



# unopar

UNIVERSIDADE PITÁGORAS UNOPAR ANHANGUERA - MARAVILHA  
ENGENHARIA DE SOFTWARE

NATAN OGLIARI - 34466876

PROJETO DE SOFTWARE

Maravilha/SC  
2024

NATAN OGLIARI - 34466876

## PROJETO DE SOFTWARE

Trabalho de portfólio apresentado como requisito parcial  
para a obtenção de pontos para a média semestral.

Orientador: Marco Ikuro Hisatomi .

Maravilha/SC  
2024

# Sumário

	Páginas
<b>1   Introdução</b>	<b>4</b>
<b>2   Métodos</b>	<b>4</b>
<b>3   Resultados</b>	<b>5</b>
3.1   Primeira etapa . . . . .	5
3.2   Segunda etapa . . . . .	6
3.2.1 <i>Product backlog</i> . . . . .	6
3.2.2   Quadro Kanban . . . . .	6
<b>4   Conclusões</b>	<b>11</b>

# 1 Introdução

Para a disciplina de **Projeto de Software** seu objetivo é analisar as metodologias de desenvolvimento/gerenciamento de projetos ágil, visando uma melhora no uso do tempo de desenvolvimento e a própria agilidade no desenvolvimento do projeto.

De forma sucinta o movimento ágil é uma alternativa a gestão tradicional de projetos, as abordagens ágeis nos tempos atuais vem sendo de suma importância e um divisor de águas para que um projeto ou *software* se manter no mercado, pois cada vez esta um ambiente incerto e turbulento.

Segundo Smartsheet (2024), os quatro valores do manifesto ágil são:

- I Indivíduos e interações mais que processos e ferramentas
- II Software em funcionamento mais que documentação abrangente
- III Colaboração com o cliente mais que negociação de contratos
- IV Responder a mudanças mais que seguir um plano

E desta maneira possuem os **doze** princípios do manifesto ágil, e que são:

1. Satisfação do cliente por meio do fornecimento contínuo e adiantado de software.
2. Acomodação de mudanças de requisito durante todo o processo de desenvolvimento.
3. Entrega frequente de software em funcionamento.
4. Colaboração entre as partes interessadas do negócio e os desenvolvedores em todo o projeto.
5. Apoio, confiança e motivação das pessoas envolvidas.
6. Possibilidade de interações presenciais.
7. Software em funcionamento é a principal medida de progresso.
8. Processos ágeis para dar apoio a um ritmo de desenvolvimento consistente.
9. Atenção aos detalhes técnicos e ao design aumenta a agilidade.
10. Simplicidade.
11. Equipes auto-organizáveis realizam ótimas arquiteturas, requisitos e designs.
12. Reflexões periódicas sobre como aumentar a eficácia.

(SMARTSHEET, 2024)

Esta aula prática tem por objetivo o desenvolvimento das etapas de um projeto hipotético ágil, com o fim de aplicar os conhecimentos adquiridos na cadeira de **projeto de software**.

## 2 Métodos

Para o desenvolvimento desta aula prática foi disponibilizado um roteiro, e segundo este roteiro é proposto elaboração em duas etapas, sendo elas:

1. Se colocar no lugar do cliente, e pensar em um aplicativo para a confecção.
2. E se colocar no lugar do *Product Owner*, que deverá elaborar o aplicativo proposto.

**Figura 1.** Logo Trello

Após as análise iniciais, o roteiro propoe o uso de algumas ferramentas para controle de etapas (**IceScrumm, Trello e Asna**), opto pelo uso do Trello, esta ferramenta proporciona a integração com o GitHub, adicionado e comentando ramificações e solicitações de mesclagem de fragmentos de códigos.



Fonte: logowik (2024)

### 3 Resultados

A elaboração desta aula prática esta dividida em duas partes, a primeira, no qual serei o cliente e deverá propor um aplicativo que se deseja desenvolver, levantar as funcionalidades e características. Na segunda etapa, deverá ser o *Product Owner*, quem ira desenvolver o aplicativo proposto e as responsabilidades desta etapa são:

- Definir as funcionalidades do produto, ou seja, desenvolver o product backlog;
- Priorizar as funcionalidades de acordo com o valor de negócio;
- Montar um quadro do Scrum (Kanban) com as divisões de etapas, tarefas, data de entrega e responsáveis por atividade. Para este item, imagine que o desenvolvimento do seu aplicativo está em um estágio mais avançado, por este motivo, deve haver tarefas em todas as etapas. Utilize uma das ferramentas propostas para montar o seu quadro.

