

# **Desenvolvimento de uma plataforma de gestão de projetos**

**Blended4Future**

**1201367 Carlos Rodrigo Marques dos Santos**

**Orientador: Professor Nuno Escudeiro**

Porto, Setembro, 2025

# Resumo

Trabalhos escritos em língua Inglesa devem incluir um resumo alargado com cerca de 1000 palavras, ou duas páginas.

Se o trabalho estiver escrito em Português, este resumo deveria ser em língua Inglesa, com cerca de 200 palavras, ou uma página.

Para alterar a língua basta ir às configurações do documento no ficheiro `main.tex` e alterar para a língua desejada ('english' ou 'portuguese')<sup>1</sup>. Isto fará com que os cabeçalhos incluídos no template sejam traduzidos para a respetiva língua.

**Palavras-chave:** Frontend, Backend, Software Architecture,

---

<sup>1</sup>Alterar a língua requer apagar alguns ficheiros temporários; O target **clean** do **Makefile** incluído pode ser utilizado para este propósito.

# Agradecimientos

The optional Acknowledgment goes here. . . Below is an example of a humorous acknowledgment.

"I'd also like to thank the Van Allen belts for protecting us from the harmful solar wind, and the earth for being just the right distance from the sun for being conducive to life, and for the ability for water atoms to clump so efficiently, for pretty much the same reason. Finally, I'd like to thank every single one of my forebears for surviving long enough in this hostile world to procreate. Without any one of you, this book would not have been possible."in "The Woman Who Died a Lot"by Jasper Fforde.

# Conteúdo

# Lista de Figuras

## Lista de Tabelas

# Lista de Símbolos





# Capítulo 1

## Introdução

### 1.1 Enquadramento

Este relatório foi desenvolvido com base num projeto enquadrado no âmbito da unidade curricular de Projeto Estágio (PESTI) da Licenciatura em Engenharia Informática (LEI) do Instituto Superior de Engenharia do Porto (ISEP)

Este projeto ocorreu no enquadramento do Projeto BlendED (também referido como Blended4Future ou BlendedMobility<sup>1</sup>). Este curso dá a estudantes a oportunidade de desenvolverem as suas *soft* e *hard skills* num projeto com alunos de diferentes culturas e países trabalhando num projeto para empresas reais

#### 1.1.1 Apresentação da organização

BlendedMobility é uma iniciativa educativa internacional que promove o desenvolvimento de projetos colaborativos no contexto do ensino híbrido. Esta visa combinar experiências de aprendizagem presencial e online, proporcionando aos estudantes uma formação mais flexível, personalizada e orientada para a prática.

O programa reúne alunos de diferentes universidades europeias que, ao longo de quatro meses, trabalham em conjunto no desenvolvimento de projetos reais para empresas parceiras. A metodologia adotada assenta em práticas ativas e colaborativas, potenciando competências essenciais como trabalho em equipa, comunicação intercultural e resolução de problemas num ambiente profissional simulado.

O percurso inicia-se com uma semana presencial no Instituto Universitário da Maia (ISMAI), onde as equipas se conhecem, recebem o enquadramento do projeto e planeiam as etapas de trabalho. O desenvolvimento dos projetos decorre num regime híbrido, combinando sessões online e trabalho autónomo. Ao final do ciclo, todas as equipas reúnem-se presencialmente na Universidade de Trier, na Alemanha, para apresentar os resultados dos seus projetos a um painel de avaliadores e representantes das empresas envolvidas.

### 1.2 Descrição do Problema

O Website do curso Blended4Future estava muito aquém do esperado, muitos elementos seguiam um design inconsistente e antiquado, ou não estavam completamente funcionais.

---

<sup>1</sup><https://blendedmobility.com>

A organização desejava uma plataforma onde se pudesse automaticamente adicionar projetos, alunos, universidades e empresas em uma só plataforma. Por tal foi colocada um proposta de desenvolvimento de uma nova aplicação web que substituiria esta anterior.

Nesta aplicação, estudantes professores e representantes de empresas poderiam ver os projetos em que estavam envolvidos, fazer posts sobre os seus respetivos projetos.

Foi ainda requisitado uma maneira de ver todas a edições do Blended4Future e todos o projetos neste envolvido

## 1.3 Objetivos

A aplicação web a desenvolver deverá incluir um sistema de autenticação com diferenciação entre perfis de utilizador, nomeadamente administradores e utilizadores comuns, assegurando uma gestão adequada de permissões.

