

Licenciatura em Engenharia Informática

Escola Superior de Tecnologia e Gestão

Instituto Politécnico de Viana do Castelo

Unidade curricular de

Engenharia de Software II

2024/2025

Alunos: Ana Oliveira - 24673 - aluisaoliveira@ipvc.pt

Filipe Ferreira - 26254 - f.ferreira@ipvc.pt

Gabriela Ribeiro - 24600 - gabrielar@ipvc.pt

João Cunha - 25033 - cunha.joao@ipvc.pt

Pedro Alves -24561 - phenriquealves@ipvc.pt

Docentes: António Cruz - miguel.cruz@estg.ipvc.pt

Abel Dantas

1. Descrição do tema	2
2. Objetivos do projeto	2
3. Metodologia Ágil	3
4. Análise e Modelação: Requisitos, Casos de Uso e Modelo de Classes do Domínio	4
5. Modelo da base de dados	7
6. Evolução das Funcionalidades por Sprint	8
6.1. Semana inicial : 24/02 - 07/03	8
6.1.1. Objetivos	8
6.1.2. Funcionalidades Implementadas	9
6.1.3. Resultados da Review	9
6.1.4. Planning - Sprint (07/03-21/03)	10
7. Referências	13

1. Descrição do tema

O projeto “**Plataforma de Gestão de Talentos IT**” tem como finalidade criar uma solução para ajudar empresas a recrutar profissionais qualificados da área de Tecnologias de Informação (TI) e permitir que os profissionais mostrem suas habilidades e experiências. A plataforma vai facilitar o encontro entre empresas e candidatos, através da criação de perfis onde os profissionais podem listar suas competências, experiência e certificações.

As empresas poderão procurar talentos com base em critérios como habilidades específicas e localização. Além disso, a plataforma permitirá que os recrutadores publiquem ofertas de trabalho e que os candidatos visualizem e se candidatem a essas oportunidades.

A plataforma será desenvolvida utilizando tecnologias atuais, assegurando um desempenho otimizado e uma interface intuitiva para os usuários. A gestão do projeto seguirá a metodologia ágil, garantindo entregas contínuas e alinhamento constante com os requisitos e expectativas do cliente.

2. Objetivos do projeto

Este projeto, desenvolvido no âmbito da disciplina de Engenharia de Software II, **tem como objetivo** a construção de uma plataforma de gestão de currículos para profissionais da área de Tecnologias de Informação (TI). O desenvolvimento será realizado utilizando a **metodologia ágil Scrum**, com foco em uma implementação web moderna, eficiente e escalável.

A plataforma a ser desenvolvida permitirá aos utilizadores criar e gerir perfis profissionais, associando habilidades (skills) com experiência, bem como registar propostas de trabalho para clientes.

As principais funcionalidades incluem a criação de perfis com detalhes como nome, país, e-mail, preço por hora, e a possibilidade de adicionar e editar skills, experiências profissionais e detalhes sobre projetos anteriores.

A plataforma também permitirá a pesquisa de talentos com base em skills específicas e a criação de propostas de trabalho com requisitos de habilidades e experiência.

A **arquitetura do sistema** será baseada nas tecnologias e padrões recomendados, nomeadamente:

- **Backend:** A API será desenvolvida utilizando .NET Core (versões 8 ou 9), com a implementação de Web API e Entity Framework para a interação com a base de dados.
- **Frontend:** A interface será construída com Blazor ou Razor, utilizando componentes web interativos para a gestão de perfis e propostas de trabalho.

- **Base de dados:** A base de dados será implementada em PostgreSQL, garantindo a persistência de dados e o suporte a operações complexas de pesquisa e filtragem de perfis.
- **Arquitetura e Design:** O sistema será estruturado utilizando o padrão de arquitetura MVC (Model-View-Controller), juntamente com o padrão de repositório para a gestão de acesso à base de dados e o uso de design patterns para garantir uma arquitetura limpa e escalável.

Além das funcionalidades principais, será implementada uma camada de autenticação, com três níveis de permissões: User, UserManager e Admin, controlando o acesso às diversas operações do sistema, como a criação de utilizadores e a gestão de permissões.

