

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Faculdade de Tecnologia de Araras – Antonio Brambilla**

Desenvolvimento De Software E Multiplataforma

**MATHEUS LUIS**

**THIAGO CÉSAR ALVAREZ**

**VITOR EDUARDO DE OLIVEIRA**

**PLATAFORMA: BETTERPRICE**

RELATÓRIO TÉCNICO – PROJETO INTEGRADOR IIII

**Araras – SP**

**2024**



\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Faculdade de Tecnologia de Araras – Antonio Brambilla**

Desenvolvimento De Software E Multiplataforma

**MATHEUS LUIS**

**THIAGO CÉSAR ALVAREZ**

**VITOR EDUARDO DE OLIVEIRA**

**PLATAFORMA: BETTERPRICE**

RELATÓRIO TÉCNICO – PROJETO INTEGRADOR IIII

Relatório Técnico apresentado à Faculdade de Tecnologia de Araras – Antonio Brambilla, como requisito parcial de aprovação na disciplina Projeto Integrador IIII do curso de Desenvolvimento de Software e Multiplataforma.

Orientadora: Proº Fernando Bryan Frizzarin.

**Araras - SP**

**2024**

**SUMÁRIO**

[1. PLANEJAMENTO 4](#_Toc183602354)

[2. DESCRIÇÃO DO PROJETO 5](#_Toc183602355)

[3. FUNCIONALIDADES 6](#_Toc183602356)

[3.1 DESCRIÇÃO DA FUNCIONALIDADE 6](#_Toc183602357)

[3.2 ESTÓRIA DO USUARIO: 7](#_Toc183602358)

[3.2.1. Sueli (Usuária): 7](#_Toc183602359)

[3.2.2. Mercados (Profissional): 7](#_Toc183602360)

[4. REQUISITOS TÉCNOLÓGICOS 7](#_Toc183602361)

[4.1 Tecnologias utilizadas: 7](#_Toc183602362)

[4.2 APIs: 7](#_Toc183602363)

[5. SPRINTS/BACKLOGS 7](#_Toc183602364)

[5.1 Backlogs 8](#_Toc183602365)

[5.2.1 1º Sprint: 9](#_Toc183602366)

[5.2.2 2º Sprint: 10](#_Toc183602367)

[5.2.3 3º Sprint: 11](#_Toc183602368)

[5.2.4 4º Sprint: 11](#_Toc183602369)

[5.2.4 4º Sprint: 12](#_Toc183602370)

[6. PROTOTIPAGEM 13](#_Toc183602371)

[6.1 Tela inicial: 13](#_Toc183602372)

[7. DIAGRAMAÇÃO DO PROJETO 16](#_Toc183602373)

[7.1 Caso de uso: 16](#_Toc183602374)

[7.2 diagrama entidade-relacionamento: 17](#_Toc183602375)

# PLANEJAMENTO

* **(21/08 - 03/09):**
  + Definir o escopo do projeto: funcionalidades, público-alvo e requisitos.
  + Realizar pesquisas de mercado sobre sites similares.
  + Definir meios a serem utilizados (frameworks, metodologia e configuração de IDE.
* **(04/09 - 24/09):**
  + Implementar a estrutura HTML/CSS das páginas principais.
  + Desenvolver o layout para desktop.
  + Modelagem do banco de dados
* **(25/09 - 15/10):**
  + Início do Back-End.
  + Integrar uma API de mapas (se necessário).
  + Realizar testes de usabilidade e acessibilidade.
* **(16/10 - 05/11):**
  + Integra o back-end com o front-end.
  + Configurar a API para 100% de funcionamento (caso utilizado)
  + Finalização das funcionalidades principais
* **(06/11 - 26/11):**
  + Correção de bugs (caso existam)
  + Ajustes do que se julgar necessário.
  + Ajustes do Docker (caso tenham)
* **(27/11):** 
  + Entrega

**CRONOGRAMA**

|  |  |
| --- | --- |
| Data | Eventos |
| 14/08/2024 | Definição do tema e escolha do grupo |
| 21/08/2024 | Definir um escopo e realizar as pesquisas |
| 28/08/2024 | Configurar IDE para os frameworks utilizados |
| 04/09/2024 | Início do Front-End. |
| 11/09/2024 | Ajustes necessários ao Front-End. |
| 18/09/2024 | Finalização do Front-End. |
| 25/09/2024 | Início do Back-End. |
| 02/10/2024 | Integração de API (Caso necessário) /Ajustes Back-End. |
| 09/10/2024 | Finalização do Back-End. |
| 16/10/2024 | Integra Back-End. com Front-End. |
| 23/10/2024 | Configurar API (caso tenha) /início da finalização das funcionalidades principais |
| 30/10/2024 | Finalização da API |
| 06/11/2024 | Finalização das funcionalidades/criação do Docker |
| 13/11/2024 | Ajuste do Docker (caso tenha) |
| 20/11/2024 | Correção de bugs/detalhamento final |
| 27/11/2024 | Entrega |

