|  |  |  |  |
| --- | --- | --- | --- |
| ISTEC - Instituto Superior de Tecnologias Avançadas | Lisbon | | Curso de CTeSP em  Desenvolvimento para Dispositivos Móveis  Laboratório de Desenvolvimento II (LabDev II) | |
|  |  | |  |
| **Ano Letivo 2025/2026** |  | | **2º Ano, 1º Semestre** |
|  | | | |
| **Projeto de LabDev II** | | | |

Diagram

Description automatically generated

**Relatório de Acompanhamento do**

**Projeto**

|  |  |
| --- | --- |
|  |  |
| **Grupo: N10** | **Docente:** Paulo Meira |
| **Nº** 2024541 | Guilherme Hutchinson |
| **Nº** 2024523 | Martim Pinto |

[Índice de Figuras 5](#_Toc216330536)

[Índice de Tabelas 6](#_Toc216330537)

[1 Introdução 8](#_Toc216330538)

[1.1 Sumário executivo 8](#_Toc216330539)

[2 Especificação do Sistema 9](#_Toc216330540)

[2.1 Definição da Lógica de Negócio 9](#_Toc216330541)

[2.2 Análise de Impacto 9](#_Toc216330542)

[2.3 Análise Concorrencial 10](#_Toc216330543)

[2.3.1 Trello 10](#_Toc216330544)

[2.3.2 Asana 11](#_Toc216330545)

[2.3.3 Jira 11](#_Toc216330546)

[2.3.4 Comparação dos Sistemas 12](#_Toc216330547)

[2.3.5 Enquadramento da análise concorrencial no SI 13](#_Toc216330548)

[2.4 Wireframes/Mockups 14](#_Toc216330549)

[3 Scrum 15](#_Toc216330550)

[3.1 Aplicação do Scrum ao Projeto 15](#_Toc216330551)

[**Contextualização da metodologia ágil** 15](#_Toc216330552)

[**Forma como foi aplicado o Scrum e principais adaptações** 15](#_Toc216330553)

[**Formato das reuniões** 15](#_Toc216330554)

[**Eventos Scrum seguidos** 15](#_Toc216330555)

[**Detalhes das Sprints** 16](#_Toc216330556)

[**Ferramentas utilizadas** 16](#_Toc216330557)

[3.2 Stakeholders e Scrum Team 16](#_Toc216330558)

[3.3 Product backlog 16](#_Toc216330559)

[3.4 Execução do projeto 16](#_Toc216330560)

[3.4.1 Sprint 1 (16/10/2025 a 30/10/2025) 17](#_Toc216330561)

[3.4.2 Sprint 2 (30/10/2025 a 13/11/2025) 18](#_Toc216330562)

[3.4.3 Sprint 3 (Dia de Mês de 2025 a Dia de Mês de 2025) 20](#_Toc216330563)

[3.4.4 Sprint 4 (Dia de Mês de 2025 a Dia de Mês de 2025) 21](#_Toc216330564)

[3.5 *Retrospective Summary* do Projeto 22](#_Toc216330565)

[4 Conclusão 24](#_Toc216330566)

**ÍNDICE**

# Índice de Figuras



[Figura 1 – Wireframe/Mockup do ecrã principal (no exemplo: esq. Wireframe; dir. Mockup) 10](#_heading=h.2ukpk54s6d2o)

# Índice de Tabelas

[Índice de Figuras 5](#_Toc216339639)

[Índice de Tabelas 6](#_Toc216339640)

[1 Introdução 8](#_Toc216339641)

[1.1 Sumário executivo 8](#_Toc216339642)

[2 Especificação do Sistema 9](#_Toc216339643)

[2.1 Definição da Lógica de Negócio 9](#_Toc216339644)

[2.2 Análise de Impacto 9](#_Toc216339645)

[2.3 Análise Concorrencial 10](#_Toc216339646)

[2.3.1 Trello 10](#_Toc216339647)

[2.3.2 Asana 11](#_Toc216339648)

[2.3.3 Jira 11](#_Toc216339649)

[2.3.4 Comparação dos Sistemas 12](#_Toc216339650)

[2.3.5 Enquadramento da análise concorrencial no SI 13](#_Toc216339651)

[2.4 Wireframes/Mockups 14](#_Toc216339652)

[3 Scrum 15](#_Toc216339653)

[3.1 Aplicação do Scrum ao Projeto 15](#_Toc216339654)

[**Contextualização da metodologia ágil** 15](#_Toc216339655)

[**Forma como foi aplicado o Scrum e principais adaptações** 15](#_Toc216339656)

[**Formato das reuniões** 15](#_Toc216339657)

[**Eventos Scrum seguidos** 15](#_Toc216339658)

[**Detalhes das Sprints** 16](#_Toc216339659)

[**Ferramentas utilizadas** 16](#_Toc216339660)

[3.2 Stakeholders e Scrum Team 16](#_Toc216339661)

[3.3 Product backlog 16](#_Toc216339662)

[3.4 Execução do projeto 16](#_Toc216339663)

[3.4.1 Sprint 1 (16/10/2025 a 30/10/2025) 17](#_Toc216339664)

[3.4.1.1 Sprint Planning 17](#_Toc216339665)

[3.4.1.2 Daily Meetings (1 por semana) 17](#_Toc216339666)

[3.4.1.3 Sprint Retrospective 18](#_Toc216339667)

[3.4.2 Sprint 2 (30/10/2025 a 13/11/2025) 18](#_Toc216339668)

[3.4.2.1 Sprint Planning 18](#_Toc216339669)

[3.4.2.2 Daily Meetings (1 por semana) 18](#_Toc216339670)

[3.4.2.3 Sprint Retrospective 19](#_Toc216339671)

[3.4.3 Sprint 3 (13/11/2025 a 27/11/2025) 20](#_Toc216339672)

[3.4.3.1 Sprint Planning 20](#_Toc216339673)

[3.4.3.2 Daily Meetings (1 por semana) 20](#_Toc216339674)

[3.4.3.3 Sprint Retrospective 21](#_Toc216339675)

[3.4.4 Sprint 4 (27/11/2025 a 11/12/2025) 21](#_Toc216339676)

[3.4.4.1 Sprint Planning 21](#_Toc216339677)

[3.4.4.2 Daily Meetings (1 por semana) 21](#_Toc216339678)

[3.4.4.3 Sprint Retrospective 22](#_Toc216339679)

[3.5 *Retrospective Summary* do Projeto 23](#_Toc216339680)

[4 Conclusão 25](#_Toc216339681)

[**Resumo do Estado Final do Software** 25](#_Toc216339682)

[**Conformidade com os Requisitos Definidos** 25](#_Toc216339683)

[**Estado do Backlog** 25](#_Toc216339684)

[**Riscos Identificados Durante o Desenvolvimento** 25](#_Toc216339685)

[**Medidas de Mitigação** 25](#_Toc216339686)

[**Problemas Ocorridos e Resolução** 25](#_Toc216339687)

[**Propostas para Continuidade ou Manutenção do Projeto** 26](#_Toc216339688)

[**Conclusões Gerais** 26](#_Toc216339689)

# Introdução

O relatório tem como objetivo principal descrever o nosso progresso ao longo de todos os sprints na criação de uma aplicação móvel para a gestão de tarefas, que tem por base o conceito Kanban. Esta aplicação pretende potencializar o acompanhamento e controlo das atividades de desenvolvimento de software, dando assim, uma maior organização, transparência e eficiência no fluxo de trabalho das equipas envolventes.

