolvido um conjunto de *interfaces* gráficas com o auxílio da ferramenta *Power BI*, permitindo ter uma visão global e abrangente do trabalho desenvolvido nos diferentes projetos. Na Figura 16, apresenta-se um painel de monitorização de um projeto, o qual exibe informações essenciais para avaliar o estado de andamento do projeto. Este painel oferece uma visão clara da percentagem de conclusão do projeto, permitindo uma avaliação instantânea do progresso alcançado em relação às metas estabelecidas. De ressaltar que, a percentagem de conclusão do exemplo encontra-se a 100%, uma vez que o projeto já se encontra concluído.



### Responsáveis por cada tarefa

| Tarefa   | Primeiro Responsável |
|--|----------------------|
| 1 - Desenvolvimento do Design  | -                    |
| 1.1 Pagina Inicial   | José                 |
| 1.2 Pagina dos serviços prestados                                    | José                 |
| 1.3 Pagina dos Contactos   | José                 |
| Adaptação para mobile e outros dispositivos                          | Nuno                 |
| Colocar todas os hiperligações das rede sociais aos icons respetivos | José                 |
| Testes de Qualidade  | André                |



### Responsáveis por cada resolução de BUG

| ID do Bug | Primeiro Responsável pela Resolução | Primeiro Responsável pela tarefa |
|-----------|-------------------------------------|----------------------------------|
| BUG-002   | José                                | José                             |
| BUG-003   | José                                | José                             |

Figura 16 – Dashboard em Power BI de monitorização de um projeto.



O *dashboard* disponibiliza também informações sobre o tempo de execução de cada tarefa e a identificação do responsável pela sua realização. O *Gantt* fornecido na *dashboard* oferece uma representação visual da linha do tempo do projeto.

No mesmo painel, são apresentados dados relacionados ao número de *bugs* registados durante o desenvolvimento do projeto, bem como o tempo necessário para a resolução de cada um deles. Estes dados são bastante importantes pois permite uma análise detalhada do desempenho, possibilitando a comparação entre o tempo efetivamente gasto e o tempo estimado inicialmente para a conclusão das tarefas. Estes registos são fundamentais pois permite tomar melhores decisões no futuro, uma vez que possibilita uma avaliação precisa das tendências e projeções com base em projetos passado.

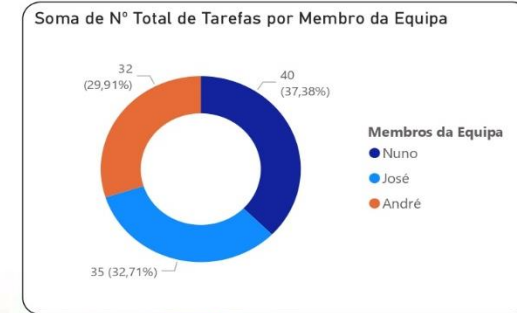
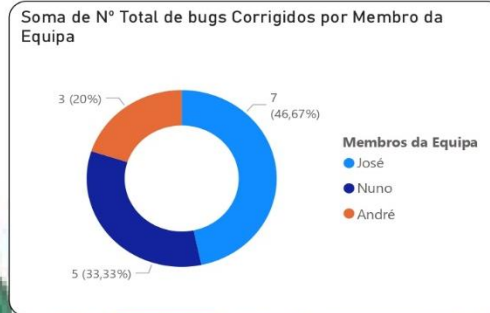
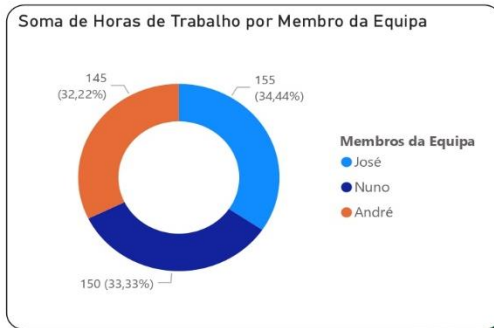
As informações fornecidas pelo *dashboard*, juntamente com as atualizações diárias do *Kanban* elaborado no *Trello*, são cruciais para a análise do progresso do projeto em relação aos prazos estabelecidos. Essa análise permite tomar decisões informadas, incluindo a possibilidade de ajustar a prioridade de determinadas tarefas, a fim de assegurar a conclusão do projeto ou das funcionalidades principais dentro dos prazos definidos, evitando assim atrasos na entrega final ao cliente.

