



## **Universidade Federal do Ceará - *campus* Quixadá**

### **Projeto Integrado I**

Processos, Tecnologias e Ferramentas

#### **Docente:**

Jeferson Kenedy Moraes Vieira

#### **Discentes:**

Daniel Christopher Souza Rodrigues, 552493;

Daniel Lucas Ulisses Magalhães, 554563;

João Evangelista de Souza Alves, 548346;

Julian Cesar Pereira Cardoso, 553782;

Kleyton Kluesriton Ferreira Pereira, 555425:

12.11.2024

# Processos

O framework **Scrum** será adotado com adaptações para alinhar-se melhor à rotina da equipe. Em vez de reuniões diárias formais (dailies), a equipe optará por alinhamentos informais a cada dois dias, pois os membros se encontram regularmente, o que facilita a troca contínua de informações sobre as atividades, desafios e próximos passos. Essas reuniões informais ajudam a manter o grupo alinhado e engajado.

## Papeis no Scrum:

- Daniel Rodrigues atuará como **Product Owner (P.O.)**: responsável por priorizar o backlog e garantir que as entregas se concentrem nas atividades de maior valor.
- João Evangelista será o **Scrum Master**: desempenhando a função de facilitador, assegurando que os processos ágeis sejam seguidos e removendo impedimentos para o progresso da equipe. A equipe de desenvolvimento será autogerida e colaborativa, buscando cumprir as metas estabelecidas para cada sprint.

## Estrutura das Sprints:

- As sprints terão duração de duas semanas. No início de cada sprint, será realizada uma reunião de planejamento para revisar o backlog, priorizado pelo P.O., e definir as tarefas que serão abordadas. Durante a sprint, haverá uma reunião intermediária para revisar o andamento das atividades e fazer ajustes, se necessário, assegurando que o trabalho esteja alinhado com os objetivos do projeto.

## Revisão e Retrospectiva da Sprint:

- Ao término de cada sprint, a equipe realizará uma reunião de revisão para demonstrar o que foi entregue e obter feedback do Product Owner e dos demais interessados. Em seguida, uma retrospectiva será conduzida para discutir o que funcionou bem, o que poderia melhorar e como aprimorar o processo para as próximas sprints, reforçando a prática de melhoria contínua.

## Gestão do Backlog e Kanban:

- O backlog será constantemente revisado e atualizado para garantir que as tarefas selecionadas estejam sempre alinhadas às prioridades. Para o acompanhamento das atividades, será utilizado um quadro Kanban no Trello, onde as tarefas serão distribuídas em diferentes colunas, permitindo uma visualização clara do progresso.

## Métricas de Desempenho:

- A equipe também poderá monitorar métricas como a velocidade da sprint e a taxa de conclusão das tarefas, o que contribuirá para prever a capacidade de entrega e ajustar o ritmo de trabalho.
- Essas práticas combinam a estrutura do Scrum com a flexibilidade do Kanban, criando um ambiente ágil e adaptado às necessidades da equipe, permitindo o desenvolvimento de maneira colaborativa, eficiente e com foco na entrega de valor.

# Tecnologias

- **TypeScript**: Linguagem de programação escolhida para o desenvolvimento da lógica do sistema, proporcionando tipagem estática e maior segurança no código, o que facilita a manutenção e o entendimento do projeto.

- **Electron.JS:** Ferramenta utilizada para transformar o sistema em uma aplicação desktop, possibilitando a criação de uma interface que facilita a interação dos usuários, independentemente do sistema operacional.
- **Node.JS:** Utilizado para o desenvolvimento do servidor backend do sistema, aproveitando seu ambiente assíncrono e orientado a eventos para gerenciar as requisições dos usuários de forma eficiente.
- **Express:** Framework para Node.js que simplifica o desenvolvimento do backend, agilizando a criação de APIs RESTful para integrar o sistema e gerenciar as operações de pedidos, usuários, e controle financeiro.
- **Docker:** Ferramenta de containerização que facilita a implementação do sistema em diferentes ambientes, padronizando o ambiente de desenvolvimento e permitindo uma fácil replicação das configurações para homologação e produção.
- **Insomnia:** Ferramenta de testes de APIs que será usada para validar e simular as requisições de dados do backend, garantindo que as funcionalidades de controle de pedidos, cadastro e geração de relatórios funcionem corretamente.
- **Git:** Sistema de versionamento de código para acompanhar as mudanças realizadas no código ao longo do desenvolvimento, permitindo o controle de versões e a colaboração entre a equipe.

## Ferramentas

- **Figma:** Usado para criar protótipos e wireframes da interface do sistema, facilitando o desenvolvimento de uma interface amigável e coerente com as expectativas dos usuários.
- **Trello:** Utilizado para organizar e monitorar as tarefas da equipe, seguindo o quadro Kanban e permitindo um acompanhamento visual do progresso das atividades.
- **GitHub:** Plataforma utilizada para hospedar o repositório do projeto e gerenciar o versionamento do código, facilitando a colaboração e o controle de versões com o uso de branches e pull requests.
- **VSCode:** Um editor de código versátil e popular, perfeito para quem busca uma ferramenta completa para seus projetos de desenvolvimento.
- **Astah:** Ferramenta utilizada para criar e modelar diagramas UML, como diagramas de classe, sequência e atividade. Essencial para representar a estrutura e o comportamento do sistema, facilitando a compreensão dos fluxos e das interações entre diferentes componentes.