

---

**NOME:** Pedro Frasson

**PROJETO:** Plano de contas

## **RESUMO DO PROJETO:**

**Aplicação web aonde é possível gerenciar seus gastos e rendas de tal dia, e visualizar o total final de todas elas, foi utilizado sqlite como banco de dados e Javascript como linguagem principal para desenvolver as regras de negócio**

## **REQUISITOS DO PROJETO:**

### **Motivação:**

O sistema visa proporcionar aos usuários uma ferramenta eficiente para o gerenciamento simplificado de suas finanças pessoais. A motivação é fornecer uma solução acessível, permitindo que os usuários controlem suas despesas e rendas de maneira eficaz.

### **Problema a Ser Resolvido:**

O sistema deve abordar as necessidades fundamentais dos usuários em relação ao controle financeiro, concentrando-se em funcionalidades essenciais para o registro e análise de gastos e rendas. A solução busca simplificar a gestão financeira, oferecendo uma visão clara e acessível das transações.

### **Requisitos Funcionais:**

**Interface Simples:** O sistema deve fornecer uma interface intuitiva e descomplicada para os usuários registrarem suas despesas e rendas.

**Registro de Entradas:** Os usuários devem ser capazes de registrar entradas contendo data, descrição, tipo (renda ou gasto) e valor.

---

**Armazenamento Local:** As entradas registradas pelos usuários devem ser armazenadas localmente no banco de dados alocado no dispositivo.

**Cálculo do Valor Total:** O sistema deve calcular e exibir o valor total acumulado de todas as entradas registradas.

**Requisitos Não Funcionais:**

**Usabilidade:** A interface deve ser simples e direta, facilitando o uso por usuários sem experiência técnica.

**Desempenho:** Garantir um desempenho eficiente, mesmo com um grande volume de entradas registradas.

**Armazenamento Seguro:** As informações do usuário devem ser armazenadas localmente de maneira segura.

**Considerações Adicionais:**

**Design:** O design da interface deve ser limpo e fácil de entender.

**Implementação:** Utilizar tecnologias que garantam a eficiência e a segurança no armazenamento local.

**Requisitos Físicos:** O sistema deve ser compatível com os principais navegadores web para oferecer flexibilidade aos usuários.

## **SCRUM:**

**Sprint 1: Planejamento**

**Product Backlog:**

**Criar a estrutura do aplicativo:**

---

Desenvolver os arquivos HTML, CSS e JavaScript para a estrutura básica do aplicativo.

Adicionar a funcionalidade de adicionar entradas financeiras:

Implementar a lógica que permite aos usuários adicionar novas entradas à lista.

Adicionar a funcionalidade de editar entradas:

Desenvolver a capacidade de editar o texto das entradas já existentes na lista.

Implementar a funcionalidade que permite visualizar o total de todas estas estradas.

Estilizar a interface do usuário:

Aplicar estilos à interface para torná-la atraente e fácil de usar.

Planejamento da Sprint:

Revisão detalhada do Product Backlog.

Seleção das histórias a serem incluídas na Sprint.

Estimativas de tempo para cada história.

Sprint 2: Implementação

Planejamento da Sprint:

Revisão do Product Backlog.

Seleção das histórias a serem incluídas na Sprint.

Estimativas de tempo para cada história.

Desenvolvimento:

(Continuação das atividades da Sprint 1)

---

## Sprint 3: Revisão e Retrospectiva

### Revisão da Sprint:

Apresentação do produto ao Product Owner.

Verificação pelo Product Owner se as histórias foram implementadas de acordo com os critérios de aceitação.

### Retrospectiva da Sprint:

Revisão do que funcionou bem e identificação de áreas de melhoria.

Identificação de lições aprendidas para aplicação nas próximas Sprints.

### Observações Finais:

O Scrum Master desempenha um papel crucial facilitando a colaboração e removendo obstáculos.

A comunicação contínua entre a equipe de desenvolvimento e o Product Owner é essencial.

O ciclo de Sprints se repete, permitindo adaptação contínua às mudanças e melhorias incrementais no produto.

---