Dado que a empresa está envolvida em múltiplos projetos de diversas tecnologias, a avaliação dos colaboradores em termos de eficiência (número de *bugs* corrigidos) e eficácia (conclusão bem-sucedida do projeto) torna-se uma tarefa desafiadora.

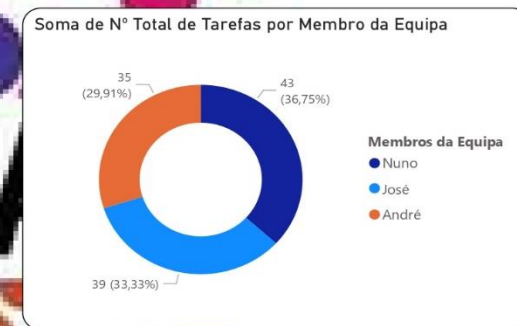
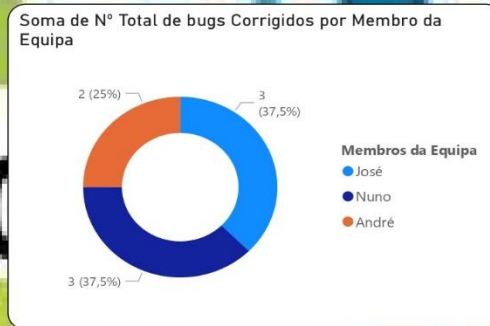
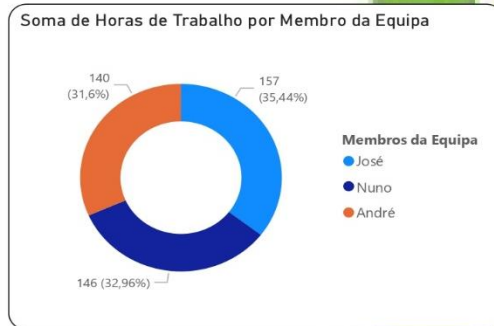
Como forma de monitorização de produtividade da empresa, mais especificamente de cada desenvolvedor, criou-se um *dashboard* para agregação da informação de desempenho dos colaboradores, através da identificação de KPI ajustados ao contexto de uma empresa de desenvolvimento de soluções informáticas. Esse *dashboard* foi desenvolvido a partir da integração da informação do *Trello*, onde cada tarefa a realizar é atribuída a um dos desenvolvedores, bem como o seu tempo de realização das tarefas. Estes dados foram extraídos colocados numa folha de cálculo em *MS Excel* e, com recurso ao *Power BI* gerou-se o *dashboard* apresentado na Figura 17. Como se pode verificar, através do *dashboard* é possível analisar numa ótica mensal, para cada um dos colaboradores os seguintes indicadores:

- Percentagem relativa do número total de horas trabalhadas em tarefas de projeto;
- Percentagem relativa do número total tarefas realizadas;
- Quantidade de *bugs* solucionados.

## 1º Mês



## 2º Mês



## 3º Mês

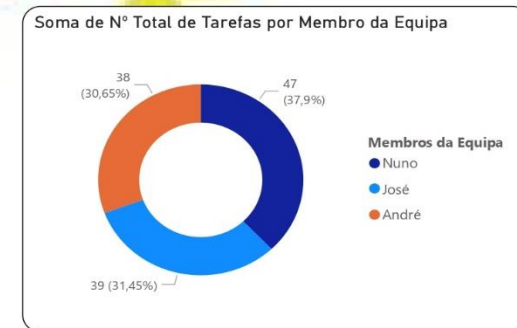
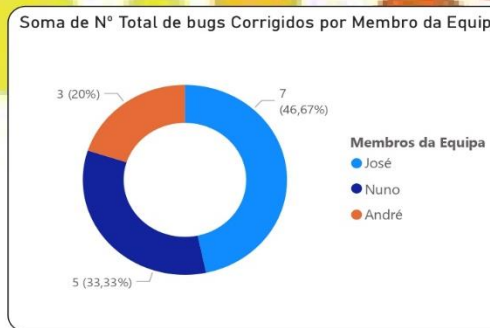
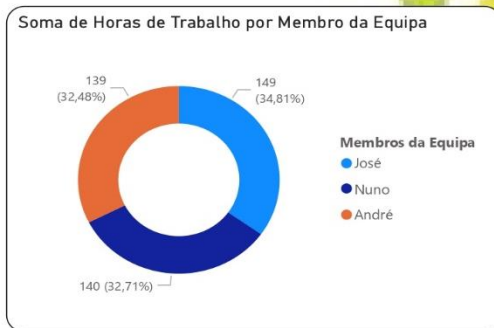