O projeto consiste na implementação de uma aplicação simples e de uma forma intuitiva, composta por três listas clássicas do modelo Kanban - **“**ToDo”, “Doing” e “Done” - que representam as tarefas em falta, em progresso, e as finalizadas. Cada programador pode visualizar, atualizar e gerir cada uma das suas tarefas, o que facilita a comunicação e o alinhamento entre os variados membros da equipa.

O plano geral do projeto está dividido em várias secções:

1. Análise de requisitos e definição das principais funcionalidades;
2. Mockup da interface e arquitetura da aplicação;
3. Desenvolvimento da aplicação e implementação dos componentes;
4. Testes funcionais e de qualidade;
5. Execução e avaliação final.

A equipa envolvida é constituída por programadores, um gestor de projeto e um designer de interface, o que assegura uma abordagem colaborativa e multidisciplinar ao longo de todo o processo de desenvolvimento.

Para garantir uma execução eficiente e iterativa, a metodologia ágil *Scrum* foi adotada. Esta metodologia caracteriza-se pela divisão do trabalho em *sprints* semanais, reuniões de acompanhamento regulares e entregas incrementais de valor. Esta abordagem permitiu ajustar o desenvolvimento com base no feedback dos utilizadores e assegurar que o produto final responde de forma eficaz às necessidades internas da organização.

Em suma, o presente relatório descreve todas as etapas do projeto, desde a preparação até à implementação da aplicação móvel e destaca as decisões técnicas e metodológicas que sustentaram todo o seu desenvolvimento.

## Sumário executivo

O projeto foi desenvolvido com o objetivo de criar uma solução funcional que respondesse aos requisitos definidos inicialmente. A equipa organizou o trabalho com a metodologia ágil Scrum, ajustando-a sempre que necessário para melhorar a comunicação e a produtividade. Apesar de algumas dificuldades iniciais, a equipa adaptou-se e concluiu o projeto com sucesso, garantindo um produto final consistente, testado e alinhado com os objetivos estabelecidos.

# Especificação do Sistema

## Definição da Lógica de Negócio

A aplicação que vamos desenvolver tem como objetivo gerir e acompanhar tarefas dadas por um gestor a um programador (fictício), através de um modelo Kanban.

* O sistema que vamos criar cria resposta às seguintes necessidades:
* Distribuição organizada de tarefas entre membros de uma equipa.
* Controlo do progresso das tarefas.
* Manter um registo detalhado de todo o processo.
* Fornece diversos formatos que permitam aos gestores avaliarem melhor o desempenho e a eficiência de uma equipa.

Temos como objetivo várias ações, tais como o login individual de cada utilizador, sendo gestor ou programador; garantir acesso apenas aos gestores a todas as funcionalidades de gestão; permitir aos programadores acederem às tarefas atribuídas e mudar os seus estados; manter um fluxo controlado de trabalho, limitando o número de tarefas em execução permitidas; e registar automaticamente as datas de criação, início e conclusão de cada tarefa.

## Análise de Impacto

**Análise de Impacto**

O desenvolvimento e a implementação do sistema vão revelar diversos impactos, tanto positivos como negativos, que importa considerar nos contextos da organização, social e de mercado.

**Impactos Positivos:**

* **Melhoria da eficiência interna:** O sistema automatiza tarefas manuais, reduz os erros nas equipas e aumenta a produtividade das mesmas.
* **Acesso a informação:** A disponibilização de todos os dados em tempo real permite decisões mais rápidas e fundamentadas o que faz reduzir os erros.
* **Melhoria da experiência para o utilizador:** Uma interface intuitiva e rápida faz com que a utilização seja mais simples e agradável a todos os seus utilizadores.
* **Vantagem competitiva:** A introdução do sistema pode reforçar a posição da empresa no mercado, distinguindo-a da concorrência e criar inovação nos métodos de trabalho.
* **Redução de custos empresariais:** Os processos ajudam a canalizar a diminuição do consumo de tempo desnecessário em outros métodos que se vai traduzir também em ganhos monetários para a empresa.

**Impactos Negativos:**

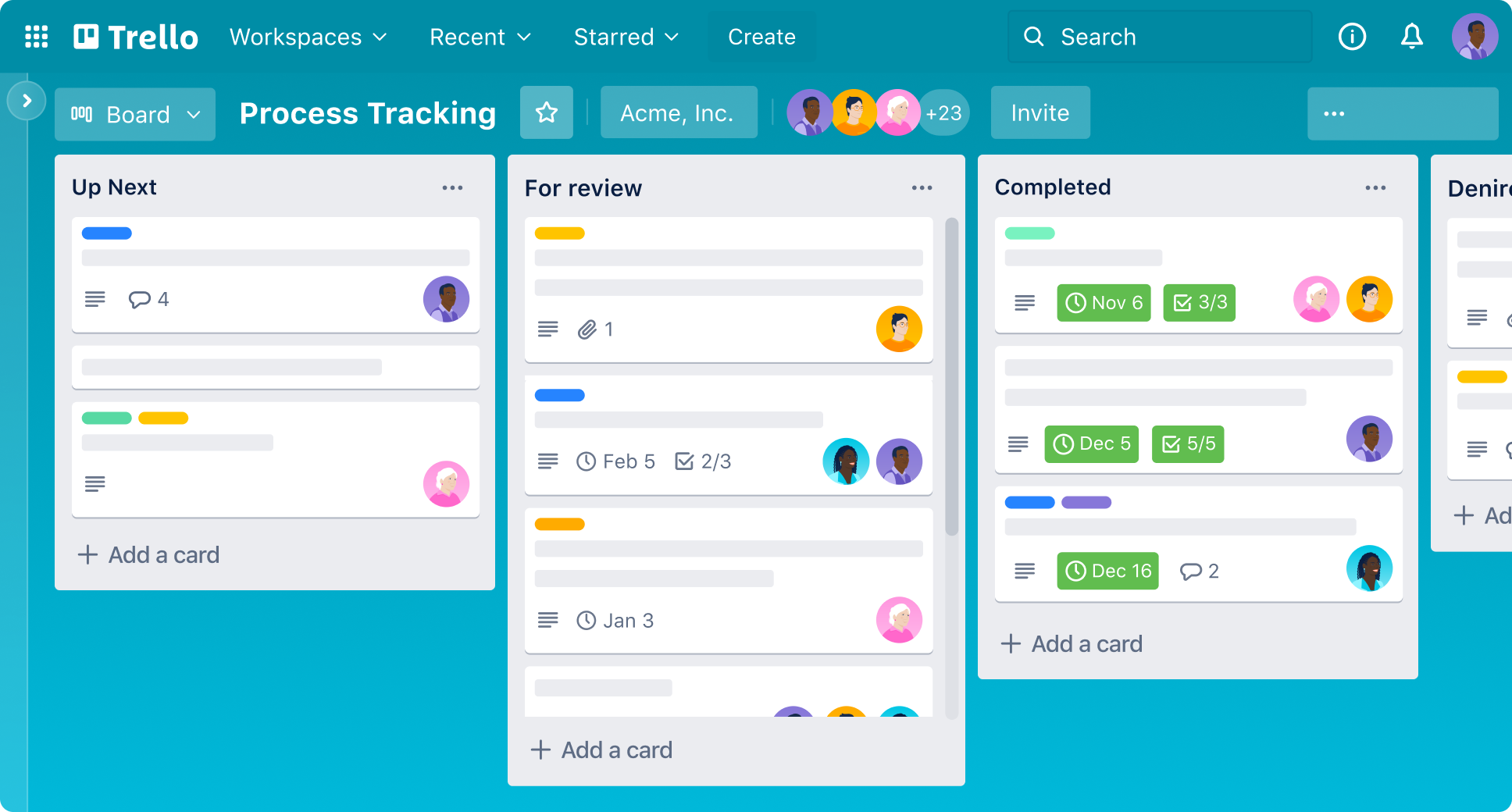
* **Resistência à mudança:** Alguns utilizadores podem revelar dificuldades à adaptação a novas ferramentas ou processos de organização.
* **Custos iniciais elevados:** O investimento necessário no desenvolvimento, formação e infraestruturas tecnológicas pode representar um encargo inicial significativo.
* **Risco de dependência tecnológica:** O funcionamento diário da organização pode tornar-se excessivamente dependente do sistema.
* **Questões de privacidade e segurança:** O tratamento de dados digitais envolve riscos relacionados com a proteção da informação e a conformidade com regulamentos como o RGPD.