Adicionalmente, deverá permitir a criação de novos projetos e a associação de diferentes intervenientes a cada um, consoante o seu papel. Para além disso, deverá ser implementada uma biblioteca de projetos, acessível a qualquer utilizador da plataforma, onde estarão disponíveis todos os projetos desenvolvidos no âmbito do curso, promovendo a sua consulta e divulgação.

A interface da aplicação deverá seguir as diretrizes de design definidas previamente por um membro da equipa dedicado ao design visual da aplicação.

## 1.4 Abordagem

### 1.4.1 A equipa

A equipa foi composta por um grupo de estudantes provenientes de várias universidades europeias, com a seguinte constituição:

- 5 Desenvolvedores
- 1 Designer
- 1 Marketer

A equipa de IT, constituída por cinco desenvolvedores, foi organizada em duas subequipas:

- 3 estudantes dedicados ao desenvolvimento de *Backend*
- 2 estudantes dedicados ao desenvolvimento de *Frontend*, incluindo o autor deste relatório, que integrou esta subequipa para participar no desenvolvimento da interface de utilizador.

Esta divisão teve como objetivo garantir um maior avanço na lógica de negócio durante a fase inicial do projeto. Numa etapa posterior, quando a lógica estivesse próxima da sua conclusão, alguns dos estudantes poderiam transitar para a subequipa de *Frontend*, focando-se então na criação da interface de utilizador.

É importante salientar que, no âmbito do projeto Blended4Future, cada estudante participou com um número distinto de créditos (ECTS), definidos em função do seu respetivo curso.

Assim, tornou-se necessário organizar a distribuição das tarefas de forma proporcional, garantindo que a carga de trabalho atribuída a cada elemento refletisse adequadamente o valor dos créditos em que estava inscrito.

### 1.4.2 Metodologia de trabalho

Para uma melhor organização do trabalho, foi adotada a metodologia Scrum, com sprints de duas semanas de duração. Além disso, foi estabelecida, por consenso do grupo, a realização de reuniões semanais com o objetivo de atualizar o progresso das tarefas e promover um ambiente de trabalho mais colaborativo e comunicativo. Estas reuniões não possuíam uma data definida tendo em conta os fusos horários de cada um dos membros

#### Divisão de sprints

Sprint	Semanas	Data de início	Data de fim	Objetivos definidos
1	1-2	24/02	02/03	1. Documentação geral do projeto
2	3-4	10/03	23/03	2. Configuração dos repositórios de Backend/Frontend com pipelines de CI/CD
3	5-6	24/03	06/04	3. Finalização da documentação geral
4	7-8	07/04	20/04	3. Configuração de DevOps/Azure 2. Início do desenvolvimento
5	9-10	21/04	04/05	4. Análise SWOT do negócio
6	11-12	05/05	18/05	
7	13-14	19/05	01/06	
8	15-16	02/06	15/06	

Tabela 1.1: Organização de sprints definida pela equipa

A tabela ?? demonstra a organização de sprints que foi escolhida pela equipa durante a primeira semana de *kickoff* do projeto.

### 1.4.3 Tecnologias Definidas

Na referida semana de kickoff foi necessária uma deliberação sobre as tecnologias e ferramentas a utilizar. Durante o desenvolvimento foram adicionadas ainda outras com o intuito de agilizar ainda mais este.

#### Ferramentas de Gestão de projetos

Primeiramente foi necessário definir uma plataforma central para onde cada elemento do grupo pudesse ver o estado do projeto, incluindo ainda a totalidade do *product backlog*, as suas respetivas tarefas, duração dos sprints e onde se pudesse ter acesso a todos os repositórios necessários ao projeto.

Por aconselhamento dos professores membros dos projeto foi escolhido a aplicação da *Azure DevOps*. (Ver fig. ?? e ??)