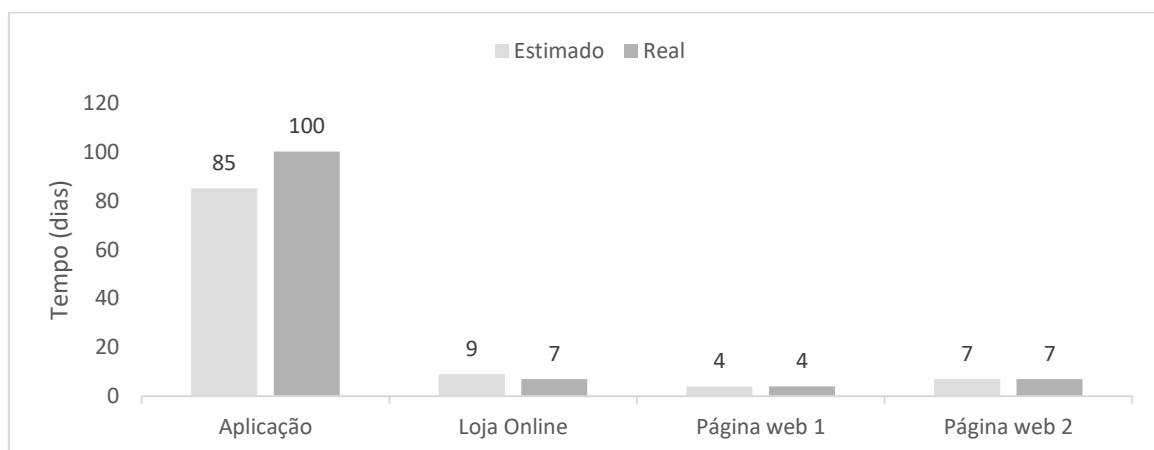
Figura 17 – Dashboard em Power BI de análise de desempenho dos colaboradores da ProgramArt.

Contudo uma vez que a empresa desenvolve diferentes opções de serviços, sendo estes distintos um dos outros, tanto a nível de complexidade como a nível de tempo de execução, a avaliação mês-a-mês pode não ser conclusiva, uma vez que em alguns meses os projetos em andamento podem ser mais complexos do que em outros. Todavia, é uma forma objetiva de quantificar desempenhos em relação às novas medidas impostas na empresa. Para fins de controlo de divisão de trabalho este *dashboard* pode também ser útil, uma vez que é atualizado diariamente e que em cada *sprint* podem ser reavaliadas as atribuições das tarefas em função das suas taxas de execução e da disponibilidade dos desenvolvedores.

### 5.5. Análise de resultados das propostas de melhoria

A análise dos tempos é essencial na aplicação de metodologias como *Scrum* na gestão de projetos de desenvolvimento de páginas *web*, lojas *online* e aplicações móveis. A medição, análise e padronização dos tempos contribuem para melhorias contínuas, aumentam a previsibilidade de execução dos projetos e promovem a eficiência ao longo dos seus ciclos de vida. Isso permite que as empresas alcancem resultados consistentes e atendam às expectativas dos clientes de forma eficaz.

Após a implementação das sistemáticas, da reorganização na empresa e da adoção das ferramentas de gestão *Agile* de projetos apresentados, foi possível ainda acompanhar quatro novos projetos: um projeto de desenvolvimento de aplicação móvel, um projeto de loja *online* e dois projetos de conceção de páginas *web*. Os tempos estimados pela equipa para a conclusão de cada projeto e os tempos efetivos estão representados na Figura 17.