#### 3.1 Primeira etapa

A aplicativo em questão deverá atender uma necessidade de geração de documentos personalizados, no qual o mesmo deverá ler tabelas em um arquivo com extensão pdf *Portable Documento Format*, possuem mais do que um tipo de documento gerado, deve ter a opção de escolher qual quero gerar e os documentos irão gerar para todos os nomes da tabela do arquivo pdf.

1. Buscar o arquivo com extensão pdf mais novo na area de trabalho do usuario;
2. Ler um arquivo pdf e extrair das tabelas os nomes e numeros de prontuarios;
3. Selecionar o modelo de documentos que serão gerados para todos os nomes com seus respectivos numeros de prontuarios;

4. Escolher se desejo imprimir e em qual bandeja;
5. Escolher se desejo juntar os termos em um único pdf ou deixar em arquivos separados e salvar em uma pasta na área de trabalho.

## 3.2 Segunda etapa

Nesta etapa passo de cliente para desenvolvedor da solução, e para tal utiliza-se as ferramentas de controle, versionamento de projetos e gerenciamento de projetos.

### 3.2.1 Product backlog

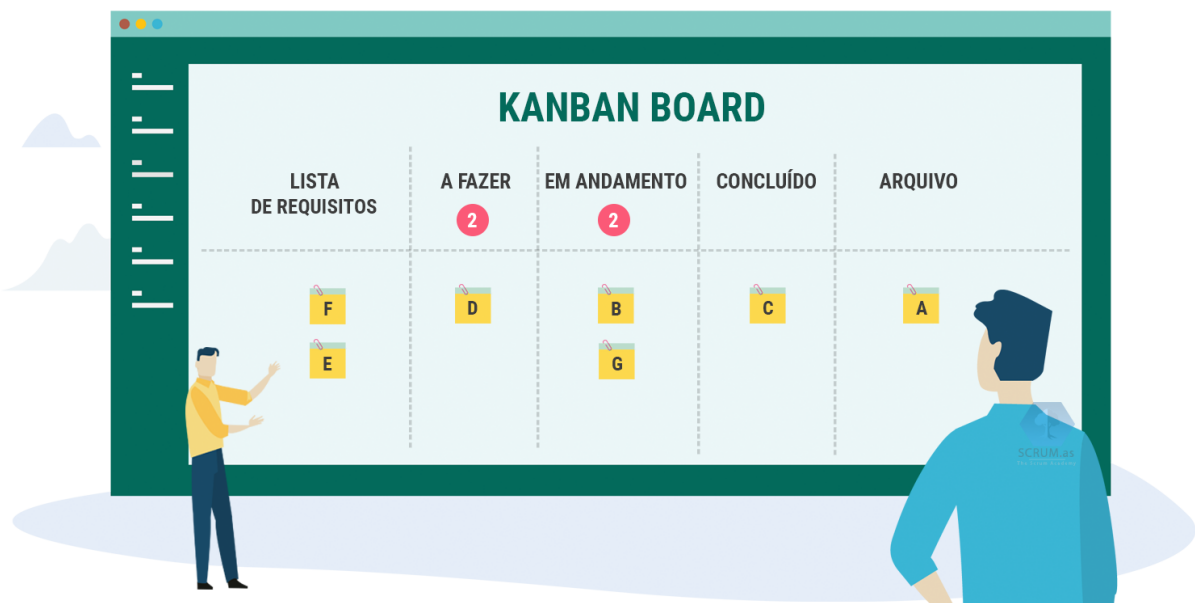
O *Product backlog*, tem a finalidade de listar todas as características, funções, requisitos e mudanças a futura que o produto precisa veja na tabela 3.2.1.