Esta ferramenta destaca-se como uma plataforma integrada que oferece um equilíbrio entre funcionalidades avançadas e facilidade de utilização, suportando todo o ciclo de vida do

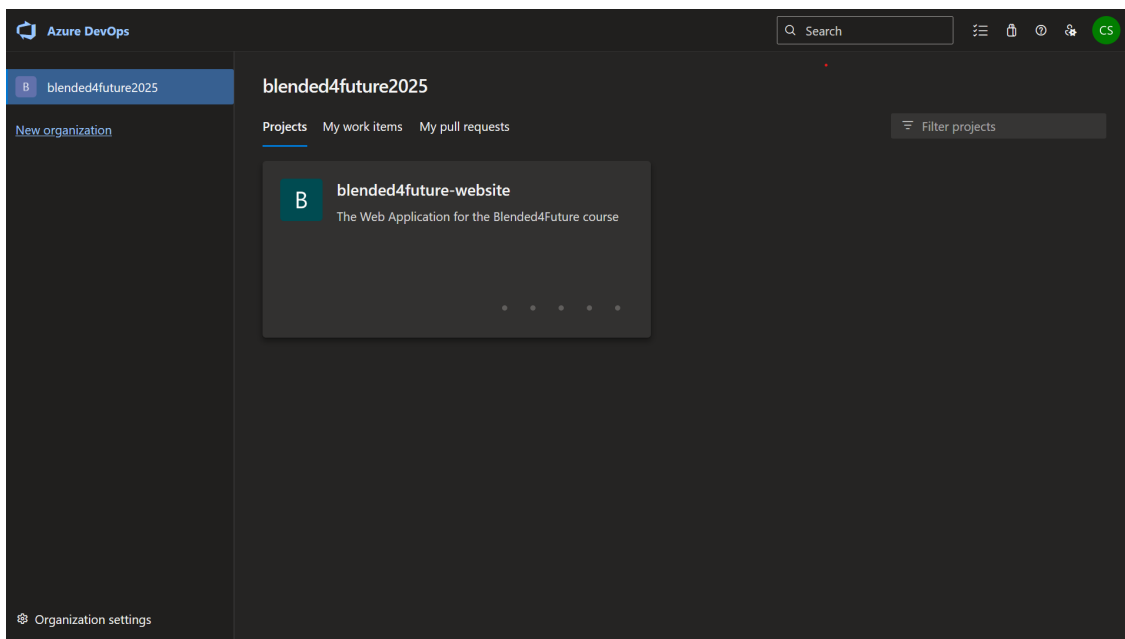
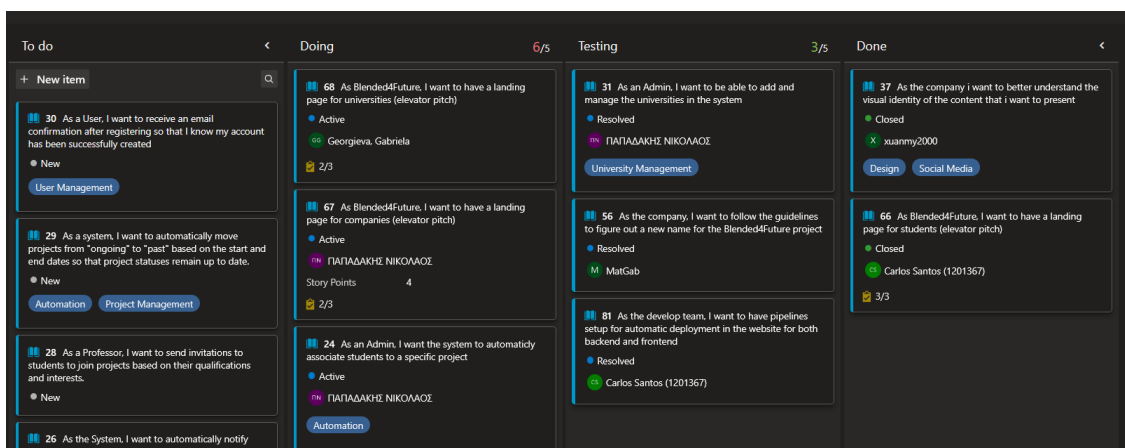


Figura 1.1: Homepage do projeto

Figura 1.2: Exemplificação do uso das *boards* no DevOps

desenvolvimento de software. A sua interface intuitiva, combinada com um elevado grau de personalização dos painéis de trabalho, automações eficazes e relatórios detalhados, facilita a gestão ágil e a integração com o ecossistema Microsoft, bem como com diversas outras ferramentas DevOps.