**Figura 18** – Estimativas de tempo e tempo real de diferentes projetos depois de implementadas as melhorias.

Com a implementação das propostas de melhoria, verificou-se que a discrepância entre o tempo estimado e o tempo de execução do desenvolvimento de uma aplicação móvel correspondeu a 17,6%, um valor muito inferior aos registados antes do desenvolvimento dos processos de melhoria (37% para uma aplicação móvel). No caso do projeto de desenvolvimento da loja *online*, verificou-se inclusive que o tempo real de execução foi inferior do que o tempo estimado. No caso dos dois projetos de conceção de página *web*, o tempo estimado e o tempo de execução foram os mesmos. Não obstante estas constatações, considerando que as propostas apenas foram implementadas em quatro projetos, é importante notar que as conclusões devem ser interpretadas com cuidado, dado o tamanho limitado da amostra de projetos acompanhados. No entanto, verificam-se melhorias significativas nos tempos de desempenho, após a implementação das propostas.

Apesar da redução entre tempo estimado e tempo de execução no desenvolvimento de aplicações móveis, este tipo de serviço prestado pela ProgramArt deve ser analisado de forma mais cuidada dados os elevados *lead times* que apresentam quando comparados com as páginas *web* e lojas *online*.

Uma vez que, processualmente, este tipo de solução informática é mais complicado de ser acompanhado por parte do cliente, é comum existirem discrepâncias entre o objetivo e resultado esperado pelo cliente e a solução prática implementada. Tomando-se como exemplo o projeto de aplicação da Figura 18, no início do projeto, o cliente tinha uma ideia de conceção bastante sólida do que desejava, apesar da equipa perceber desde cedo que essa abordagem não seria a mais adequada. No entanto, dada a relutância do cliente em receber sugestões, a equipa procedeu ao levantamento de todos os requisitos e desenvolveu a aplicação conforme solicitado. Em nova reunião, o cliente percebeu que a implementação prática não refletia a melhor solução para seu objetivo. Desta forma, o projeto acabou por regressar à fase inicial, onde foi necessário refazer o levantamento e análise de requisitos. Nesse ponto, ficou evidente que o tempo estimado inicialmente não seria suficiente, tendo o cliente concordado em prolongar o prazo para a conclusão do projeto.

Nesta fase, a equipa reconheceu a necessidade de criar *mockups*, como na Figura 19, para ser possível levar para reuniões com os clientes. Esses *mockups* serviram como representações visuais do projeto no estado final, permitindo uma compreensão clara de como o projeto se pareceria quando concluído. Essa abordagem evitou o desperdício de tempo na implementação e programação de algo que poderia ser posteriormente modificado, como no exemplo apresentado.

A criação de *mockups* permitiu assim identificar problemas, ajustar detalhes e fazer melhorias antes do início efetivo do trabalho. Isso facilitou a comunicação entre o cliente e a equipa, além de validar a aparência e o funcionamento do projeto antes da sua construção efetiva.