De forma geral, os benefícios esperados superam os impactos negativos, desde que sejam aplicadas medidas de redução adequadas, nomeadamente formação aos utilizadores, monitorização contínua do desempenho do sistema e políticas rigorosas de segurança e privacidade para a assegurar o bom funcionamento do mesmo.

## Análise Concorrencial

Nesta secção vão ser analisados três sistemas concorrenciais que apresentam funcionalidades semelhantes ao sistema que vamos aqui desenvolver, nomeadamente no que diz respeito à gestão de tarefas, acompanhamento do progresso das equipas e ao controlo de desempenho e tendo sempre por base o modelo de Kaban e esses três exemplos são o Trello, Asana e também a aplicação que está a ser usada para a realização deste projeto, o Jira.

### Trello

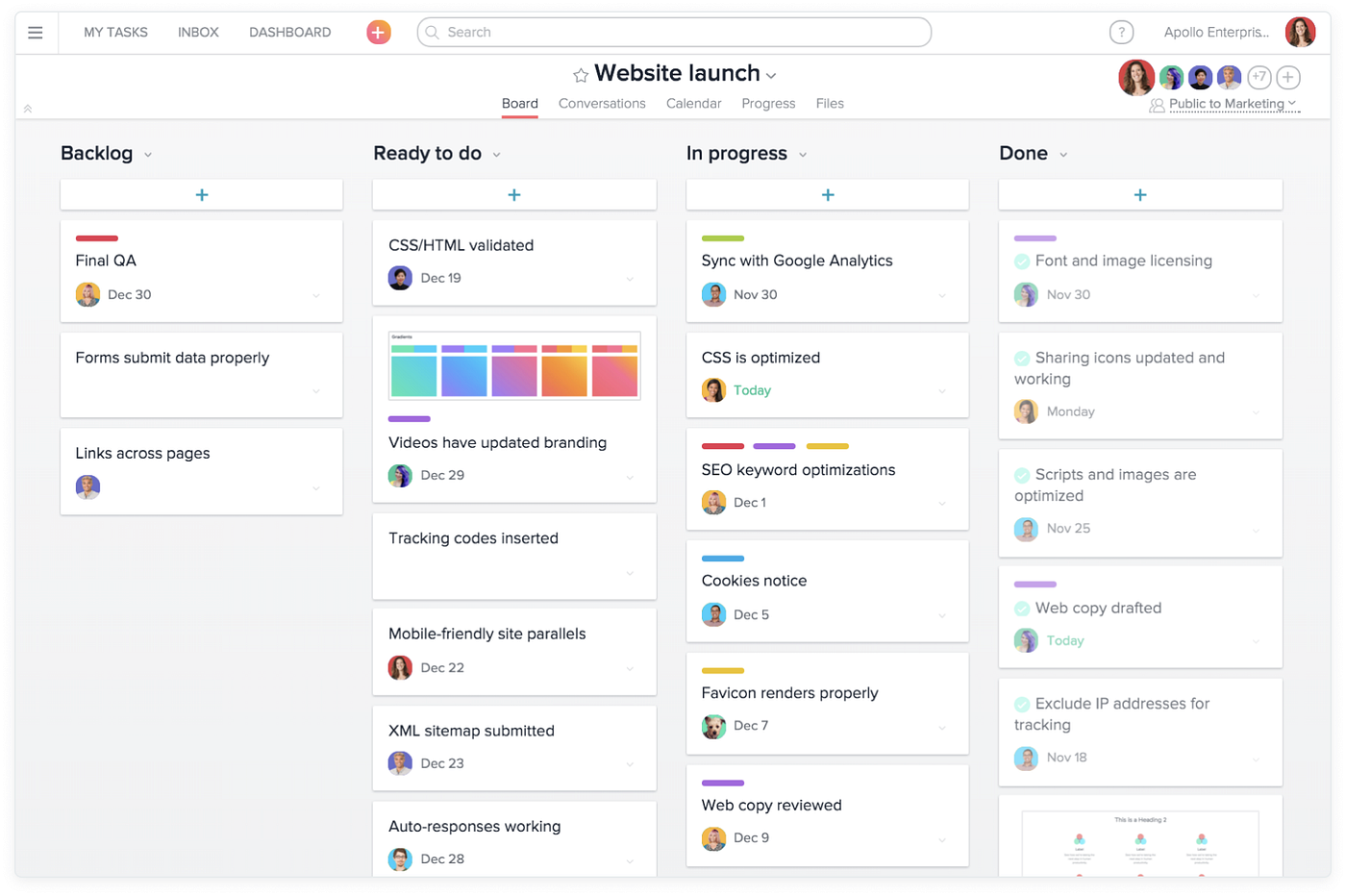


A próxima tabela resume as características do sistema:

|  |  |
| --- | --- |
| Nome | Trello |
| Site | https://trello.com |
| Descrição: | O Trello é uma aplicação web e móvel baseada no método Kanban. Permite criar quadros (boards), listas e cartões (cards) para organizar tarefas, atribuir responsáveis e acompanhar o progresso do trabalho em tempo real. É amplamente usado por equipas para planeamento colaborativo e gestão de projetos. |
| Vantagens: | Interface bastante simples e intuitiva; integração com várias ferramentas (Google Drive, Slack, etc.); sistema de arrastar e largar muito eficiente; suporte a automações. |
| Desvantagens: | Limitações na versão gratuita; falta de relatórios detalhados de produtividade; ausência de controlo de tarefas em simultâneo (WIP). |
| O que falta: | Ferramentas nativas de avaliação de desempenho e métricas detalhadas de eficiência por utilizador. |