Outras possíveis e válidas escolhas para este tipo de ferramenta seriam, por exemplo:

**Github Projects** O GitHub Projects distingue-se pela sua simplicidade e integração direta no ambiente de desenvolvimento, permitindo um acompanhamento natural e ágil das tarefas dentro dos próprios repositórios de código. Esta característica potencia a colaboração entre programadores e torna a ferramenta particularmente útil para projetos de pequena escala ou equipas que privilegiam a rapidez e a facilidade de uso. No entanto, apresenta limitações em termos de funcionalidades de gestão ágil e geração de relatórios detalhados, pelo que pode

não ser adequado para projetos com requisitos organizacionais mais rigorosos ou estruturas de equipa complexas.

**Jira** O Jira apresenta-se como uma ferramenta de gestão de projetos altamente robusta, especialmente orientada para equipas que adotam metodologias ágeis, como Scrum e Kanban. Destaca-se pela capacidade de personalização detalhada dos fluxos de trabalho, pela integração aprofundada com ferramentas DevOps e pela produção de relatórios analíticos complexos, que facilitam o acompanhamento rigoroso do progresso em projetos de elevada complexidade. Contudo, a sua curva de aprendizagem é acentuada e a interface pode revelar-se excessivamente densa para utilizadores menos experientes, o que implica um investimento significativo em configuração e formação inicial, podendo limitar a sua eficiência em equipas pequenas ou com

### Tecnologias para desenvolvimento

Atendendo à heterogeneidade da equipa, composta por indivíduos com diferentes experiências e origens académicas, tornou-se imperativo definir consensualmente as tecnologias a utilizar no projeto. Para tal, procedeu-se ao levantamento sistemático das ferramentas previamente utilizadas pelos elementos da equipa, tendo-se optado, em média, pelas tecnologias mais frequentemente referidas e consideradas mais adequadas à criação de interfaces de utilizador e à implementação de APIs do tipo REST.

Tecnologia	Nº de referências
Java	5
Python	4
CSS	4
MySQL	4
HTML	5
SQL Server	3
C	2
JavaScript	5
TypeScript	2
C++	2
Bash Scripting	1
R	1
ReactTS	1
C#	1
.NET (ASP)	1
Entity Framework	1
H2	1
MongoDB	1
PHP	1

Tabela 1.2: Levantamento das tecnologia que os elementos da equipa já utilizaram

Houve uma clara preferência quanto ao uso de linguagens com tipagem forte. Estas permitem o melhor rastreamento de erros e garantem o formato dos objetos no programa. Esta preferência teve como base principal a inexperiência de vários elementos da equipa.

A tabela ?? demonstra este levantamento que permitiu à equipa fundamentar a decisão sobre as tecnologias a adoptar:

**Java** Linguagem de programação orientada a objetos amplamente utilizada no desenvolvimento de aplicações empresariais e sistemas robustos. A sua integração com o ecossistema Spring possibilita a criação eficiente de APIs REST, suportando a escalabilidade e manutenção do software. A sua tipagem forte permite que erros sejam mais fáceis de evitar, principalmente para desenvolvedores com pouca experiência.

**Spring** Framework para a plataforma Java caracterizada pela sua modularidade e flexibilidade, que simplifica o desenvolvimento de aplicações empresariais. Entre as suas principais vantagens técnicas destaca-se a Inversão de Controlo e a Injeção de Dependência, que promovem baixo acoplamento e facilitam a manutenção e testabilidade do código. Suporta também Programação Orientada a Aspectos, permitindo isolar funcionalidades transversais como segurança e logging. Entre as suas várias funcionalidades denota-se o Spring Data, que abstrai o acesso a repositórios e bases de dados, facilitando a interação com diversos sistemas de armazenamento através de interfaces padronizadas; assim, os repositórios do Spring permitem a execução simplificada de operações CRUD e consultas complexas sem necessidade de escrever código SQL.

**Junit** Framework amplamente utilizada no desenvolvimento Java para a criação de testes unitários, essencial para garantir a qualidade e fiabilidade do código ao permitir a verificação automática e repetível do comportamento de unidades individuais dentro de uma aplicação. Este framework proporciona uma estrutura simples e eficiente para escrever, organizar e executar testes, suportando anotações que facilitam a definição de métodos de teste, fases de inicialização e limpeza, bem como a gestão de exceções esperadas.

**TypeScript** Linguagem de programação baseada em JavaScript, que integra tipagem estática e recursos avançados, facilitando o desenvolvimento de interfaces de utilizador complexas e a manutenção de grandes bases de código. A sua adoção é relevante na construção de aplicações web modernas e escaláveis.