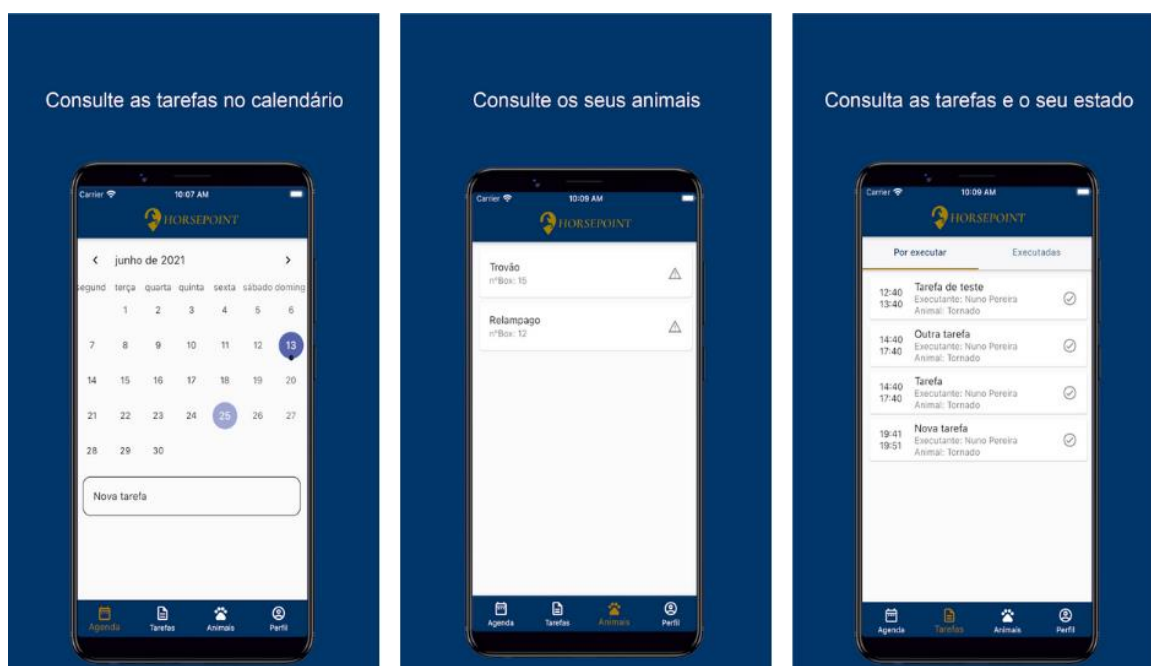


Figura 19 – *Mockups* reais apresentados ao cliente.

A partir do momento que a empresa passou a desenvolver estes *mockups*, passou a ser mais fácil a sincronização entre as estimativas de tempo de desenvolvimento de aplicações móveis, as expectativas do cliente e os tempos reais de execução.

Em suma, com a implementação das propostas de melhoria, em qualquer um dos três tipos de serviços prestados pela ProgramArt, foi possível um melhor alinhamento das expectativas do cliente com as da equipa de desenvolvimento. O melhor conhecimento das tarefas, as estimativas de tempo de execução mais assertivas e a melhoria do processo de comunicação e integração dos clientes durante as fases de desenvolvimento trouxeram uma maior flexibilidade e capacidade de resposta da empresa para colmatar de forma mais eficiente as solicitações do mercado de soluções informáticas.

Esta abordagem demonstrou também ser eficaz na gestão interna do portfólio de projetos, na gestão das equipas de desenvolvimento e na identificação de indicadores de desempenho para uma monitorização em tempo real.

## 6. Conclusões

Neste capítulo, são apresentadas as principais conclusões derivadas da dissertação sobre a aplicação de *Scrum* e técnicas *Lean* na gestão do portfólio de projetos de uma empresa de desenvolvimento de soluções informáticas. Além disso, são delineadas propostas de trabalho futuro com base nas descobertas e nos desafios identificados ao longo do estudo.

### 6.1. Principais conclusões

A dissertação abordou a implementação de metodologias *Agile*, como *Scrum* e técnicas *Lean*, numa empresa de desenvolvimento de *software*, a ProgramArt. Estas ferramentas e metodologias de gestão facilitam a comunicação mais eficaz entre a equipa de desenvolvimento e o cliente. Por exemplo, o *Trello* permite a criação de quadros personalizados que mostram o fluxo de trabalho de cada projeto, o que facilita o acompanhamento das tarefas e a localização de problemas. O *Scrum* e o *Kanban* permitem que as equipas tenham reuniões regularmente para discutir o progresso e ajustar o plano de execução, aumentando a eficiência da equipa. Assim, a aplicação de ferramentas e metodologias de gestão *Agile* ajuda as empresas a alcançar maior eficiência, qualidade e satisfação do cliente. Fatores como a complexidade do projeto, o tamanho e a experiência da equipa, os requisitos do cliente, muitas vezes, em constante mudança, assim como a integrações com serviços adicionais aos projetos, podem influenciar a estimativa de orçamentação da empresa.

A transformação digital na empresa foi notável com a digitalização da informação produzida pela empresa, contendo dados relativos a projetos, tarefas, tempos e responsáveis. Esse nível de detalhe e precisão na recolha de informação representou um avanço significativo em relação às práticas anteriores da empresa.

A normalização e sistematização implementadas permitiu à empresa obter informações minuciosas e rigorosas sobre suas operações e projetos. Essas ferramentas proporcionaram à empresa a capacidade de gerar orçamentos e propostas mais precisas e competitivas, resultando em maior eficiência e qualidade na prestação dos seus serviços.