*Tabela 1 – Descrição do Trello*

### Asana

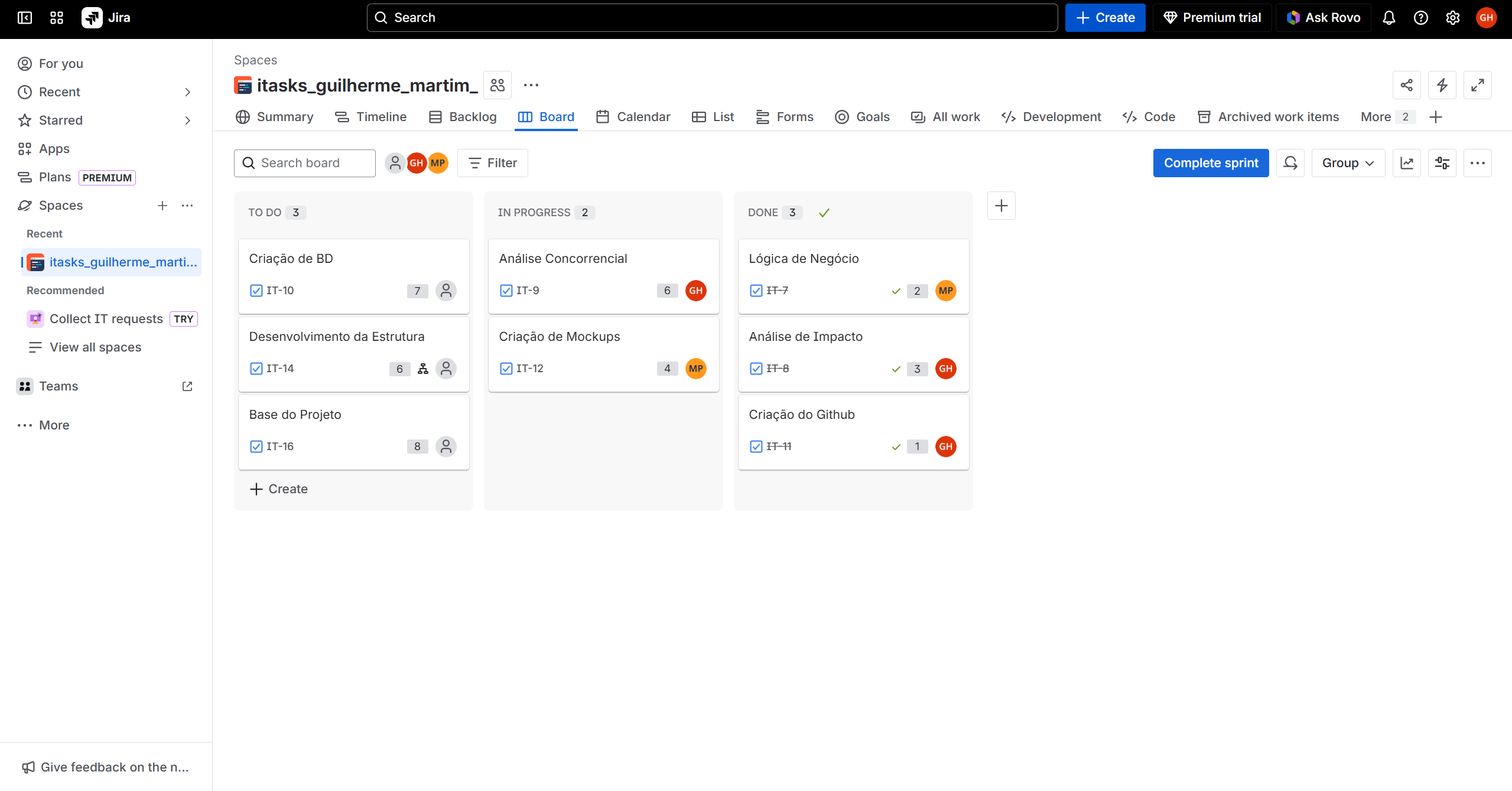


A próxima tabela resume as características do sistema:

|  |  |
| --- | --- |
| Nome | Asana |
| Site | https://asana.com |
| Descrição: | Asana é uma plataforma de gestão de projetos que permite às equipas planear, organizar e acompanhar o trabalho em diferentes formatos (listas, cronogramas, Kanban). É focada em melhorar a colaboração e transparência entre os membros da equipa. |
| Vantagens: | Visualizações variadas (lista, cronograma, Kanban); gestão de dependências entre tarefas; geração de relatórios automáticos; integrações com diversas plataformas. |
| Desvantagens: | Curva de aprendizagem inicial elevada; interface sobrecarregada em projetos grandes; algumas funcionalidades avançadas requerem planos pagos. |
| O que falta: | Sistemas de limitação de tarefas em execução (WIP) e registo automático de tempos de execução |

*Tabela 2 – Descrição do Asana*

### Jira



A próxima tabela resume as características do sistema...

|  |  |
| --- | --- |
| Nome | Jira |
| Site | https://www.atlassian.com/software/jira |
| Descrição: | O Jira, desenvolvido pela Atlassian, é uma plataforma profissional de gestão de projetos, muito utilizada em equipas de desenvolvimento de software. Baseia-se em metodologias ágeis (Scrum e Kanban) e oferece recursos robustos para planeamento, acompanhamento de progresso e relatórios de desempenho. |
| Vantagens: | Suporte total a metodologias ágeis (Scrum, Kanban); relatórios e gráficos automáticos (burn-up, burndown, velocity); controlo detalhado de tarefas e papéis; integrações com múltiplas ferramentas DevOps. |
| Desvantagens: | Complexidade de configuração inicial; interface menos intuitiva para utilizadores novos; custo elevado em equipas grandes. |
| O que falta: | Curva de aprendizagem mais suave e limitação nativa de tarefas em execução por utilizador (WIP manual). |

*Tabela 3 – Descrição do Jira*

### Comparação dos Sistemas

*Tabela 4 – Resumo das características dos Sistemas concorrenciais*

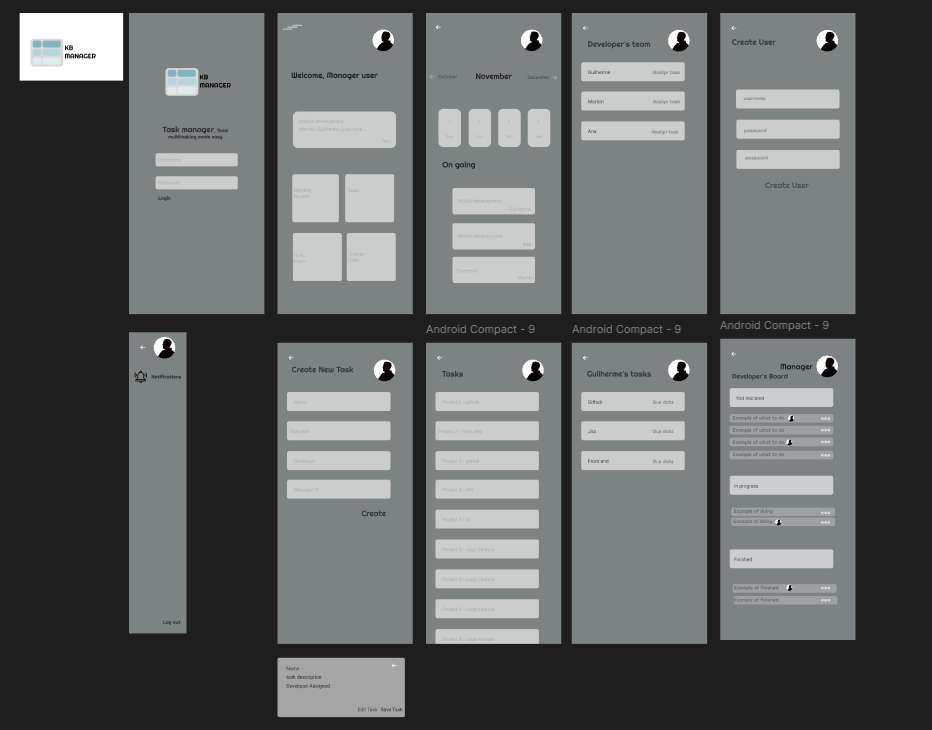
|  |  |  |  |
| --- | --- | --- | --- |
| Características/funcionalidades | Trello | Asana | Jira |
| Interface intuitiva e visualmente simples | Sim | Sim | Não |
| Utilização do modelo Kaban | Sim | Sim | Sim |
| Relatórios de desempenho | Não | Sim | Sim |
| Limite de tarefas em execução (WIP) | Não | Não | Não |
| Registo automático de datas | Não | Não | Sim |
| Gestão de funções de (Gestor/Programador) | Não | Sim | Sim |
| Integração com outras plataformas | Sim | Sim | Sim |
| Versão gratuita funcional | Sim | Sim | Sim |
| Adequado a equipas pequenas | Sim | Sim | Não |