A plataforma também deverá incluir partes analíticas, como a geração de preços médios mensais por categoria de talento, país e skill, e a possibilidade de filtrar talentos com base nas propostas de trabalho em aberto. O sistema permitirá também que os utilizadores gerem relatórios detalhados sobre o estado atual dos perfis, incluindo a associação de skills e experiências, bem como o histórico de propostas enviadas e aceites.

Este projeto será desenvolvido em sprints, com entregas periódicas contendo o código fonte, documentação, testes unitários e vídeos explicativos. O sucesso do projeto será avaliado com base na funcionalidade, usabilidade e conformidade com os requisitos estabelecidos, além da aplicação de boas práticas de desenvolvimento de software, como a utilização de testes automatizados e a implementação de padrões de design e arquitetura.

3. Metodologia Ágil

A metodologia ágil é um conjunto de práticas que busca entregar valor de forma contínua e incremental, priorizando a colaboração, flexibilidade e adaptação às mudanças ao longo do desenvolvimento de software. Para o nosso projeto, foi adotada uma abordagem ágil focada em entregas rápidas e contínuas. As sprints terão uma duração de **duas semanas**, permitindo ciclos curtos de desenvolvimento e ajustando constantemente o produto às necessidades e feedbacks do cliente.

A equipa segue os **princípios do Scrum**, com os seguintes papéis:

- **Product Owner (PO):** Responsável por priorizar as funcionalidades e garantir que o backlog do produto reflita as necessidades do cliente, garantindo que as entregas atendam às expectativas.
- **Scrum Master (SM):** Facilita o processo, removendo obstáculos e assegurando que a equipa siga as práticas do Scrum, promovendo a comunicação eficiente e a colaboração.

- **Desenvolvedores (Devs):** Responsáveis pela implementação do código e pela entrega das funcionalidades de acordo com o backlog do produto, garantindo qualidade e segurança no desenvolvimento.

Adicionalmente, a nossa equipa realiza **daily stand ups** através de videochamada no Discord, promovendo a comunicação e alinhamento entre os membros. Além disso, ao final de cada sprint, fazemos **Sprint Review** e **Sprint Retrospective**, permitindo avaliar o progresso, identificar melhorias e planejar as ações para a próxima sprint.

O acompanhamento do desenvolvimento do projeto é feito através do [Trello](#), onde todos os itens do backlog são organizados e acompanhados, permitindo que a equipa visualize facilmente o progresso de cada tarefa e se mantenha alinhada com os objetivos do projeto.

Esta abordagem garante que o projeto esteja sempre em constante evolução, com entregas incrementais e feedback contínuo, permitindo um desenvolvimento ágil e alinhado com as necessidades do cliente.

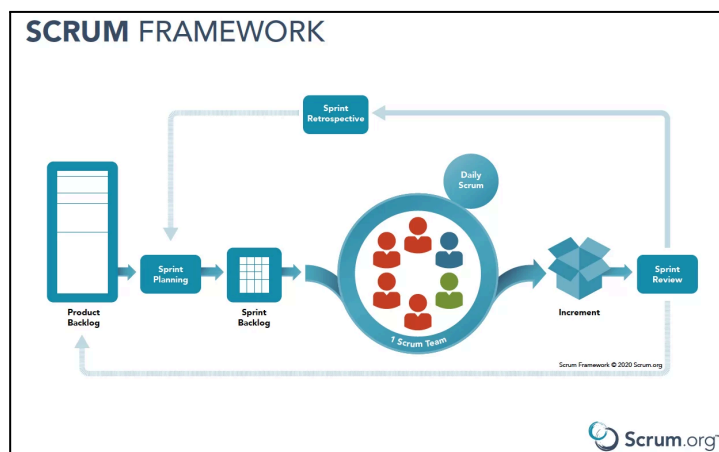


Imagem I - SCRUM Framework

4. Análise e Modelação: Requisitos, Casos de Uso e Modelo de Classes do Domínio

A análise e modelação de um sistema é uma etapa importante no desenvolvimento de software, pois permite compreender profundamente os requisitos do projeto e estruturar a solução de forma eficiente. Nesta fase, o foco encontra-se em identificar as necessidades do cliente e os processos de negócio a serem automatizados ou otimizados pela plataforma. A partir dessa compreensão, elaboram-se os casos de uso, que descrevem as interações entre os utilizadores e o sistema, e o modelo de classes do domínio, que define as entidades e os relacionamentos fundamentais para o funcionamento da aplicação. Estes artefactos servem como a base para a construção do sistema, garantindo que ele atenda de forma precisa às necessidades e expectativas dos utilizadores.