## 6.2. Propostas de trabalho futuro

Devido à limitação do tempo de aplicação das propostas sugeridas, não foi possível reunir dados em quantidade suficiente para estabelecer uma validação conclusiva da adequação desta metodologia. Não obstante, essas propostas têm o potencial de melhorar substancialmente a qualidade dos orçamentos, das estimativas de lead time de execução e, conseqüentemente, do desempenho financeiro da empresa. Com um conjunto de dados mais extenso, seria possível identificar padrões e lacunas nos processos de desenvolvimento dos variados projetos em execução.

Sugere-se a possibilidade de automatizar por meio de uma *Application Programming Interface* (API) a transferência de dados entre a plataforma *Trello*, onde os valores são inseridos, e os painéis de controlo construídos no *software Power BI*. Essa integração facilitaria a recolha e análise de informações, tornando o processo mais eficiente e preciso.

Também devem ser desenvolvidos mais *dashboards* que permitam identificar mais medidas de monitorização, como por exemplo em relação à faturação da empresa, criando um maior controlo em todos os âmbitos da empresa.

Em relação aos *dashboards* já desenvolvidos, é necessário dividi-los em três, ou seja, criar um *dashboard* específico para páginas *web*, aplicações móveis e lojas *online*. Só assim é possível ter um controlo mais minucioso e também avaliar de forma mais conclusiva a evolução da empresa ao longo do tempo, tanto a nível de produtividade, como na diminuição de *bugs*, como no aumento da rapidez de execução das tarefas entre outros.

Adicionalmente, é fundamental manter o portfólio criado no âmbito desta dissertação atualizado de maneira regular. Isso garantirá que a empresa está alinhada com as necessidades dos seus clientes, demonstrando um compromisso contínuo com a excelência e a qualidade de seus serviços.

Por último, em relação ao desenvolvimento de aplicações móveis, considerando que este serviço tem uma duração muito superior à duração do desenvolvimento de páginas *web* e lojas *online*, sugere-se como trabalho futuro a realização de testes intercalares a versões não finais das aplicações.

## Referências bibliográficas

- Ahmad, M. O., Markkula, J., & Oivo, M. (2013). *Kanban in software development: A systematic literature review. Proceedings - 39th Euromicro Conference Series on Software Engineering and Advanced Applications, SEAA 2013*, 9–16. <https://doi.org/10.1109/SEAA.2013.28>
- Al-Saqqa, S., Sawalha, S., & Abdelnabi, H. (2020). *Agile Software Development: Methodologies and Trends. International Journal of Interactive Mobile Technologies (IJIM)*, 14(11), 246–270. <https://doi.org/10.3991/IJIM.V14I11.13269>
- Anderson, D., Concas, G., Lunesu, M. I., & Marchesi, M. (2011). Studying *Lean-Kanban* approach using *software* process simulation. *Lecture Notes in Business Information Processing*, 77 *LNBIP*, 12–26. [https://doi.org/10.1007/978-3-642-20677-1\\_2/COVER](https://doi.org/10.1007/978-3-642-20677-1_2/COVER)
- Berezutskyi, I., Tsiutsiura, S., Rusan, I., Sachenko, I., & Danylyshyn, S. (2023). *Disadvantages of Using Scrum Model in IT Projects*. 89–93. <https://doi.org/10.1109/SIST58284.2023.10223589>
- Coutinho, C. P., Sousa, A., Dias, A., Bessa, F., Ferreira, M. J. R. C., & Vieira, S. R. (2009). *Investigação-acção: metodologia preferencial nas práticas educativas*. <https://repositorium.sdum.uminho.pt/handle/1822/10148>
- da Graça Portela Lisbôa, M., & Pentiado Godoy, L. (2012). APLICAÇÃO DO MÉTODO 5W2H NO PROCESSO PRODUTIVO DO PRODUTO: A JOIA. *Iberoamerican Journal of Industrial Engineering*, 7, 32–47.
- Damij, N., & Damij, T. (2021). An Approach to Optimizing *Kanban* Board Workflow and Shortening the Project Management Plan. *IEEE Transactions on Engineering Management*. <https://doi.org/10.1109/TEM.2021.3120984>
- Dingsøy, T., Bjørnson, F. O., Schrof, J., & Sporse, T. (2023). A longitudinal explanatory case study of coordination in a very large development programme: the impact of transitioning from a first- to a second-generation large-scale *Agile* development method. *Empirical Software Engineering*, 28(1). <https://doi.org/10.1007/S10664-022-10230-6>
- Dombrowski, U., & Zahn, T. (2011). Design of a *Lean* development framework. *IEEE International Conference on Industrial Engineering and Engineering Management*, 1917–1921. <https://doi.org/10.1109/IEEM.2011.6118249>