### Enquadramento da análise concorrencial no SI

A análise dos três sistemas permitiu identificar boas práticas essenciais e falhas que existe no mercado.  
 Cada uma destas aplicações tem grandes ferramentas no qual vamos tentar aplicar na nossa app e melhorar as suas falhas, por exemplo o Trello destaca-se pela simplicidade e facilidade de uso, mas falha em métricas e controlo de tarefas de forma simultâneas. O Asana oferece uma gestão mais estratégica, com visualizações variadas, mas não controla o fluxo de execução e por fim o Jira, embora robusto e completo, é excessivamente complexo para equipas pequenas e requer uma bastante acentuada na curva de aprendizagem o que pode causar atrasos e falhas nas equipas.A nossa aplicação visa evitar estas falhas aqui descritas e adicionar ferramentas de inovação.

O sistema que aqui vamos desenvolver pretende combinar a simplicidade e fluidez do Trello, a estrutura de organização do Asana e o controlo e métricas do Jira, acrescentando uma funcionalidade diferenciadora:  
 limitação automática do número de tarefas em execução (WIP) e registo automático das datas de criação, início e conclusão, permitindo uma avaliação mais precisa da produtividade e eficiência da equipa.

## Wireframes/Mockups



*Figura 1 – Wireframe/Mockup do ecrã principal (no exemplo: esq. Wireframe; dir. Mockup)*

# Scrum

## Aplicação do Scrum ao Projeto

Ao longo do desenvolvimento do projeto, a nossa experiência com a metodologia ágil Scrum passou por uma evolução significativa. Inicialmente, a aplicação do Scrum revelou-se bastante difícil para a equipa: houve dificuldades em seguir os eventos, em definir de forma clara as tarefas e em manter uma comunicação eficiente. No entanto, à medida que o projeto avançou, fomos ajustando a metodologia à nossa realidade, aprendemos com os erros e melhoramos a organização interna. Essa adaptação permitiu-nos concretizar o projeto com eficácia e colaboração.

### **Contextualização da metodologia ágil**

O Scrum foi utilizado como estrutura principal de gestão do trabalho, mas com grandes **adaptações**, sobretudo na frequência das reuniões, na forma de documentar as tarefas e na divisão do trabalho entre elementos da equipa. A metodologia não funcionou como esperado, mas foi sendo ajustada ao longo do projeto e acabou por ser de certa forma útil.

### **Forma como foi aplicado o Scrum e principais adaptações**

* Simplificámos a criação do *Product Backlog* para torná-lo mais direto e prático.
* As tarefas foram divididas em *User Stories* mais pequenas ao longo das primeiras semanas, quando percebemos que as histórias iniciais eram demasiado amplas o que causou bastantes atrasos.
* A comunicação foi reforçada através de canais digitais para garantir que todos os membros se mantinham alinhados entre sprints.
* Os *Daily Scrums* foram utilizados para ajustar à disponibilidade real dos membros e foram alterados para Scrums de 3 em 3 dias o que facilitou a comunicação da equipa.

### **Formato das reuniões**

As reuniões decorreram num formato:

**Digitalmente**, através da plataforma online (ex.: Discord ), especialmente para os *Scrums* e para esclarecimentos rápidos ao decorrer do tempo.

### **Eventos Scrum seguidos**

A equipa adotou os principais eventos formais do Scrum:

* **Sprint Planning**

**Daily Scrum – adaptado para apenas de 3 em 3 dias**

* **Sprint Review**
* **Sprint Retrospective**

Nem todos foram seguidos de forma rígida nos primeiros sprints, mas com o tempo implementámo-los de forma mais disciplinada.

### **Detalhes das Sprints**

**Número de Sprints:** 4

**Duração de cada Sprint:** normalmente **1 a 2 semanas**, ajustada conforme o ritmo do projeto.

**Definition of Done (DoD):** considerávamos uma tarefa concluída quando:

* Estava implementada e testada;
* Era validada por ambos os membros responsáveis pelo controlo de qualidade/validação interna;

### **Ferramentas utilizadas**

* **Jira** para gestão do backlog e das tarefas;
* **GitHub** para controlo de versões e colaboração no código;
* **Discord / WhatsApp** para comunicação rápida e reuniões digitais;
* **Google Drive** para partilha de documentação.

## Stakeholders e Scrum Team

*Tabela 5 – Identificação e funções dos Stakeholders e Scrum Team*

|  |  |  |
| --- | --- | --- |
|  | Nome | Funções |
| Cliente | Professor Paulo Meira | * Avaliar o produto final |
| Product Owner | Martim Pinto  Guilherme Hutchinson | * Garantir a qualidade durante o progresso do produto |
| Scrum Master | Professor Paulo Meira | * Garantir o progresso continuo do projeto ao longo do tempo |
| Development Team | Martim Pinto  Guilherme Hutchinson | * Criação do produto |

## Product backlog

## Execução do projeto

<Product backlog do projeto:

* Inicial
* Sprint Backlog 1
* Sprint Backlog 2
* Sprint Backlog 3
* Sprint Backlog 4

Cada item do Product Backlog deve corresponder a uma Issue (Jira) do tipo Task, Story ou Bug. As issues devem ser estimadas em Story Points utilizando a sequência de Fibonacci: 1, 2, 3, 5, 8, 13, 21.

### Sprint 1 (16/10/2025 a 30/10/2025)

De seguida encontram-se descritos os principais eventos Scrum da Sprint 1.

