

Test Plan TopBids

1 Introdução

O projeto TopBids é uma aplicação de leilão online que visa melhorar a eficiência, segurança e comodidade tanto para os leiloeiros quanto para os participantes. Este plano de teste abrange todas as funcionalidades descritas no projeto, com foco na garantia da qualidade e na identificação de possíveis problemas antes do lançamento.

2 Arquitetura

A aplicação será desenvolvida utilizando Django para o backend, React Native para o desenvolvimento mobile, Bootstrap + CSS para o frontend web, hospedada na AWS, com utilização de RMQ (RabbitMQ) para mensageria e Docker para contêineres. Isso garante uma arquitetura robusta e escalável, com uma combinação de tecnologias consolidadas e de alto desempenho.

3 Funcionalidades

Funcionalidade	Cadastro de Leiloeiro pela Web
Comportamento Esperado	Ao acessar a página de cadastro, o Leiloeiro preenche os campos obrigatórios com suas informações pessoais e de pix
Verificações	Senha min 8 caracteres e no máximo 18 Todos os campos devem ser obrigatórios. Redirecionar o usuário para landpage logado. Exibir a mensagem de falha em caso de usuário existente. Exibir mensagem de falha em caso de confirmação de senha não ser igual Exibir mensagem de falha no caso de campo obrigatório incompleto.
CrITÉrios de Aceite	Senha de 8 caracteres, no mínimo. Todos os campos devem ser obrigatórios.

Funcionalidade	Cadastro de Apostador no Mobile
----------------	---------------------------------

Comportamento Esperado	O Apostador acessa a aplicação mobile e preenche os campos necessários para criar sua conta
Verificações	Senha min 8 caracteres e no máximo 18 Todos os campos devem ser obrigatórios. Redirecionar o usuário para landpage logado. Exibir a mensagem de falha em caso de usuário existente. Exibir mensagem de falha em caso de confirmação de senha não ser igual Exibir mensagem de falha no caso de campo obrigatório incompleto.
Critérios de Aceite	Senha de 8 caracteres, no mínimo. Todos os campos devem ser obrigatórios.

Funcionalidade	Login Web do Leiloeiro
Comportamento Esperado	O Leiloeiro insere suas credenciais nos campos de login
Verificações	Os campos de usuário e senha estão presentes e preenchidos corretamente. O sistema verifica se as credenciais são válidas.
Critérios de Aceite	Os campos de usuário e senha estão preenchidos. As credenciais são válidas e permitem o acesso ao sistema.

Funcionalidade	Login Mobile de Apostador
Comportamento Esperado	O Apostador insere suas credenciais nos campos de login no aplicativo

Verificações	Os campos de usuário e senha estão presentes e preenchidos corretamente. O sistema verifica se as credenciais são válidas.
Critérios de Aceite	Os campos de usuário e senha estão preenchidos. As credenciais são válidas e permitem o acesso ao aplicativo.

Funcionalidade	Cadastro de Leilão
Comportamento Esperado	O Leiloeiro preenche os campos obrigatórios no formulário de cadastro de leilão
Verificações	Todos os campos obrigatórios estão presentes e preenchidos corretamente. O sistema valida os dados inseridos, incluindo data, horário e valor mínimo do lance.
Critérios de Aceite	Todos os campos obrigatórios estão preenchidos corretamente. Os dados são válidos e atendem aos critérios estabelecidos.

Funcionalidade	Lista de Leilões
Comportamento Esperado	O Apostador visualiza a lista de leilões disponíveis
Verificações	A lista de leilões é exibida corretamente na interface. Cada item na lista contém as informações relevantes, como nome, descrição e valor mínimo do lance.
Critérios de Aceite	A lista de leilões é exibida corretamente. As informações de cada leilão estão completas e corretas.

Funcionalidade	Buscar Leilão
----------------	---------------

Comportamento Esperado	O Apostador utiliza a função de busca para encontrar um leilão específico
Verificações	A busca retorna resultados relevantes com base no termo inserido. Os resultados são apresentados de forma clara e organizada.
CrITÉrios de Aceite	A busca retorna resultados relevantes. Os resultados são apresentados de forma clara e organizada.

Funcionalidade	Iniciar Leilão
Comportamento Esperado	O Leiloeiro inicia o leilão para que os Apostadores possam começar a fazer lances
Verificações	O sistema permite ao Leiloeiro definir a data e o horário de início do leilão. A interface do Leiloeiro exibe claramente as opções para iniciar o leilão.
CrITÉrios de Aceite	O leilão é iniciado conforme as informações fornecidas pelo Leiloeiro.

4 Estratégia de Teste

• Escopo de Testes

O escopo de testes abrange todas as funcionalidades descritas no projeto TopBids. Isso inclui desde o cadastro de usuários até a pesquisa de itens nos leilões, passando pelo login, gerenciamento de perfil, navegação na lista de leilões e todas as outras interações previstas na aplicação. O plano de testes exclui a funcionalidade de cadastro de itens nos leilões, uma vez que não está contemplada no escopo deste plano.