1. Levantamento de requisitos:

Login e autenticação:

- Como utilizador, eu quero fazer login na plataforma para aceder ao meu painel de controlo e aos meus dados pessoais.
- Como utilizador, eu quero poder recuperar a minha palavra-passe, caso a tenha esquecido, para poder aceder à plataforma novamente.

Gestão de Perfil

- Como utilizador, eu quero editar as minhas informações pessoais (nome, email, experiência) para manter o meu perfil atualizado.
- Como utilizador, eu quero adicionar as minhas competências técnicas e certificações ao meu perfil para que os recrutadores possam encontrar o meu perfil com base nas minhas habilidades.

Pesquisa de Talentos

- Como recrutador, eu quero pesquisar candidatos com base nas suas competências, experiência e localização para encontrar o talento ideal para a vaga.
- Como recrutador, eu quero filtrar candidatos por nível de experiência e habilidades específicas para refinar a minha pesquisa.

Gestão de Oportunidades de Trabalho

- Como recrutador, eu quero criar e publicar novas oportunidades de emprego para atrair candidatos qualificados.
- Como recrutador, eu quero editar ou remover oportunidades de emprego existentes, caso os requisitos ou a vaga mudem.

Candidaturas a Oportunidades

- Como utilizador, eu quero poder candidatar-me a oportunidades de emprego para me candidatar às vagas que me interessam.
- Como utilizador, eu quero acompanhar o status das minhas candidaturas (em andamento, aceito, rejeitado) para saber onde estou no processo.

Notificações

- Como utilizador, eu quero receber notificações sobre novas oportunidades que correspondem às minhas competências, para estar sempre atualizado sobre novas vagas.

- Como recrutador, eu quero ser notificado sobre novas candidaturas para poder avaliar os candidatos rapidamente.

Dashboard de Performance

- Como recrutador, eu quero ver um dashboard que me mostre estatísticas de visualizações de vagas e candidaturas, para avaliar o desempenho das minhas publicações.
- Como utilizador, eu quero um painel que me mostre as vagas às quais me candidatei, o status de cada candidatura e outras informações úteis, como feedback de recrutadores.

Sistema de Feedback

- Como recrutador, eu quero dar feedback aos candidatos sobre as suas candidaturas, para ajudá-los a melhorar nas próximas oportunidades.
- Como utilizador, eu quero receber feedback construtivo dos recrutadores para melhorar a minha candidatura em futuras oportunidades.