1. Tabela 3.2.1
2. fbfn

### 3.2.2 Quadro Kanban

O quadro **Kanban**, representa um fluxo de tarefas o que passa um *backlog* (Lista tarefas) e divide em sub tarefas e estas tarefas são organizadas em quadros do tipo: **A FAZER; FAZENDO e CONCLUÍDO**, observa-se um exemplo na figura 2. Na coluna **LISTA DE REQUISITOS**, seria o *backlog* que posteriormente o mesmo é dividido em sub tarefas e adicionado nas colunas **A FAZER; FAZENDO e CONCLUÍDO**.

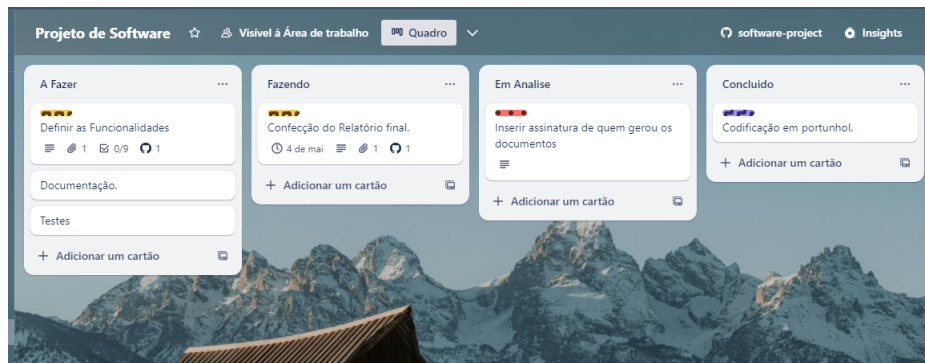
**Figura 2.** Exemplo de quadro Kanban



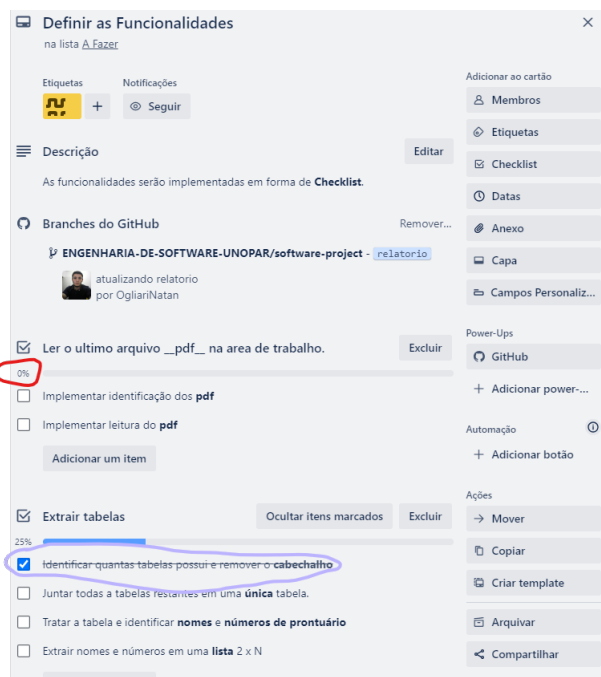
Fonte: SCRUM.AS (2024)

Deste modo prosseguimos para a elaboração do quadro **Kanban** da aula prática a qual pode ser observada na sequência das figuras 3, o quadro *Kanban* permite a inclusão de várias equipes de trabalho, porém nesta pratica é confeccionado para equipe única.

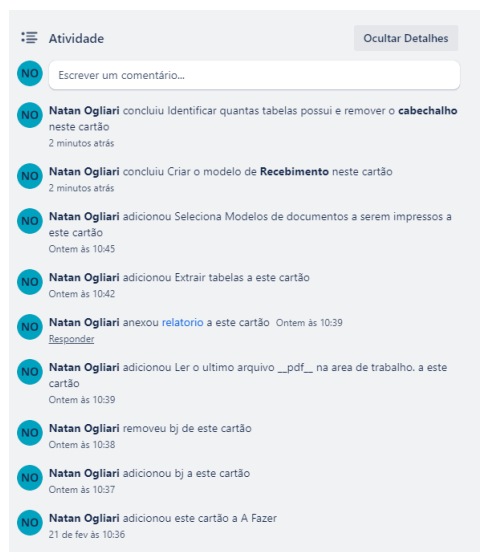
**Figura 3. Resultado da Aula prática**



(a) Quadro Geral.



(b) Fragmento do quadro.