#### Sprint Planning

|  |  |
| --- | --- |
|  |  |
| Data: | <1 de Maio de 2021> |
| *Sprint Backlog*: Lógica de Negócio, Análise de Impacto e Concorrencial, Criação de Mockups e do Github | |

#### Daily Meetings (1 por semana)

|  |  |
| --- | --- |
|  |  |
| Data: | 19/10/2025 |
| Guilherme Hutchinson   * O que fez na semana anterior: Nada * O que vai fazer esta semana: Criação do Github , Análise de Impacto e Concorrencial * Dificuldades que prevê: Análise Concorrencial   Martim Pinto   * O que fez na semana anterior: Nada * O que vai fazer esta semana: Criação de Mockups e Lógica de Negócio * Dificuldades que prevê: Lógica de Negócio | |

|  |  |
| --- | --- |
|  |  |
| Data: | 26/10/2025 |
| Guilherme Hutchinson   * O que fez na semana anterior: Criação do Github , Análise de Impacto e Concorrencial * O que vai fazer esta semana: Organizar o sistema de projeto * Dificuldades que prevê: Estruturar o trabalho   Martim Pinto   * O que fez na semana anterior: Criação de Mockups e Lógica de Negócio * O que vai fazer esta semana: Design final dos Mockups * Dificuldades que prevê: Dificuldades com Design e ajustar para o front-end | |

#### Sprint Retrospective

|  |  |
| --- | --- |
|  |  |
| Data: | 30/10/2025 |
| *Conclusões*: Inicio um pouco lento, mas tentar organizar o trabalho para o próximo Sprint | |

### Sprint 2 (30/10/2025 a 13/11/2025)

De seguida encontram-se descritos os principais eventos Scrum da Sprint 2.

#### Sprint Planning



|  |  |
| --- | --- |
|  |  |
| Data: | 30/10/2025 |
| *Sprint Backlog*: Mockup V2 | |

#### Daily Meetings (1 por semana)

|  |  |
| --- | --- |
|  |  |
| Data: | 02/11/2025 |
| Guilherme Hutchinson   * O que fez na semana anterior: Estruturar o trabalho * O que vai fazer esta semana: Começar o backend * Dificuldades que prevê: A estrutura do backend   Martim Pinto   * O que fez na semana anterior: Design final dos Mockups * O que vai fazer esta semana: Mockup diferente, versão 2 * Dificuldades que prevê: A dificuldade de Design do Mockup | |

|  |  |
| --- | --- |
|  |  |
| Data: | 09/11/2025 |
| Guilherme Hutchinson   * O que fez na semana anterior: Aprendizagem sobre o Backend * O que vai fazer esta semana: Inicio do Backend * Dificuldades que prevê:   Martim Pinto   * O que fez na semana anterior: Mockup diferente, versão 2 * O que vai fazer esta semana:Aprendizagem sobre o Frontend * Dificuldades que prevê: | |

#### Sprint Retrospective

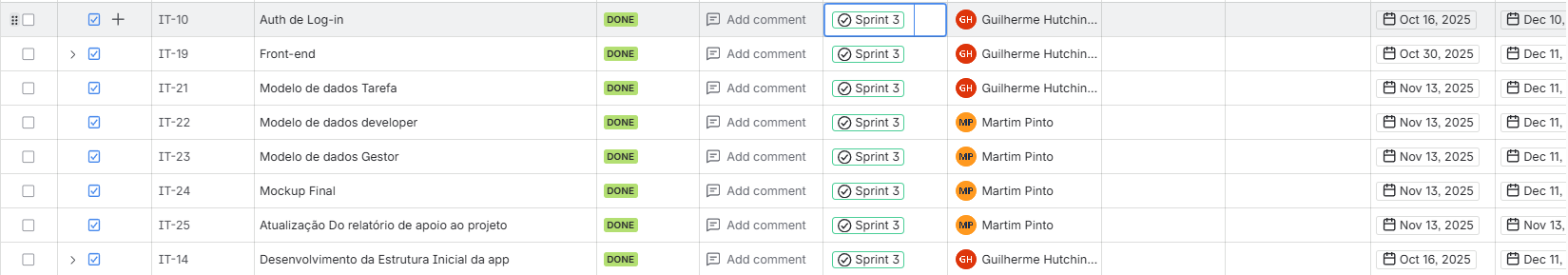
|  |  |
| --- | --- |
|  |  |
| Data: | 13/11/2025 |
| *Conclusões*: um Sprint pouco produtivo, pois falta conhecimento sobre o que fazer a seguir no projeto | |

### Sprint 3 (13/11/2025 a 27/11/2025)

De seguida encontram-se descritos os principais eventos Scrum da Sprint 3.

#### Sprint Planning

|  |  |
| --- | --- |
|  |  |
| Data: | 13/11/2025 |
| *Sprint Backlog*: | |



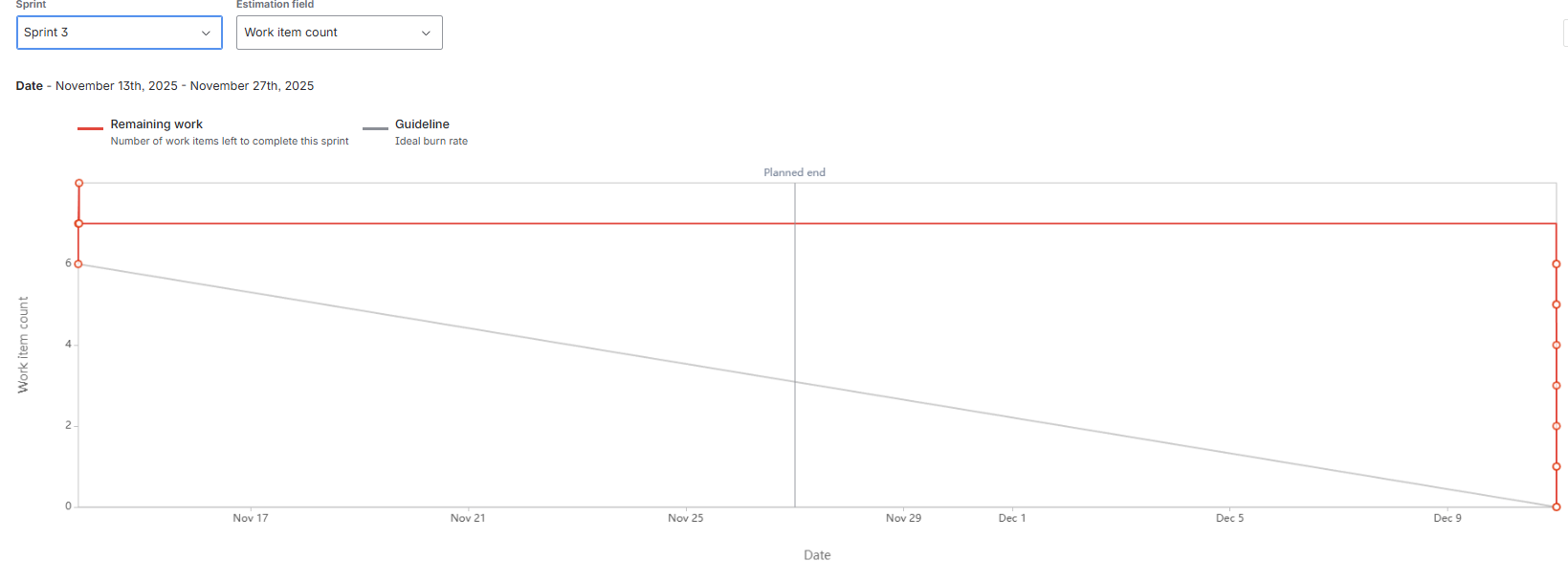
#### Daily Meetings (1 por semana)

|  |  |
| --- | --- |
|  |  |
| Data: | 16/11/2025 |
| Guilherme Hutchinson:   * O que fez na semana anterior: Modelo de Dados Tarefa * O que vai fazer esta semana: resolver erros relacionados ao Frontend * Dificuldades que prevê: Resolução de erros   Martim Pinto   * O que fez na semana anterior: Aprendizagem sobre o Frontend , fazer a ligaçao com Firebase * O que vai fazer esta semana: Finalização do Mockup * Dificuldades que prevê: | |

|  |  |
| --- | --- |
|  |  |
| Data: | 23/11/2025 |
| Guilherme Hutchinson   * O que fez na semana anterior: resolver erros relacionados ao Frontend * O que vai fazer esta semana: Continuação do Front-end * Dificuldades que prevê:   Martim Pinto   * O que fez na semana anterior: Finalização do Mockup * O que vai fazer esta semana: Criação dos Modelos, Resolução de problemas do Backend e com o Firebase * Dificuldades que prevê: | |