2. Modelo de Casos de Uso:



Imagem II - Modelo de Casos de Uso

3. Modelo de Classes de Domínio:

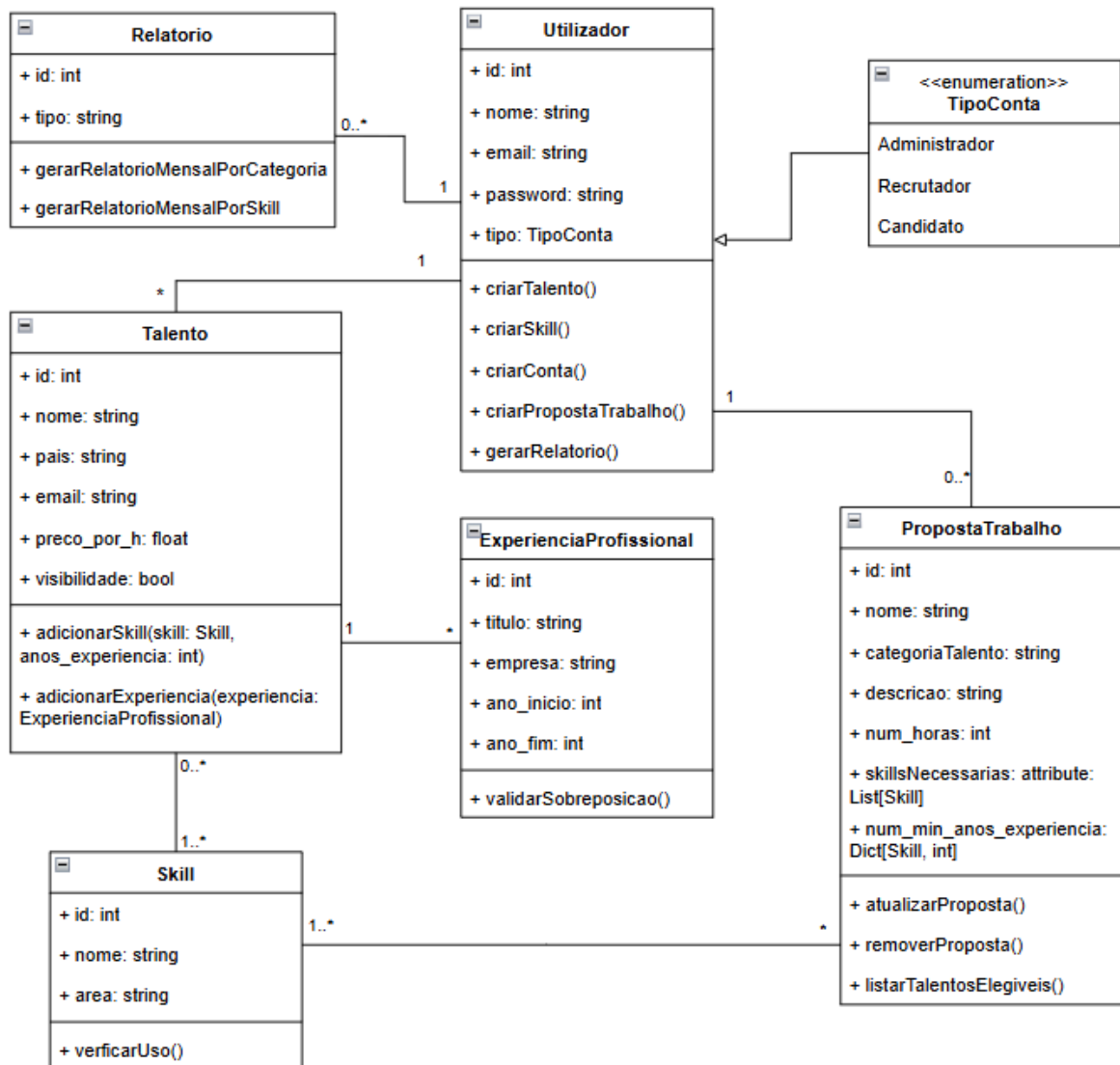


Imagem III - Modelo de Classes de Domínio

5. Modelo da base de dados

A base de dados descrita tem como objetivo apoiar uma **Plataforma de Gestão de Talentos IT**, e estrutura-se em várias entidades interligadas que representam as informações-chave sobre os utilizadores, talentos, competências, experiências, clientes e propostas de emprego.