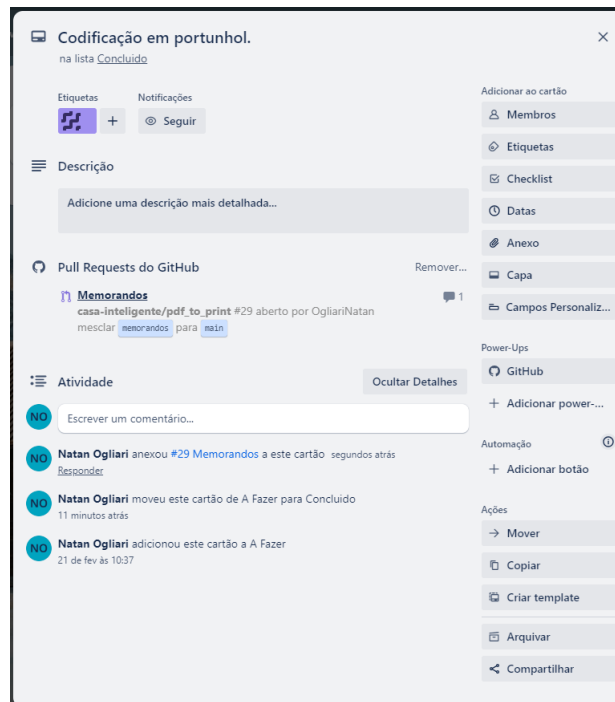


(c) Registro de atividades com o Trello.

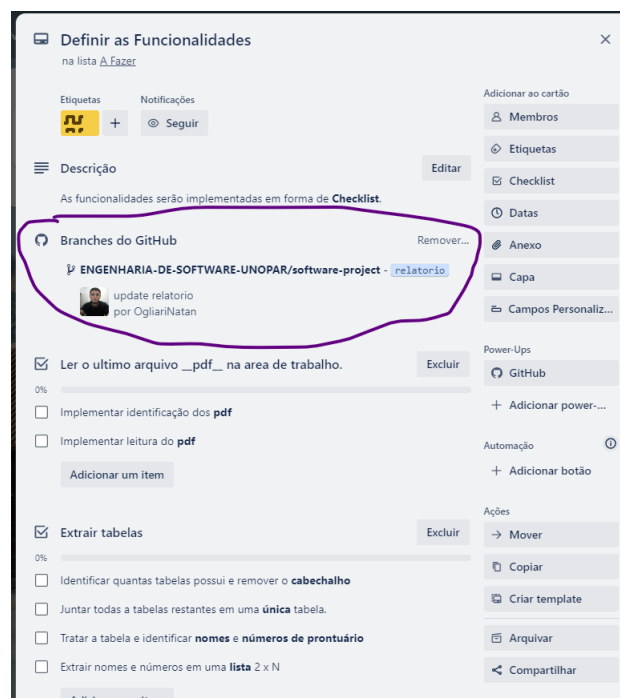


Com o uso do Trello<sup>1</sup>, para esta aula prática evidencia as funcionalidades deste SaaS, como demonstrado na figura 4(b) o qual evidencia o uso de *check list* e o percentual das subtarefas desenvolvidas, já na figura 4(c) é possível verificar os registros de atividades relacionadas a tarefa em questão sem contar na integração com a plataforma GitHub conforme demonstrado na figura 4

**Figura 4.** Resultado integração com GitHub



(a) Integração GitHub.



(b) Integração GitHub.

Fonte: O autor (2024)

## **4 Conclusões**

Podemos aferir com elaboração da aula prática, a necessidade do uso de metodologias com na gestão de software.

## Referências

LOGOWIK. **Trello**. 2024. Acessado em: 29 fev. 2024. Disponível em: <<https://logowik.com/trello-new-vector-logo-8078.html>>.

SCRUM.AS. **Fundamentos Internacionais Master Kanban**. 2024. Acessado em: 15 mar. 2024. Disponível em: <<https://www.scrum.as/academy.php?show=7&chapter=15>>.

SMARTSHEET. **Guia completo do manifesto ágil**. 2024. Acessado em: 20 fev. 2024. Disponível em: <<https://pt.smartsheet.com/comprehensive-guide-values-principles-agile-manifesto>>.