Testes Unitários:

- Os desenvolvedores serão responsáveis por escrever testes unitários que cubram, no mínimo, 60% do código.
- Os testes unitários visam garantir que cada componente individual da aplicação funcione conforme o esperado e que mudanças de código não causem regressões inesperadas.

Testes de Integração:

- Serão realizados testes de integração em todos os endpoints da aplicação.
- Esses testes serão executados para garantir que os diferentes componentes da aplicação se comuniquem corretamente entre si e que os dados sejam transmitidos de forma consistente.

Testes Automatizados:

- Serão realizados testes automatizados, especialmente testes end-to-end na funcionalidade de login.
- Os testes automatizados garantem que as principais funcionalidades da aplicação sejam testadas de forma automatizada, agilizando o processo de testes e permitindo uma maior cobertura de casos de uso.

Testes Manuais:

- Todas as funcionalidades serão testadas manualmente pelo time de qualidade.
- Os testes manuais serão conduzidos seguindo a documentação de cenários de teste e deste TestPlan, garantindo que a aplicação se comporte conforme o esperado em diferentes cenários de uso.

• **Ambiente e Ferramentas**

Exemplo: Os testes serão feitos do ambiente de homologação, e contém as mesmas configurações do ambiente de produção com uma massa de dados gerada previamente pelo time de qualidade.

As seguintes ferramentas serão utilizadas no teste:

Ferramenta	Time	Descrição
Curl	Qualidade	Ferramenta para realização de testes de API
Django.Test	Desenvolvimento	Framework utilizada para testes unitários
Cypress	Qualidade	Ferramenta para testes end-to-end
Lighthouse	Desenvolvimento	Avaliação de performance e acessibilidade da aplicação
Gravador de Passos	Desenvolvimento	Prover evidências dos testes

5 Classificação de Bugs

Durante os testes, os bugs serão classificados em diferentes níveis de acordo com sua gravidade, facilitando a priorização e resolução por parte da equipe de desenvolvimento. A classificação de bugs será realizada nos seguintes níveis:

ID	Nível de Severidade	Descrição
1	Crítico	<ul style="list-style-type: none"> • Este é o nível mais alto de gravidade. • Bugs críticos são aqueles que impedem o funcionamento adequado de uma função ou recurso essencial da aplicação, podendo até mesmo causar falha total na aplicação. • Exemplos incluem funcionalidades principais que não funcionam corretamente, crashes na aplicação ou falhas de segurança graves.
2	Alto	<ul style="list-style-type: none"> • Este nível de gravidade indica problemas graves que afetam diretamente a experiência do usuário ou comprometem funcionalidades importantes da aplicação. • São bugs que podem causar impacto significativo na utilização da aplicação, mas ainda permitem que a maioria das funcionalidades seja utilizada. • Exemplos incluem erros que impedem a realização de ações importantes para o usuário, inconsistências nos dados exibidos ou falhas de segurança moderadas.
3	Médio	<ul style="list-style-type: none"> • Bugs de gravidade média são aqueles que afetam funcionalidades menos críticas da aplicação ou causam problemas que podem ser contornados pelo usuário. • Embora não sejam tão graves quanto os bugs críticos ou de alto impacto, ainda representam falhas que precisam ser corrigidas para garantir uma experiência consistente e satisfatória para o usuário. • Exemplos incluem erros de validação de entrada, problemas de formatação ou pequenas inconsistências na interface do usuário.
4	Baixo	<ul style="list-style-type: none"> • Este é o nível de menor gravidade. • Bugs de baixa gravidade são aqueles que têm pouco impacto na funcionalidade ou usabilidade da aplicação, sendo geralmente problemas estéticos ou pequenas falhas que não afetam significativamente a experiência do usuário. • Exemplos incluem erros de ortografia, ajustes de layout ou outras questões menores que não prejudicam o uso da aplicação.

6 Definição de Pronto

Para uma funcionalidade ser considerada pronta, é necessário que ela esteja completa, ou seja, implementada de acordo com os requisitos especificados e atendendo às expectativas de negócio. Além disso, todos os testes devem ter sido realizados, garantindo que a funcionalidade esteja livre de bugs e regressões.

O código relacionado à funcionalidade também deve passar por uma revisão, seja por pares ou por um membro designado da equipe, para garantir sua qualidade e conformidade com os padrões de codificação estabelecidos. A documentação relacionada à funcionalidade deve ser atualizada para refletir quaisquer mudanças ou novas implementações.

Outros aspectos importantes incluem a verificação da performance da funcionalidade, sua compatibilidade em diferentes ambientes e dispositivos, e a aplicação de medidas de segurança adequadas para proteger contra vulnerabilidades e garantir a integridade dos dados do usuário. Por fim, a aprovação do stakeholder ou responsável pelo produto é necessária para garantir que a funcionalidade atenda às expectativas e necessidades do cliente.

Ao cumprir todos esses critérios, a funcionalidade pode ser considerada pronta para a entrega final.