

Backlog



Epic

Insights

Configurações de visualização

Epic

Itens sem epic

Solicitação de Calculo

Resultado do Calculo

Login e Cadastro

+ Criar epic

▼ MVP Sprint 1

Adicionar dados (4 items)

0170

Iniciar sprint

...

BS1-1 Login

LOGIN E CADASTRO

EM ANDAMENTO

3

=

BS1-2 Pág. de Calculo

SOLICITAÇÃO DE CALC...

EM ANDAMENTO

5

=

BS1-3 Pág. de Resultado do Calculo

RESULTADO DO CALCULO

EM ANDAMENTO

5

=

BS1-8 Cadastro de Vendedor

LOGIN E CADASTRO

EM ANDAMENTO

4

=

+ Criar item

4 items

Estimativa: 17

▼ Backlog (6 items)

0470

Plano no quadro branco

AVALIAR

Criar sprint

BS1-10 Validações e Segurança

LOGIN E CADASTRO

EM ANDAMENTO

11

=

BS1-11 Funcionalidades principais

RESULTADO DO CALCULO

EM ANDAMENTO

8

=

BS1-12 Testes e melhorias contínuas

RESULTADO DO CALCULO

EM ANDAMENTO

7

=

BS1-13 Backlog Login

LOGIN E CADASTRO

EM ANDAMENTO

5

=

BS1-14 Backlog Calculo

SOLICITAÇÃO DE CALC...

EM ANDAMENTO

7

=

BS1-15 Backlog Resultado Calculo

RESULTADO DO CALCULO

EM ANDAMENTO

9

=

+ Criar item

Validações e Segurança

+ Adicionar

📱 Apps

Descrição

- **Descrição:**
As validações e a segurança são aspectos críticos para garantir que os dados do usuário sejam tratados corretamente e que o sistema seja seguro contra acessos não autorizados, entradas inválidas e vulnerabilidades de segurança.
- **Goal:**
Garantir que os dados inseridos pelos usuários sejam válidos e que o sistema seja seguro, protegendo informações sensíveis e prevenindo fraudes ou ataques.
- **Critério de Aceitação:**
 - **Dado que** o usuário insere um e-mail inválido, **quando** tenta realizar o login ou cadastro, **então** o sistema deve exibir uma mensagem de erro.
 - **Dado que** o usuário insere uma senha fraca, **quando** tenta realizar o cadastro, **então** o sistema deve alertar o usuário sobre a necessidade de uma senha mais forte.
 - **Dado que** o usuário está acessando a página de login, **quando** tenta acessar sem estar autenticado, **então** o sistema deve redirecioná-lo para a página de login.
 - **Dado que** o usuário insere dados no formulário de cadastro, **quando** os dados não forem válidos, **então** o sistema deve bloquear o envio e solicitar a correção.
- **Dependências:**
 - Sistema de autenticação e autorização.
 - Criptografia de dados (senhas).
 - Bibliotecas de validação de e-mail e campos obrigatórios.
- **Prioridade:**
Alta – Fundamental para a integridade e segurança do produto e dados dos usuários.

Definition of Ready (DoR)

A história ou tarefa está pronta para ser trabalhada quando:

1. Os critérios de teste (unitários, de integração, E2E) foram definidos.
2. O tipo de teste necessário (automático ou manual) foi especificado.
3. Ferramentas de teste automatizado foram configuradas ou estão disponíveis.
4. O código foi escrito com foco em testabilidade.
5. O escopo da melhoria contínua foi claramente identificado (ex.: otimização de performance, refatoração de código, revisão de fluxo).
6. A equipe tem uma visão clara de como medir o sucesso (ex.: performance, cobertura de testes, feedback dos usuários).
7. As melhorias são definidas de forma que não impactem negativamente o sistema.

Definition of Done (DoD)

A história ou tarefa será considerada **feita** quando:

1. Todos os testes necessários (unitários, integração, E2E) foram implementados e executados com sucesso.
2. O código passou em todas as revisões de qualidade e cobertura de testes.
3. A cobertura de testes foi aumentada, caso necessário, e não há falhas críticas nos testes.
4. O sistema foi testado e a performance foi validada (incluindo testes de carga e estresse, se necessário).
5. As melhorias identificadas (ex.: refatoração, otimização) foram implementadas e documentadas.
6. Todos os problemas e bugs encontrados durante os testes foram resolvidos.
7. O código está otimizado e não impacta negativamente o desempenho geral do sistema.
8. As ferramentas de monitoramento e performance estão configuradas para garantir a continuidade da melhoria.