#### Sprint Retrospective

|  |  |
| --- | --- |
|  |  |
| Data: | 27/11/2025 |
| *Conclusões*: Uma melhoria na adaptação ao sistema Scrum com avanços no projeto, como o início do front-end mas com grandes dificuldades ao manter um ritmo de trabalho que seja estável e continuo | |



### Sprint 4 (27/11/2025 a 11/12/2025)

De seguida encontram-se descritos os principais eventos Scrum da Sprint 4.

#### Sprint Planning

|  |  |
| --- | --- |
|  |  |
| Data: | 27/11/2025 |
| *Sprint Backlog*: | |

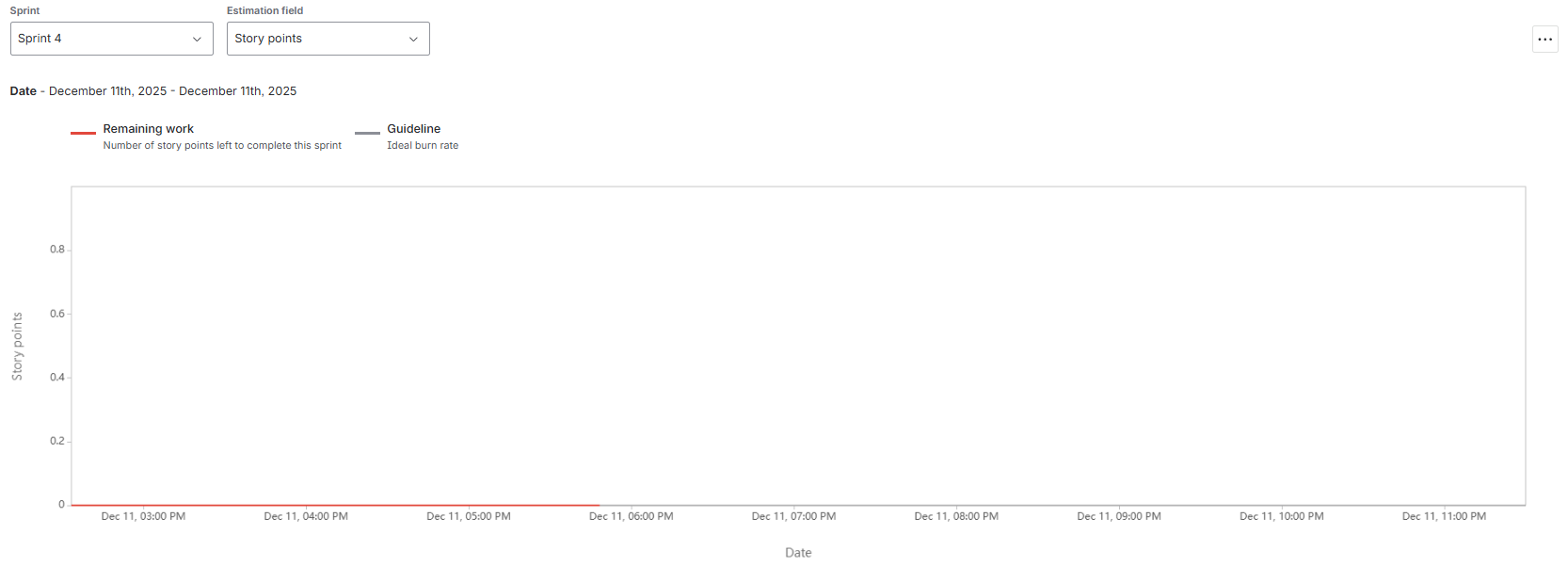
#### Daily Meetings (1 por semana)

|  |  |
| --- | --- |
|  |  |
| Data: | 30/11/2024 |
| Guilherme Hutchinson   * O que fez na semana anterior: Reorganização do Projeto em Geral * O que vai fazer esta semana: Continuação do Front-end * Dificuldades que prevê:Manter o Design Inicial   Martim Pinto   * O que fez na semana anterior: Reorganização do Projeto em Geral * O que vai fazer esta semana: aprender mais sobre Firebase e concluir * Dificuldades que prevê: utilização do Firebase | |

|  |  |
| --- | --- |
|  |  |
| Data: | 07/12/2025 |
| Guilherme Hutchinson   * O que fez na semana anterior: alterações no design da app * O que vai fazer esta semana: resolver problemas relacionados ao mobile, atualização do relatório * Dificuldades que prevê:   Martim Pinto   * O que fez na semana anterior: aprender mais sobre Firebase e concluir * O que vai fazer esta semana: atualizar o README e todo o GitHub * Dificuldades que prevê: | |

#### Sprint Retrospective

|  |  |
| --- | --- |
|  |  |
| Data: | 11/12/2025 |
| *Conclusões*: bom progresso projeto em geral e finalização do mesmo com algumas dificuldades, nomeadamente a parte do mobile. | |