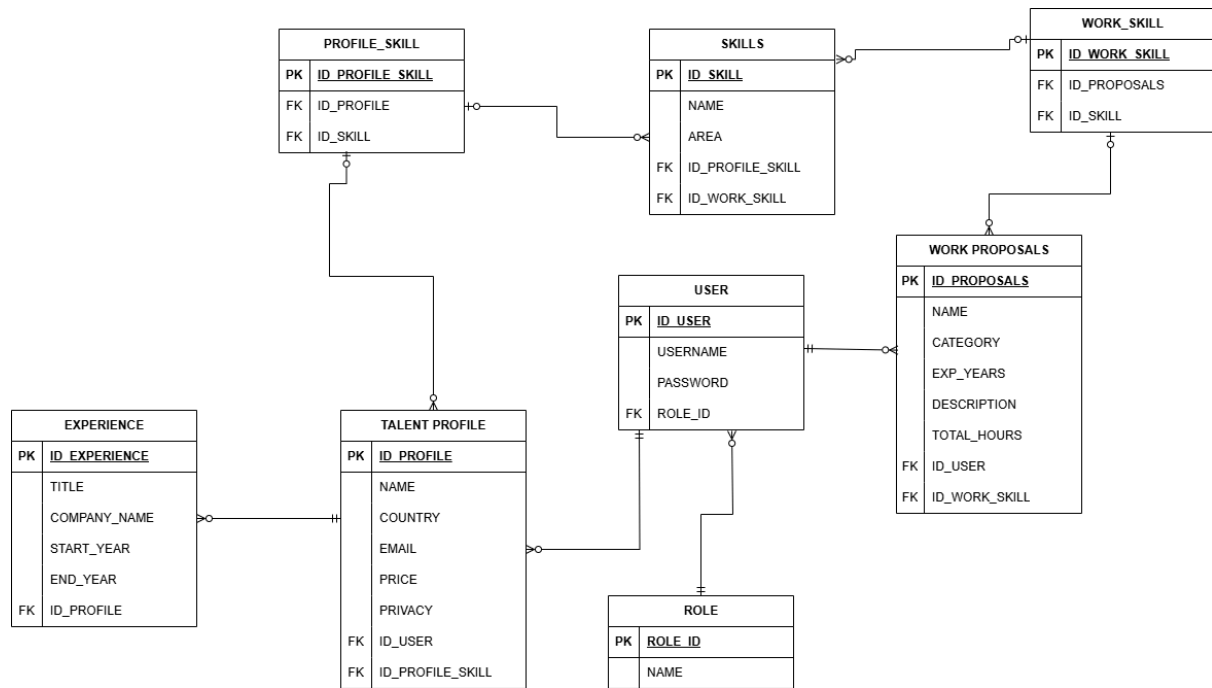


Imagem IV - Modelo de Base de Dados

6. Evolução das Funcionalidades por Sprint

6.1. Semana inicial : 24/02 - 07/03

6.1.1. Objetivos

Os objetivos para esta primeira semana foram : fazer o setup das ferramentas necessárias e do projeto em si para existir um ponto de partida em comum e finalizar o modelo inicial da base de dados. Também tivemos outras tarefas como uma investigação relacionado a migrações usando o Entity Framework Core.

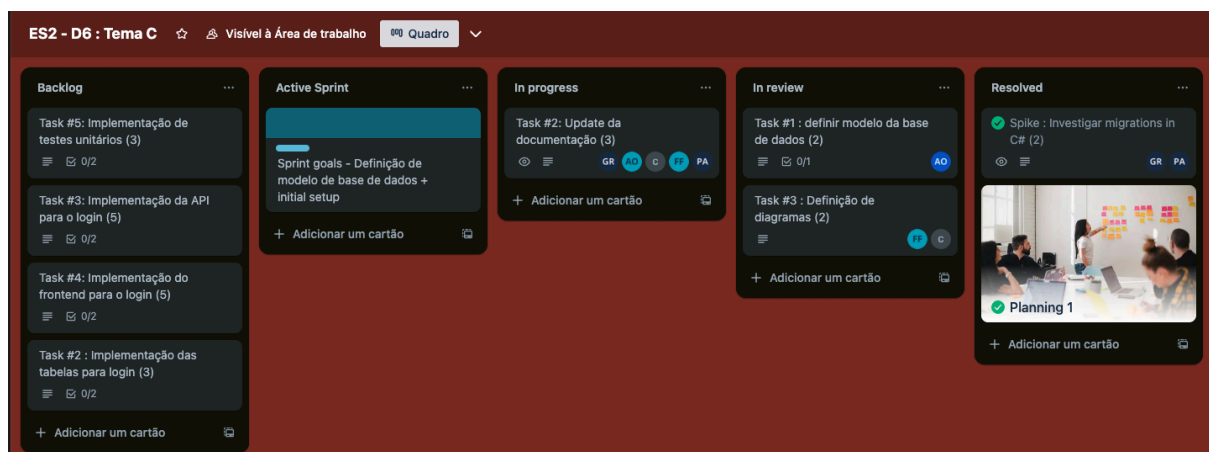


Imagem V - Sprint da Semana Inicial

6.1.2. Funcionalidades Implementadas

O modelo inicial da base de dados foi definido e através da tarefa de investigação pode-se obter algumas conclusões tais como:

Uma DB Migration (migração de banco de dados) em C# permite gerir mudanças no esquema do banco de dados de forma controlada e versionada. O Entity Framework Core (EF Core), um ORM (Object-Relational Mapper) popular em .NET, é frequentemente usado para esse propósito.