Funcionalidades principais

+ Adicionar

@ Apps

Descrição

- **Descrição:**

As funcionalidades principais do sistema são as funcionalidades essenciais que o produto oferece ao usuário, como o cálculo de cubagem, o login e o cadastro de vendedores. Elas formam a base do produto e são fundamentais para que o usuário consiga realizar suas tarefas.

- **Goal:**

Garantir que as funcionalidades essenciais do sistema sejam implementadas corretamente, proporcionando ao usuário uma experiência completa e funcional.

- **Critério de Aceitação:**

- **Dado que** o usuário acessa o sistema, **quando** insere as dimensões do objeto e clica em "Calcular", **então** o cálculo de cubagem deve ser realizado corretamente.
- **Dado que** o usuário insere credenciais válidas na página de login, **quando** clica em "Entrar", **então** o sistema deve redirecioná-lo para a página principal.
- **Dado que** o administrador insere as informações de um vendedor no cadastro, **quando** clica em "Cadastrar", **então** o vendedor deve ser registrado corretamente.

- **Dependências:**

- Algoritmos de cálculo.
- Banco de dados para armazenamento de informações do usuário e vendedores.
- Sistema de autenticação e autorização.

- **Prioridade:**

Alta – Essencial para a operação do produto.

Definition of Ready (DoR)

A história ou tarefa está pronta para ser trabalhada quando:

1. Os critérios de teste (unitários, de integração, E2E) foram definidos.
2. O tipo de teste necessário (automático ou manual) foi especificado.
3. Ferramentas de teste automatizado foram configuradas ou estão disponíveis.
4. O código foi escrito com foco em testabilidade.
5. O escopo da melhoria contínua foi claramente identificado (ex.: otimização de performance, refatoração de código, revisão de fluxo).
6. A equipe tem uma visão clara de como medir o sucesso (ex.: performance, cobertura de testes, feedback dos usuários).
7. As melhorias são definidas de forma que não impactem negativamente o sistema.

Definition of Done (DoD)

A história ou tarefa será considerada **feita** quando:

1. Todos os testes necessários (unitários, integração, E2E) foram implementados e executados com sucesso.
2. O código passou em todas as revisões de qualidade e cobertura de testes.
3. A cobertura de testes foi aumentada, caso necessário, e não há falhas críticas nos testes.
4. O sistema foi testado e a performance foi validada (incluindo testes de carga e estresse, se necessário).
5. As melhorias identificadas (ex.: refatoração, otimização) foram implementadas e documentadas.
6. Todos os problemas e bugs encontrados durante os testes foram resolvidos.
7. O código está otimizado e não impacta negativamente o desempenho geral do sistema.
8. As ferramentas de monitoramento e performance estão configuradas para garantir a continuidade da melhoria.

Testes e melhorias contínuas

+ Adicionar

📱 Apps

Descrição

- **Descrição:**

Testes e melhorias contínuas se referem ao processo de garantir que o sistema está funcionando corretamente por meio de testes regulares e da implementação de melhorias ao longo do tempo, com base no feedback dos usuários e em dados de uso.

- **Goal:**

Assegurar que o produto seja estável, funcional e esteja sempre evoluindo de acordo com as necessidades dos usuários e as mudanças tecnológicas.

- **Critério de Aceitação:**

- **Dado que** a funcionalidade de login foi implementada, **quando** executamos testes automatizados, **então** o login deve ser bem-sucedido com entradas válidas e falhar com entradas inválidas.
- **Dado que** uma nova versão da plataforma foi lançada, **quando** o sistema for testado, **então** ele deve funcionar corretamente em todos os navegadores e dispositivos suportados.
- **Dado que** um bug foi identificado na calculadora, **quando** a correção for implementada, **então** o sistema deve passar nos testes de validação e garantir que o erro foi resolvido.

- **Dependências:**

- Ferramentas de testes automatizados (unitários, integração e E2E).
- Ferramentas de monitoramento de desempenho e logs.
- Processos de revisão e feedback contínuo de usuários.

- **Prioridade:**

Alta – Garantir qualidade e estabilidade contínuas do produto.

Definition of Ready (DoR)

A história ou tarefa está pronta para ser trabalhada quando:

1. Os critérios de teste (unitários, de integração, E2E) foram definidos.
2. O tipo de teste necessário (automático ou manual) foi especificado.
3. Ferramentas de teste automatizado foram configuradas ou estão disponíveis.
4. O código foi escrito com foco em testabilidade.
5. O escopo da melhoria contínua foi claramente identificado (ex.: otimização de performance, refatoração de código, revisão de fluxo).
6. A equipe tem uma visão clara de como medir o sucesso (ex.: performance, cobertura de testes, feedback dos usuários).
7. As melhorias são definidas de forma que não impactem negativamente o sistema.