- Dyba, T., & Dingsoyr, T. (2009). What do we know about *Agile software* development? *IEEE Software*, 26(5), 6–9. <https://doi.org/10.1109/MS.2009.145>
- Dyba, T., & Dingsoyr, T. (2015). *Agile Project Management: From Self-Managing Teams to Large-Scale Development*. *Proceedings - International Conference on Software Engineering*, 2, 945–946. <https://doi.org/10.1109/ICSE.2015.299>
- Feng, P. P., & Ballard, G. (2008). *Standard Work From a Lean Theory Perspective* (pp. 703–712). <http://p2sl.berkeley.edu/>
- Firdaus, M. B., Patulak, I. M., Tejawati, A., Bryantama, A., Putra, G. M., & Pakpahan, H. S. (2019). *Agile-Scrum Software Development Monitoring System*. *ICEEIE 2019 - International Conference on Electrical, Electronics and Information Engineering: Emerging Innovative Technology for Sustainable Future*, 288–293. <https://doi.org/10.1109/ICEEIE47180.2019.8981471>
- Ikumapayi, O. M., Akinlabi, E. T., Mwema, F. M., & Ogbonna, O. S. (2020). Six sigma versus *Lean* manufacturing – An overview. *Materials Today: Proceedings*, 26, 3275–3281. <https://doi.org/10.1016/J.MATPR.2020.02.986>
- Klock, A. C. T., Gasparini, I., & Pimenta, M. S. (2016). 5W2H framework: A guide to design, develop and evaluate the user-centered gamification. *ACM International Conference Proceeding Series, Part F128046*. <https://doi.org/10.1145/3033701.3033715>
- Kothapalli, Dr. S. R., & Nanda M, P. (2019). Productivity Improvement Through *Lean* Adoption In CMMI Appraised IT Services Organization In Bangalore. *Think India Journal*, 22(10), 1368–1375. <https://thinkindiaquarterly.org/index.php/think-india/article/view/10977>
- Lalmi, A., Fernandes, G., & Souad, S. B. (2021). A conceptual hybrid project management model for construction projects. *Procedia Computer Science*, 181, 921–930. <https://doi.org/10.1016/J.PROCS.2021.01.248>
- MacHado, M., Salerno, L., Marczak, S., & Bastos, R. (2020). Assessment Models for Evaluating the Combined use of *Agile*, User-Centered Design and *Lean* Startup in the Context of *Software* Development: A Grey Literature Review. *ACM International Conference Proceeding Series*. <https://doi.org/10.1145/3439961.3439970>
- Mccutcheon, M. R., & Bunch, W. (2021). *The Advantages of Managing IT Projects Using Agile / Scrum*.