O que é uma Migração?

Uma migração é um conjunto de instruções que modificam a estrutura do banco de dados, incluindo:

- Criação/remoção de tabelas
- Adição/remoção de colunas
- Alteração de tipos de dados
- Criação de chaves estrangeiras e índices

Como funciona?

O Entity Framework Core permite definir o modelo de dados em código C# e gerar automaticamente scripts SQL para sincronizar o banco de dados.

Benefícios das Migrations

Controla a evolução do banco de dados ao longo do tempo

Permite versionamento e rollback de alterações

Evita a necessidade de escrever SQL manualmente

Com isso, as migrações ajudam a manter o banco de dados alinhado ao código da aplicação de forma eficiente.

6.1.3. Resultados da Review

Houve ajustes relativamente às tarefas inicialmente planeadas no backlog pois ainda não havia muita noção relativamente às capacidades da equipa e à complexidade das tarefas o que levou a uma mudança de tarefas neste intervalo de tempo, mesmo assim, a equipa mostrou uma comunicação efetiva e proativa, o que ajuda a colmatar o facto de a maior parte dos elementos deste grupo serem trabalhadores-estudantes e não terem uma disponibilidade completa.

Apesar de não termos sempre o tempo necessário, conseguimos combater essa falha com uma gestão eficiente das nossas tarefas, aproveitando ao máximo os períodos disponíveis e mantendo um foco constante nos objetivos, garantindo assim o cumprimento das nossas responsabilidades e prazos.

6.1.4. Planning - Sprint (07/03-21/03)

Na meeting do sprint planning foram atribuídas as tarefas aos membros da equipa e definido o objetivo principal para esta sprint que consiste em implementar o login dos utilizadores. O modelo utilizado para medir o esforço associado a cada tarefa foi o Practical Fibonacci.

Practical Fibonacci	
0	No effort is required , or there is some effort required, but there is no business value delivered, so no Points are accumulated for doing the work. An example is a desired behavioral change deriving from the Scrum Retrospective.
1	Extra small. Developers feel they understand most requirements and consider it relatively easy, probably the smallest item in the Sprint and mostly likely completed in one day.
2	Small. A little bit of thought, effort, or problem-solving is required, but the Developers have done this a lot, so they have confidence in the requirements. Or, it sounds extra small, but they want to hedge their bet just a bit.
3	Average. Developers have done this a lot; they know what needs to be done. There may be a few extra steps, but that's it. It's doubtful that they will need to research anything.
5	Large. This is complex work, or the Developers don't do this very often. Most Developers will need assistance from someone else on the team. This is probably one of the largest items that can be completed within a Sprint.
8	Extra Large. This is going to take some time and research and probably more than one developer to complete within two weeks. In addition, Developers need to make several assumptions that increase the risk and could affect getting it Done.
13	Warning! This is a complex piece of work with a lot of unknowns and requires multiple assumptions to size. It is too much to complete in one Sprint. Instead, split this into multiple items that can be completed independently.
21	Hazard! A "21" or "34" reflects too much complexity to be done within one Sprint. It will need to be refined more. The large size also indicates more risk, assumptions, and dependencies involved to complete this item.
?	Danger! As a Developer, we don't want to do this work the way it is currently written. It is very complex and cannot be completed in the timeframe of an iteration or Sprint. Perhaps the requirements are so fuzzy that it's rife with danger.