Definition of Done (DoD)

A história ou tarefa será considerada **feita** quando:

1. Todos os testes necessários (unitários, integração, E2E) foram implementados e executados com sucesso.
2. O código passou em todas as revisões de qualidade e cobertura de testes.
3. A cobertura de testes foi aumentada, caso necessário, e não há falhas críticas nos testes.
4. O sistema foi testado e a performance foi validada (incluindo testes de carga e estresse, se necessário).
5. As melhorias identificadas (ex.: refatoração, otimização) foram implementadas e documentadas.
6. Todos os problemas e bugs encontrados durante os testes foram resolvidos.
7. O código está otimizado e não impacta negativamente o desempenho geral do sistema.
8. As ferramentas de monitoramento e performance estão configuradas para garantir a continuidade da melhoria.

Backlog Login

+ Adicionar

@ Apps

Descrição

- **Descrição:**
A página de login permite que os usuários acessem a plataforma da calculadora de cubagem inserindo suas credenciais (e-mail e senha). Também inclui funcionalidades como recuperação de senha e opção de manter o usuário logado.
- **Goal:**
Permitir que os usuários façam login de maneira segura e eficiente, com suporte a recuperação de senha e funcionalidade de "lembrar-me".
- **Critério de Aceitação:**
 - **Dado que** o usuário possui um cadastro válido, **quando** insere e-mail e senha corretos, **então** o sistema deve permitir o acesso à página principal.
 - **Dado que** o usuário insere e-mail ou senha incorretos, **quando** tenta fazer login, **então** o sistema deve exibir uma mensagem de erro.
 - **Dado que** o usuário não preenche os campos obrigatórios, **quando** tenta realizar o login, **então** o sistema deve alertar o usuário sobre os campos ausentes.
 - **Dado que** o usuário esqueceu a senha, **quando** clica em "Esqueci minha senha", **então** o sistema deve permitir a recuperação via e-mail.
- **Dependências:**
 - Banco de dados de usuários.
 - Sistema de envio de e-mails para recuperação de senha.
 - Configuração de segurança para criptografia de senhas.
- **Prioridade:**
Alta – Essencial para o acesso ao sistema.

Definition of Ready (DoR)

A história ou tarefa está pronta para ser trabalhada quando:

1. Os critérios de teste (unitários, de integração, E2E) foram definidos.
2. O tipo de teste necessário (automático ou manual) foi especificado.
3. Ferramentas de teste automatizado foram configuradas ou estão disponíveis.
4. O código foi escrito com foco em testabilidade.
5. O escopo da melhoria contínua foi claramente identificado (ex.: otimização de performance, refatoração de código, revisão de fluxo).
6. A equipe tem uma visão clara de como medir o sucesso (ex.: performance, cobertura de testes, feedback dos usuários).
7. As melhorias são definidas de forma que não impactem negativamente o sistema.

Definition of Done (DoD)

A história ou tarefa será considerada **feita** quando:

1. Todos os testes necessários (unitários, integração, E2E) foram implementados e executados com sucesso.
2. O código passou em todas as revisões de qualidade e cobertura de testes.
3. A cobertura de testes foi aumentada, caso necessário, e não há falhas críticas nos testes.
4. O sistema foi testado e a performance foi validada (incluindo testes de carga e estresse, se necessário).
5. As melhorias identificadas (ex.: refatoração, otimização) foram implementadas e documentadas.
6. Todos os problemas e bugs encontrados durante os testes foram resolvidos.
7. O código está otimizado e não impacta negativamente o desempenho geral do sistema.
8. As ferramentas de monitoramento e performance estão configuradas para garantir a continuidade da melhoria.

Backlog Calculo

+ Adicionar

@ Apps

Descrição

- **Descrição:**

A página de cálculo permite ao usuário inserir as dimensões de um objeto e calcular o volume de cubagem. O cálculo considera diferentes unidades de medida e exibe o resultado de forma clara e imediata.

- **Goal:**

Permitir que os usuários calculem o volume de cubagem de objetos com facilidade e precisão, oferecendo suporte para unidades de medida.

- **Critério de Aceitação:**

- **Dado que** o usuário insere valores para altura, largura e profundidade, **quando** clica em "Calcular", **então** o sistema deve calcular corretamente o volume.
- **Dado que** o usuário insere valores inválidos (não numéricos), **quando** tenta calcular, **então** o sistema deve alertar sobre o erro.
- **Dado que** o usuário deseja usar unidades diferentes (cm³, m³, polegadas³), **quando** seleciona a unidade, **então** o cálculo deve ser feito de acordo com a unidade escolhida.

- **Dependências:**

- Algoritmo de cálculo de cubagem.
- Base de dados de unidades de medida.
- Validação de entradas (ex.: valores numéricos).