**React TS** Framework de desenvolvimento frontend baseada em componentes, que alia a flexibilidade do React à segurança oferecida pela tipagem forte do TypeScript. Esta combinação permite construir interfaces de utilizador dinâmicas e com maior robustez, suportando os modernos paradigmas de desenvolvimento web com principal foco na reusabilidade dos referidos componentes.

**TailwindCSS** Framework utilitária que simplifica a estilização ao permitir aplicar classes diretamente nos componentes de React, eliminando a necessidade de escrever CSS tradicional. A sua integração assegura uma personalização rápida e responsiva, facilitada pela configuração flexível e pelo suporte a interfaces modulares e tipadas com TypeScript, promovendo assim maior produtividade e manutenção eficiente do código em projetos modernos de frontend.

**MySQL** : Sistema de gestão de bases de dados relacionais (SGBDR) amplamente utilizado, reconhecido pela sua natureza *open source* e pelo desempenho eficiente em operações de leitura e escrita, particularmente em aplicações web. Entre as suas vantagens técnicas

destacam-se a facilidade de uso, a escalabilidade para lidar com grandes volumes de dados e o suporte a transações ACID. Adicionalmente, o MySQL oferece mecanismos de segurança sólidos, como autenticação, autorização e criptografia, reforçando a proteção dos dados armazenados.

## **1.5 Desafios enfrentados**

Esta seção será desenvolvida na perspectiva do autor do relatório, centrando-se na experiência de trabalho em equipa, nos principais desafios enfrentados e nas ferramentas e metodologias aplicadas na sua resolução. Ao longo do projeto, verificou-se uma significativa falta de comunicação entre os membros, o que dificultou a coordenação das atividades e conduziu, em diversas ocasiões, ao incumprimento das tarefas previamente atribuídas. Para além disso, a ausência de contributos consistentes por parte de alguns elementos da equipa originou constrangimentos adicionais, que se refletiram na execução e continuidade de determinadas etapas do desenvolvimento. Apesar da adoção de medidas de mitigação, como a redefinição de responsabilidades, a realização de reuniões adicionais e a aplicação de metodologias colaborativas, persistiram limitações que condicionaram a concretização plena de certos objetivos estabelecidos. Esta reflexão procura, assim, não apenas evidenciar os obstáculos encontrados, mas também destacar as aprendizagens resultantes da experiência, nomeadamente no que respeita à gestão de equipa, à adaptação em contextos de incerteza e à importância de uma comunicação eficaz para o sucesso coletivo.

### **1.5.1 Expectativa criada**

### **1.5.2 Problemas**

**Falta de comunicação**

**Incumprimento de tarefas**

**Desconhecimento sobre as ferramentas**

### **1.5.3 Tentativas de solução**

**Reuniões de equipa**

**Comunicação com os professores envolvidos**

**Redução da scope do projeto**

## **Capítulo 2**

# **Estado da Arte**



## Capítulo 3

# Analise do problema e desenho de uma solução

*Adicionar texto introdutorio*

### 3.1 O Dominio

Este projeto tem como objetivo a criação de uma plataforma adaptada à organização e aos seus desafios. Para tal, foi necessário esquematizar a respetiva lógica de negócio.

Após discussões

### 3.2 Engenharia de requisitos

Durantes os primeiros dois sprints foi estabelecido um *backlog* de *user stories* que refletia as funcionalidades necessárias

#### 3.2.1 User Flow Chart

#### 3.2.2 Requisitos funcionais

Casos de uso

#### 3.2.3 Requisitos não Funcionais

incluir tambem a necessidade de CI/CD Pipelines

Figura 3.1: Diagrama de classes da lógica de negócio do Blended4Future 1

## **3.3 Arquitetura do sistema**

### **3.3.1 Desenho do sistema**

### **3.3.2 Nivel 1**

### **3.3.3 Nivel 2**

### **3.3.4 Nivel 3**

### **3.3.5 Nivel 4 - Código**

### **3.3.6 Padrões utilizados**

## **Capítulo 4**

# **Implementação de uma solução**

### **4.1 A Implementação**

#### **4.1.1**

#### **4.1.2 Frontend**

#### **4.1.3 Deployments**

### **4.2 Testes**

#### **4.2.1 Testes Unitários**

#### **4.2.2 Testes de Implementação**

### **4.3 Avaliação da solução**

## **Capítulo 5**

# **Reflexão sobre trabalho em equipa**