Imagem VI - Practical Fibonacci model

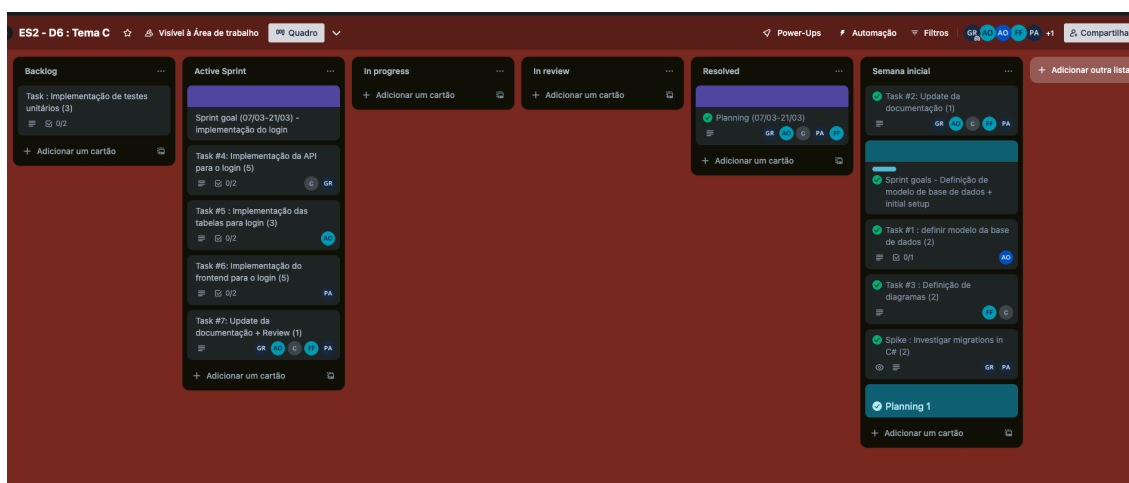


Imagem VII - Sprint Planning

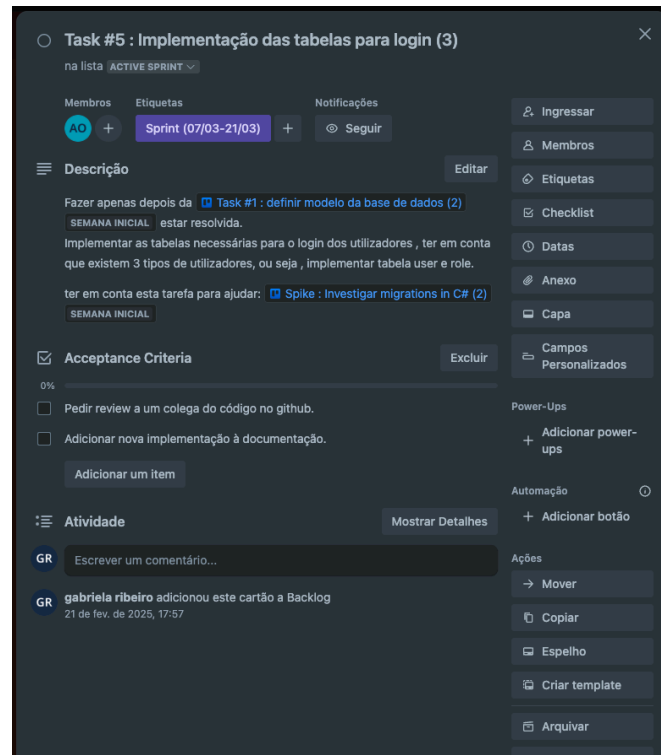


Imagem VIII - Task de implementação de tabelas para o login

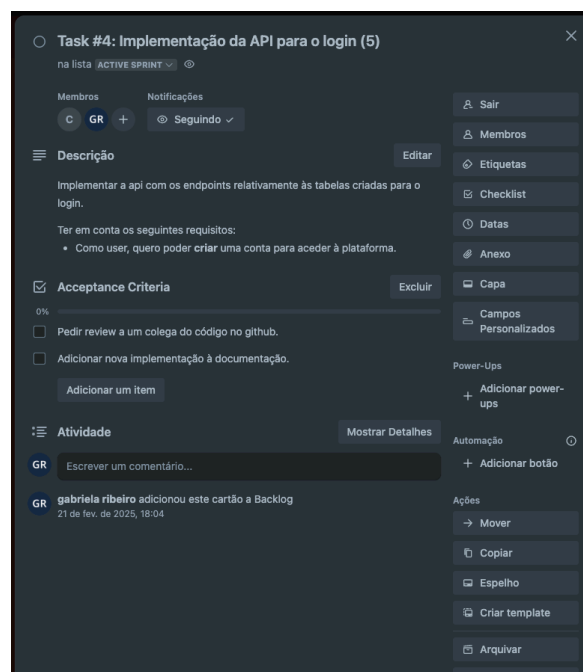


Imagem IX - Task de implementação da api para o login

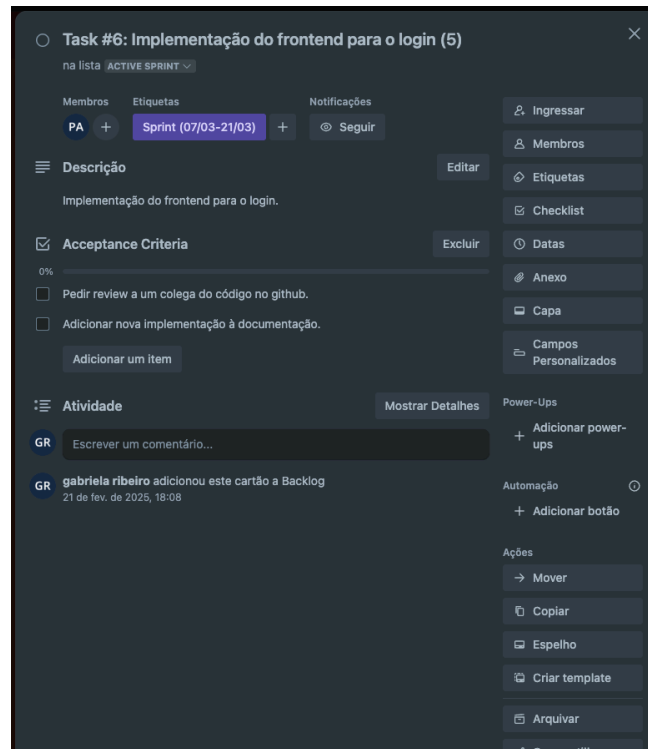