- Middleton, P. (2001). *Lean Software Development: Two Case Studies*. *Software Quality Journal*, 9(4), 241–252. <https://doi.org/10.1023/A:1013754402981/METRICS>
- Nicholas, J. M. (2018). *Lean production for competitive advantage: a comprehensive guide to Lean methodologies and management practices* (2nd ed.). *Lean Production for Competitive Advantage*, <https://doi.org/10.4324/9781351139083>
- Oomen, S., De Waal, B., Albertin, A., Ravesteyn, P., & Waal, D. (2017). *HOW CAN SCRUM BE SUCCESSFUL? COMPETENCES OF THE SCRUM PRODUCT OWNER*. [http://aisel.aisnet.org/ecis2017\\_rp/9](http://aisel.aisnet.org/ecis2017_rp/9)
- Oza, N., Fagerholm, F., & Munch, J. (2013). How does *Kanban* impact communication and collaboration in *software engineering* teams? *2013 6th International Workshop on Cooperative and Human Aspects of Software Engineering, CHASE 2013 - Proceedings*, 125–128. <https://doi.org/10.1109/CHASE.2013.6614747>
- Pinto, J. P. (2014). *Pensamento Lean - ISBN: 978-989-752-032-7*.
- Rizwan, M., & Iqbal, M. (2011). Application of 80/20 rule in *software engineering* Rapid Application Development (RAD) model. *Communications in Computer and Information Science*, 181 CCIS(PART 3), 518–532. [https://doi.org/10.1007/978-3-642-22203-0\\_45/COVER](https://doi.org/10.1007/978-3-642-22203-0_45/COVER)
- Saunders, M., Lewis, P., & Thornhill, A. (2019). Research Methods for Business Students. In *Research Methods for Business Students* (8th Edition).
- Senabre Hidalgo, E. (n.d.). *Adapting the Scrum framework for Agile project management in science: case study of a distributed research initiative*. <https://doi.org/10.1016/j.heliyon.2019>
- Shirokova, S., Kislova, E., Rostova, O., Shmeleva, A., & Tolstrup, L. (2020). Company efficiency improvement using *Agile* methodologies for managing IT projects. *ACM International Conference Proceeding Series*. <https://doi.org/10.1145/3446434.3446465>
- Stettina, C. J., & Hörz, J. (2015). *Agile* portfolio management: An empirical perspective on the practice in use. *International Journal of Project Management*, 33(1), 140–152. <https://doi.org/10.1016/J.IJPROMAN.2014.03.008>
- Stoica, M., Mircea, M., & Ghilic-Micu, B. (2013). *Software Development: Agile vs. Traditional*. *Informatica Economică*, 17(4). <https://doi.org/10.12948/issn14531305/17.4.2013.06>

- Tam, C., Moura, E. J. da C., Oliveira, T., & Varajão, J. (2020). The factors influencing the success of on-going *Agile software* development projects. *International Journal of Project Management*, 38(3), 165–176.  
<https://doi.org/10.1016/J.IJPROMAN.2020.02.001>
- Verwijs, C., & Russo, D. (2023). A Theory of *Scrum* Team Effectiveness. *ACM Transactions on Software Engineering and Methodology*, 32(3).  
<https://doi.org/10.1145/3571849>

# Apêndice 1 – Exemplo de portfolio orçamental

**PROGRAMART** BUSINESS  
**APRESENTAÇÃO ORÇAMENTAL**  
@programart

### QUEM SOMOS?

A Programart conecta, descompõe, implementa e monitoriza estratégias para as diferentes unidades operacionais. Somos uma empresa multiplataforma que trabalha para a otimização da experiência do utilizador e a eficiência dos processos. Somos a Programart, uma empresa que trabalha para a otimização da experiência do utilizador e a eficiência dos processos. Somos a Programart, uma empresa que trabalha para a otimização da experiência do utilizador e a eficiência dos processos.

### Cientes

- Mercado de ações
- ISA
- ISA CAMILLANA

### REQUISITOS DO NOSSO CLIENTE

- ✓ Integração de um design personalizado
- ✓ Implementação de um design em ReactJS
- ✓ Ferramentas personalizadas
- ✓ Todos os requisitos estão ligados entre si (dependências e testes)

### REQUISITOS DO NOSSO CLIENTE

- ✓ Integração de um design personalizado
- ✓ Implementação de um design em ReactJS
- ✓ Ferramentas personalizadas
- ✓ Todos os requisitos estão ligados entre si (dependências e testes)

### REQUISITOS DO NOSSO CLIENTE

- ✓ Configuração de plugins para adicionar mais itens à lista de artigos
- ✓ Configuração de plugins para melhorar a navegação (por exemplo a WooCommerce)
- ✓ Configuração de sistema de email (integração mensagens via WhatsApp)
- ✓ Integração com redes sociais (sincronização de produtos e redes sociais)

### ORÇAMENTO

R\$50.000 €

**DIVERSÃO DO PAGAMENTO**

- 1 - 90%
- 2 - 20%
- 3 - 28%
- 4 - 5%

**PRAZO DO PROJETO**

20 DIAS ÚTIS

### OBRIGADO POR RECEBER NOSSA PROPOSTA!

955 4 6175  
@programart  
Rua Luis Barreto, nº 233 P Lateral Sala A, 4790-95,  
Vila Home de Fátima

Figura 20 – Portfolio orçamental.