- **Prioridade:**

Alta – Funcionalidade central da aplicação.

Definition of Ready (DoR)

A história ou tarefa está pronta para ser trabalhada quando:

1. Os critérios de teste (unitários, de integração, E2E) foram definidos.
2. O tipo de teste necessário (automático ou manual) foi especificado.
3. Ferramentas de teste automatizado foram configuradas ou estão disponíveis.
4. O código foi escrito com foco em testabilidade.
5. O escopo da melhoria contínua foi claramente identificado (ex.: otimização de performance, refatoração de código, revisão de fluxo).
6. A equipe tem uma visão clara de como medir o sucesso (ex.: performance, cobertura de testes, feedback dos usuários).
7. As melhorias são definidas de forma que não impactem negativamente o sistema.

Definition of Done (DoD)

A história ou tarefa será considerada **feita** quando:

1. Todos os testes necessários (unitários, integração, E2E) foram implementados e executados com sucesso.
2. O código passou em todas as revisões de qualidade e cobertura de testes.
3. A cobertura de testes foi aumentada, caso necessário, e não há falhas críticas nos testes.
4. O sistema foi testado e a performance foi validada (incluindo testes de carga e estresse, se necessário).
5. As melhorias identificadas (ex.: refatoração, otimização) foram implementadas e documentadas.
6. Todos os problemas e bugs encontrados durante os testes foram resolvidos.
7. O código está otimizado e não impacta negativamente o desempenho geral do sistema.
8. As ferramentas de monitoramento e performance estão configuradas para garantir a continuidade da melhoria.

Backlog Resultado Calculo

+ Adicionar

@ Apps

Descrição

- **Descrição:**
Após calcular o volume de cubagem, o usuário será redirecionado para a página de resultados, onde poderá visualizar o volume calculado, compartilhar o resultado e salvar no histórico.
- **Goal:**
Exibir o resultado do cálculo de forma clara e permitir que o usuário compartilhe ou salve a informação para futuras consultas.
- **Criterio de Aceitação:**
 - **Dado que** o cálculo foi realizado, **quando** o usuário acessa a página de resultados, **então** o volume deve ser exibido corretamente com a unidade de medida.
 - **Dado que** o usuário deseja compartilhar o resultado, **quando** clica em "Compartilhar", **então** o sistema deve enviar o resultado por e-mail.
 - **Dado que** o usuário deseja salvar o resultado, **quando** clica em "Salvar", **então** o resultado deve ser armazenado no histórico do usuário.
- **Dependências:**
 - Banco de dados para armazenar histórico de cálculos.
 - Sistema de envio de e-mails.
- **Prioridade:**
Média – Complementa a experiência do usuário, mas a funcionalidade de cálculo é mais crítica.

Definition of Ready (DoR)

A história ou tarefa está pronta para ser trabalhada quando:

1. Os critérios de teste (unitários, de integração, E2E) foram definidos.
2. O tipo de teste necessário (automático ou manual) foi especificado.
3. Ferramentas de teste automatizado foram configuradas ou estão disponíveis.
4. O código foi escrito com foco em testabilidade.
5. O escopo da melhoria contínua foi claramente identificado (ex.: otimização de performance, refatoração de código, revisão de fluxo).
6. A equipe tem uma visão clara de como medir o sucesso (ex.: performance, cobertura de testes, feedback dos usuários).
7. As melhorias são definidas de forma que não impactem negativamente o sistema.

Definition of Done (DoD)

A história ou tarefa será considerada **feita** quando:

1. Todos os testes necessários (unitários, integração, E2E) foram implementados e executados com sucesso.
2. O código passou em todas as revisões de qualidade e cobertura de testes.
3. A cobertura de testes foi aumentada, caso necessário, e não há falhas críticas nos testes.
4. O sistema foi testado e a performance foi validada (incluindo testes de carga e estresse, se necessário).
5. As melhorias identificadas (ex.: refatoração, otimização) foram implementadas e documentadas.
6. Todos os problemas e bugs encontrados durante os testes foram resolvidos.
7. O código está otimizado e não impacta negativamente o desempenho geral do sistema.
8. As ferramentas de monitoramento e performance estão configuradas para garantir a continuidade da melhoria.

Linha do tempo

Enviar feedback Compartilhar Exportar

EN

Categoria do sta...

Epic

	OV	DEZ	JAN '25	
Sprints				
<input type="checkbox"/> > BS1-5 Solicitação de Calculo				
<input type="checkbox"/> > BS1-6 Resultado do Calculo				
<input type="checkbox"/> > BS1-7 Login e Cadastro				
+ Criar Epic				