Imagem X - Task de implementação do frontend para o login

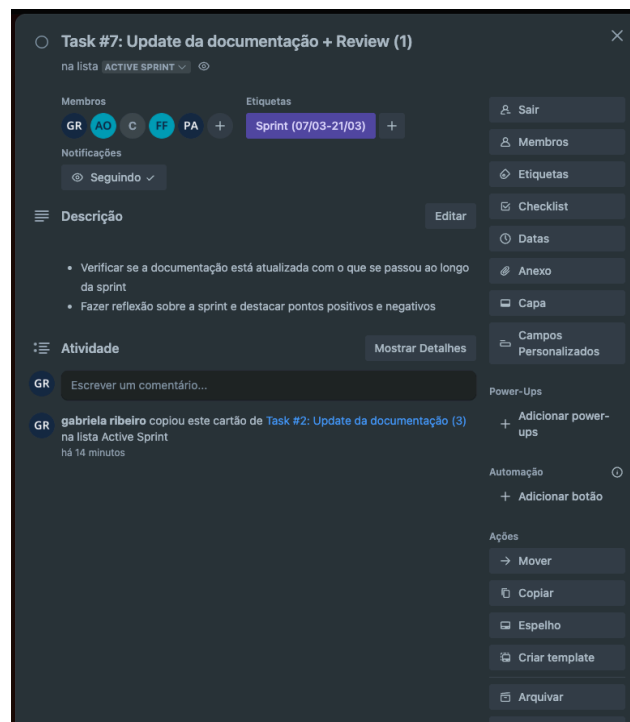


Imagem XI - Task de implementação do frontend para o login

7. Referências

Scrum Framework : <https://www.scrum.org/>

Migrations in C# : <https://www.youtube.com/watch?v=kMewc-TjO2s>

Trello : <https://trello.com/>

Discord : <https://discord.com/>