Fonte: Matheus Guedes, Thiago Alvarez e Vitor Oliveira, 2024

# DESCRIÇÃO DO PROJETO

Nosso projeto é uma plataforma que visa ajudar o usuário a encontrar os melhores preços em supermercados dos produtos que ele deseja consumir.

Ao navegar pelas categorias disponíveis dos setores do supermercado (padaria, açougue etc.) os usuários podem encontrar rapidamente os produtos que ele deseja. Além disso ele também pode procurar diretamente o nome do produto ou então entrar na própria página do mercado no site

A plataforma permite que os usuários possam adicionar os produtos escolhidos em uma lista para que ele tenha um lugar que ele possa consultar e lembrar qual item e em qual estabelecimento ele deve ir ao fazer compras

# FUNCIONALIDADES

# 3.1 DESCRIÇÃO DA FUNCIONALIDADE

A plataforma tem como objetivo auxiliar aqueles que estão à procura dos melhores preços nos produtos desejado, sendo assim ele poderia buscar na plataforma o produto em comparar os preços em diferentes mercados.

Critérios de Aceitação:

1. Cadastro de Profissionais:

Feito pelos Donos da plataforma, fornecendo estoque, nome e CNPJ da empresa

2. Usuários:

Como Usuário, posso buscar os itens que desejo e ver aquele que mais se adapta ao meu orçamento

Posso também adicionar os itens numa lista minha para eu não me perder em quais mercados ir.

3. Políticas de Privacidade e LGPD: Explicar como os dados dos usuários são tratados.

## 3.2 ESTÓRIA DO USUARIO:

## 3.2.1. Sueli (Usuária):

Descrição: Sueli acredita estar gastando muito em suas compras ela então usa a plataforma Betterprice, para buscar os produtos que deseja, ao encontrá-los ela então compara os preços e encontra o mercado no qual o item está mais barato e ela então o adiciona a sua lista e assim ela faz sucessivamente com suas compras

## 3.2.2. Mercados (Profissional):

Descrição: Os mercados somente utilizam da plataforma de vitrine para mostrar seus produtos para que os usuários possam comparar os melhores preços e assim ir até o estabelecimento comprar

# REQUISITOS TÉCNOLÓGICOS

## 4.1 Tecnologias utilizadas:

PostgreSQL, C#, Razor, HTML, CSS, Blazor.

## 4.2 APIs:

A autenticação via API do Google permite que os usuários façam login em um site usando suas contas do Google, proporcionando uma experiência mais rápida e segura. O processo começa com o usuário clicando em um botão de login, o que redireciona para a página de login do Google. Após inserir suas credenciais e conceder permissões, o Google gera um token de acesso e redireciona o usuário de volta ao site. O site valida esse token junto ao servidor do Google, que retorna informações do usuário, como nome e e-mail. Com esses dados, o site cria uma sessão para o usuário, facilitando o acesso. Essa abordagem reduz o risco de fraudes, melhora a experiência do usuário e minimiza a carga administrativa sobre os desenvolvedores (não implementada).

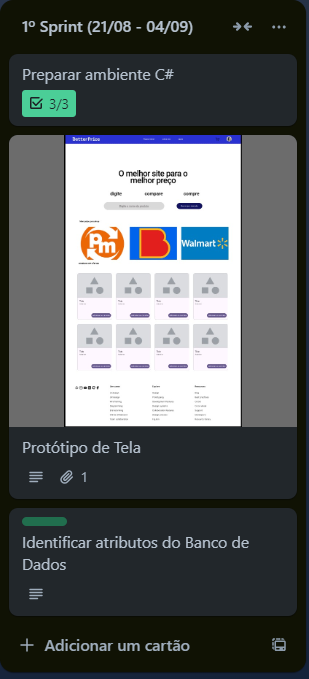
# SPRINTS/BACKLOGS

## 5.1 Backlogs

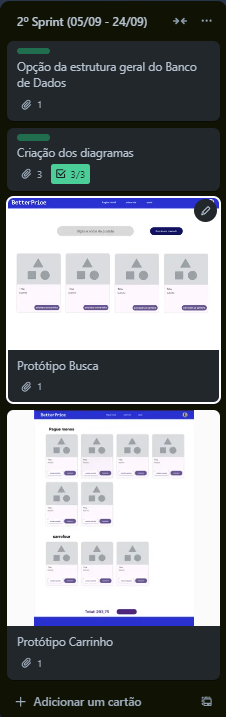
Interface gráfica do usuário, Texto, Aplicativo, chat ou mensagem de texto