## *Retrospective Summary* do Projeto

|  |  |
| --- | --- |
|  |  |
| *Things that went well* | |
| A equipa fez um esforço tremendo para fazer este projeto e entregar o mesmo com os objetivos estabelecidos e com bom rigor na data indicada, os pontos fortes do nosso projeto passam pela entreajuda de conhecimentos dos dois elementos envolventes no mesmo, um esforço para conseguir encontrar consenso de horários para as Daily Meetings e na aprendizagem das linguagens de programação utilizadas no projeto o que pode ajudar em projetos futuros, por fim | |
| *Things that could have gone better* | |
| O que podia ter corrido melhor foi a organização de todo o projeto que ficou há quem do esperado causando diversos problemas ao longo deste, o sistema de SCRUM não foi uma tarefa fácil de adaptar causando assim um grande atraso no mesmo, a utilização do GitHub é o ponto mais fraco do projeto, a falta de uso pelo facto dos elementos não terem conhecimento da utilização do mesmo causou com que o projeto apenas fosse enviado após estar completo , um ponto a investir em projetos futuros, o fraco conhecimento da linguagens de programação usada e especialmente na base de dados , onde os elementos deixaram grande parte do seu tempo para continuar a trabalhar no projeto e para concluir outro ponto que ficou deixado de parte por falta de conhecimento foi a plataforma Jira que na opinião dos elementos torna-se muito confusa e por isso foi utilizada com o rigor necessário para estar 100% completa. | |
| *Things that surprised us* | |
| Os pontos surpreenderam no nosso projeto passam pelo esforço despendido para trabalhar no projeto sacrificando o tempo pessoal, A capacidade da equipa em manter elevados níveis de esforço e compromisso, mesmo perante desafios inesperados. A entreajuda constante entre os elementos do grupo, que permitiu ultrapassar dificuldades técnicas, especialmente na aprendizagem de novas linguagens de programação. A facilidade com que foi possível, apesar das agendas diferentes, encontrar consenso para agendar as Daily Meetings, garantindo comunicação regular. A evolução rápida na compreensão das tecnologias utilizadas, que acabou por ser maior do que inicialmente previsto. | |
| *Lessons learned* | |
| A importância de uma boa organização inicial: a falta de planeamento estruturado tornou o processo mais longo, complexo e penoso. A necessidade de dominar ferramentas como GitHub e Jira para evitar atrasos e assegurar um ritmo de trabalho contínuo. A compreensão de que a adaptação ao método SCRUM exige tempo, disciplina e prática — algo que poderia ter sido preparado. O reconhecimento de que o conhecimento limitado em certas áreas, principalmente em bases de dados e nas linguagens escolhidas, impactou negativamente o ritmo e a segurança do trabalho e a falta de previsão do mesmo custou demasiado tempo que podia ter sido evitado.A importância de distribuir tarefas por fases mais pequenas para evitar que grandes blocos de trabalho se acumulem no final.  *Final Thoughts* | |
| *Things to keep*:  A cooperação e entreajuda entre os membros da equipa, que se provou fundamental para superar desafios, a realização regular de Daily Meetings, uma prática que melhorou a comunicação e o alinhamento, a motivação em aprender e evoluir tecnicamente, já que esta postura facilitou o desenvolvimento do projeto e será valiosa em trabalhos futuros, e o compromisso demonstrado pela equipa para entregar o projeto no prazo e com o rigor acordado.  *Things to change*:  Melhorar a organização e o planeamento global do projeto desde o início, definindo tarefas, prazos e prioridades de forma mais clara, investir mais tempo na aprendizagem e prática de ferramentas essenciais como GitHub e Jira para garantir um uso eficiente e contínuo durante todo o projeto, reforçar o domínio das linguagens de programação e da base de dados antes do arranque para evitar atrasos causados por dificuldades técnicas, optar por uma implementação mais estruturada e disciplinada do SCRUM, garantindo que todos compreendem e aplicam corretamente as suas práticas, e assegurar que o projeto é criado e enviado de forma incremental e não apenas no final, facilitando o controlo de qualidade e permitindo correções atempadas. | |

# Conclusão

### **Resumo do Estado Final do Software**

O projeto foi concluído e entregue na data estabelecida, cumprindo os objetivos inicialmente definidos. Apesar dos desafios enfrentados ao longo do desenvolvimento, a equipa conseguiu apresentar uma solução funcional que responde às necessidades identificadas. O esforço conjunto e a dedicação dos dois elementos da equipa foram determinantes para alcançar este resultado.

### **Conformidade com os Requisitos Definidos**

O software desenvolvido está em conformidade com os requisitos acordados no início do projeto. Ainda que o percurso tenha sido marcado por dificuldades técnicas e organizacionais, a equipa assegurou que as funcionalidades essenciais fossem implementadas com o rigor necessário. A entrega final reflete o compromisso assumido, mesmo quando isso exigiu sacrifícios de tempo pessoal.

### **Estado do Backlog**

O backlog foi concluído na sua maioria. No entanto, devido às limitações de tempo e às dificuldades de adaptação às ferramentas utilizadas, algumas funcionalidades secundárias podem beneficiar de refinamento futuro. A gestão do backlog através do Jira revelou-se desafiante pela falta de familiaridade com a plataforma, o que dificultou o acompanhamento detalhado das tarefas ao longo do projeto.

### **Riscos Identificados Durante o Desenvolvimento**

Os principais riscos identificados foram:

* Conhecimento limitado das linguagens de programação e da base de dados utilizada
* Falta de experiência com ferramentas como GitHub e Jira
* Dificuldade na adaptação ao método SCRUM
* Organização inicial insuficiente, que gerou acumulação de trabalho nas fases finais
* Agendas divergentes dos elementos da equipa, potencialmente comprometendo a comunicação

### **Medidas de Mitigação**

Para mitigar estes riscos, a equipa adotou as seguintes estratégias: dedicação de tempo extra para aprendizagem das tecnologias durante o desenvolvimento, realização consistente de Daily Meetings para manter o alinhamento e a comunicação, e entreajuda constante entre os elementos do grupo para ultrapassar dificuldades técnicas.

### **Problemas Ocorridos e Resolução**

O problema mais significativo foi a falta de organização inicial, que resultou num processo mais longo e complexo do que o previsto. A utilização tardia do GitHub — com o projeto a ser enviado apenas após estar completo — constituiu uma fragilidade importante. A adaptação ao SCRUM também causou atrasos consideráveis. Estes problemas foram resolvidos através do esforço redobrado da equipa, da flexibilidade em encontrar horários comuns e da persistência em aprender as ferramentas necessárias durante a execução do projeto.

### **Propostas para Continuidade ou Manutenção do Projeto**

Para a continuidade ou manutenção futura do projeto, recomenda-se: documentação técnica mais detalhada para facilitar intervenções posteriores, implementação de uma estrutura de versionamento adequada no GitHub, formação prévia nas tecnologias e ferramentas a utilizar, e divisão do trabalho em entregas incrementais que permitam validação contínua.

### **Conclusões Gerais**

Para além das reflexões já apresentadas no retrospective summary, importa destacar algumas considerações de âmbito mais alargado:

**Sobre o valor do projeto enquanto experiência formativa:** Este projeto transcendeu o seu objetivo técnico imediato. Mais do que desenvolver software, a equipa teve a oportunidade de vivenciar, na prática, a complexidade de um ciclo de desenvolvimento real — com pressões de prazo, limitações de recursos e a necessidade de tomar decisões sob incerteza. Esta experiência constitui uma preparação valiosa para o contexto profissional.

**Sobre a importância das soft skills:** Ficou evidente que competências técnicas, por si só, não garantem o sucesso de um projeto. A capacidade de comunicar eficazmente, gerir conflitos de agenda, manter a motivação em momentos difíceis e trabalhar em equipa revelaram-se tão importantes quanto o domínio das linguagens de programação.

**Sobre a realidade do desenvolvimento de software:** O projeto evidenciou uma verdade frequentemente subestimada: o desenvolvimento de software raramente segue um caminho linear. Imprevistos, curvas de aprendizagem e a necessidade de adaptação constante fazem parte do processo. Aceitar esta realidade desde o início pode reduzir frustrações e promover uma atitude mais resiliente.

**Sobre a autoavaliação e melhoria contínua:** A elaboração deste documento de retrospetiva demonstra maturidade profissional. A capacidade de identificar falhas, reconhecer limitações e propor melhorias é uma competência essencial que a equipa desenvolveu e que terá impacto positivo em qualquer projeto futuro, independentemente da sua natureza.

**Reflexão final:** Este projeto prova que o sucesso não se mede apenas pelo produto final, mas também pelo crescimento de quem o desenvolve com este. A equipa que termina este projeto não é a mesma que o iniciou — está mais preparada, mais consciente dos seus pontos fortes e fracos, e mais capaz de enfrentar desafios futuros com confiança e realismo.