Descrição gerada automaticamente

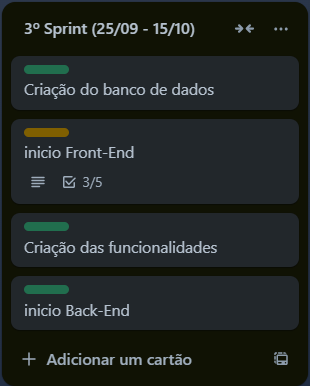
## 5.2.1 1º Sprint:



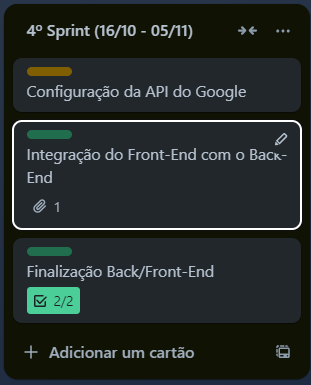
## 5.2.2 2º Sprint:



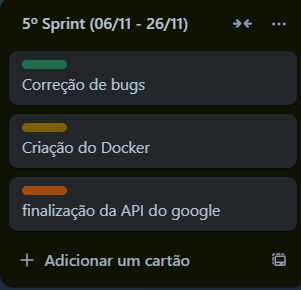
## 5.2.3 3º Sprint:



## 5.2.4 4º Sprint:

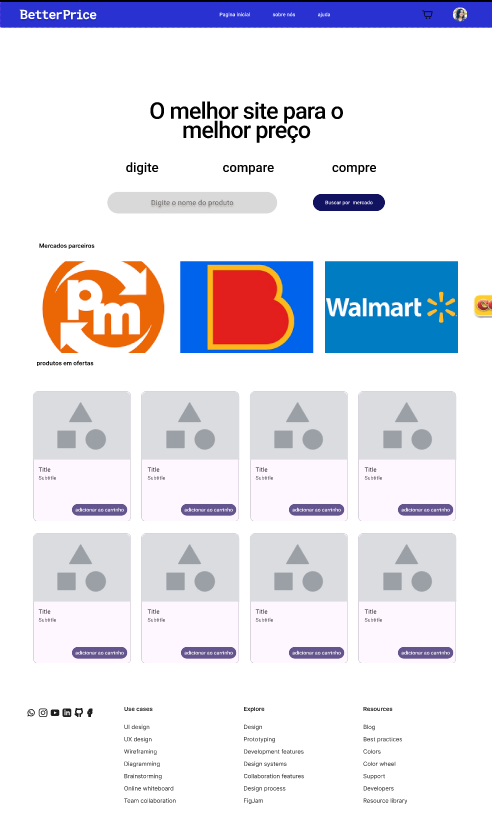


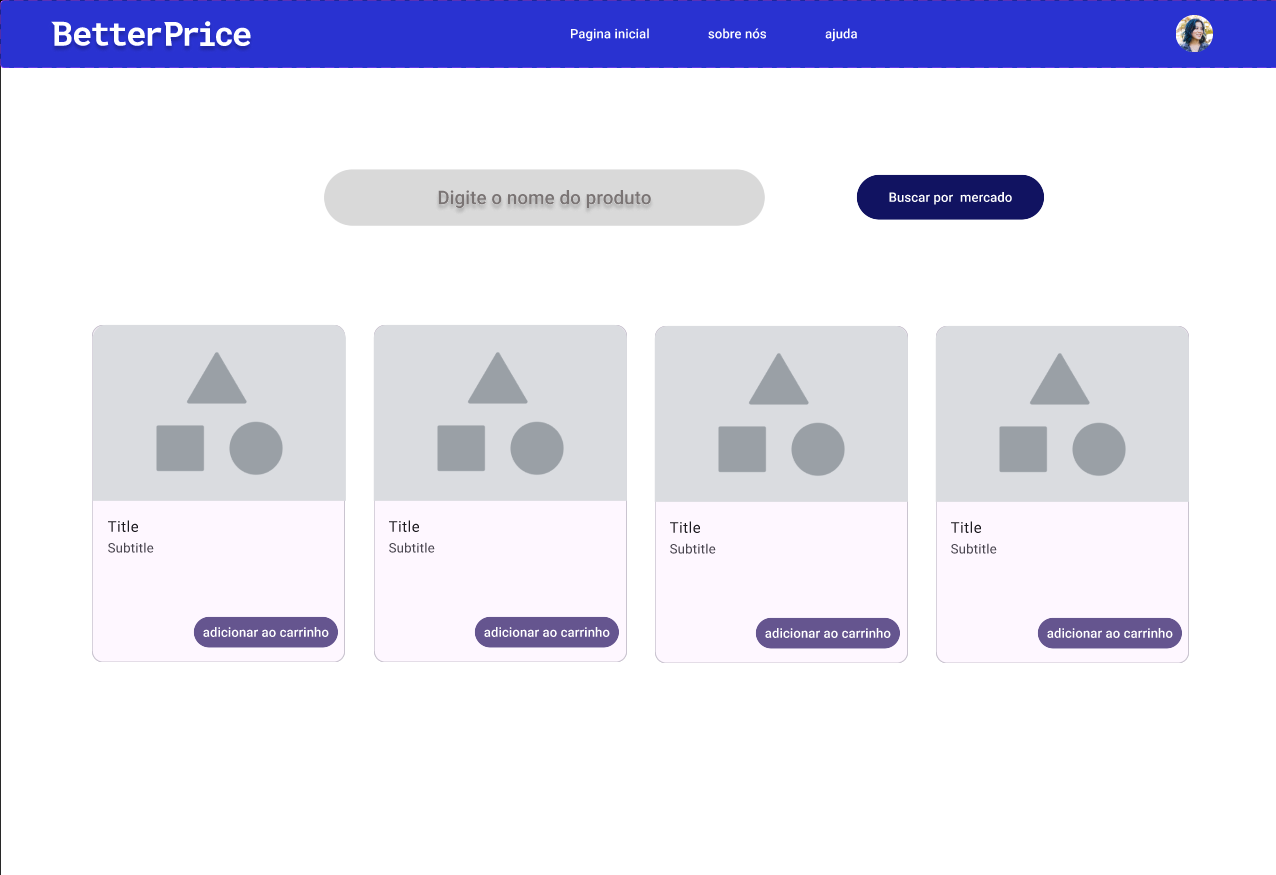
## 5.2.4 4º Sprint:



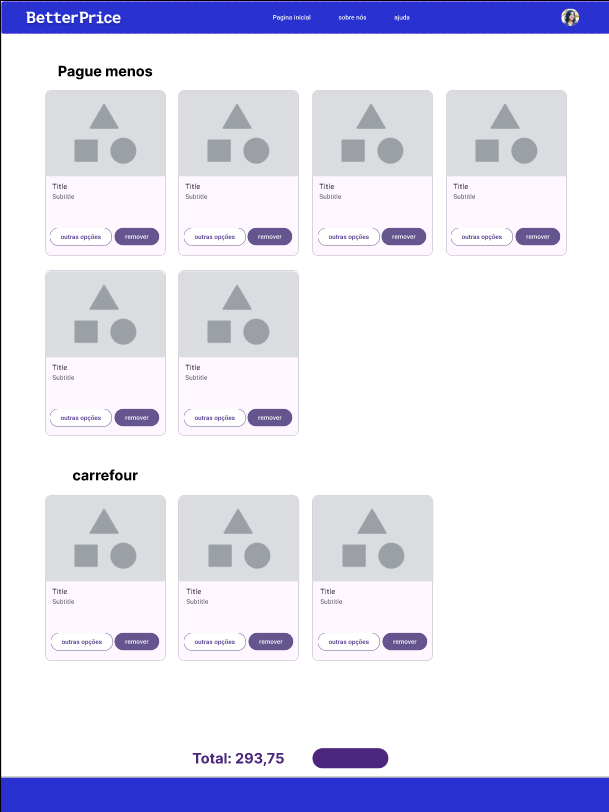
# PROTOTIPAGEM

# 6.1 Tela inicial:



**6.2 Página de pesquisa:** 

**6.3 Carrinho:**



# DIAGRAMAÇÃO DO PROJETO

# 7.1 Caso de uso:

Diagrama

Descrição gerada automaticamente

# 7.2 diagrama entidade-relacionamento:

