

Plano de testes do ServeRest

Plano de testes do e-commerce ServeRest.

Objetivo

- Garantir que a funcionalidade **Login** da aplicação ServeRest funcione conforme o esperado. Identificando falhas, validando regras de negócio e usabilidade.
- A tela **cadastro** seguirá os mesmos requisitos.

Resumo

O Servelet é um E-cormmece genérico com os usuário de administrador, administrador do sistema, e usuário comum, usuário-alvo da plataforma. Esse planejamento de teste será realizado por conta de problemas de ferramentas do sistema. O planejamento de testes terão certas pessoas envolvidas, além do escopo explicado dos testes, estratégias, critérios, cronograma, recursos, ambiente de testes e possíveis riscos.

A hipótese deste teste é que as funcionalidades de Autenticação e Cadastro da conta dos tipos de usuários do sistema (administrador e usuário comum) estejam em ótimas condições, envolvendo a segurança e os protocolos necessários.

Pessoas Envolvidas:

Neste planejamento teve apenas um total de 4 testadores envolvidos. Os testadores envolvidos são os seguintes

- Emanuel Avelino
- Kauã
- Raphael Rates
- Izadora

Funcionalidades ou Módulos a serem testados

Será testada a página de login do ServeRest.

Local dos Testes

Os testes serão realizados no ServRest, de forma home office através do próprio computador de cada testador em ambiente de produção.

Recursos necessários

Será necessário que cada testador tenha um computador e internet disponíveis.

É necessário instalar alguns software com ou sem recursos financeiros para realizar a testagem, como navegadores, editores de código (caso tivesse acesso ao código da aplicação), softwares de testes de segurança e debug.

Escopo

° Login:

1. Login com dados **válidos**, E-mail já cadastrado e senha.
2. Login com **senha e e-mail inválidos**, verificando se há necessidade de renderizar uma mensagem de erro.
3. Campos que são **obrigatórios** em branco.
4. Mensagens sendo mostradas ao **usuário quando login feito ou inválido**.
5. **Redirecionamento** após o login.

° Cadastro:

1. Cadastro com dados válidos (**nome, e-mail e senha**).
2. **Verificação de e-mail** se já está sendo utilizado ou não e **se está conforme o padrão** de e-mail.
3. Senha obrigatória de no **mínimo 6 caracteres**.
4. Confirmação visual, ou seja, **mensagem de cadastro realizado com sucesso ou não**.
5. Redirecionamento após **cadastro para a tela /login**.

Fora do escopo:

- Recuperação de senha.
- Exclusão de conta.

Critérios de Teste

- Login vai estar válido quando o usuário já estiver cadastrado e quando a senha estiver dentro do que a plataforma solicita, sem nenhum campo em branco e com os mínimos requisitos de segurança (8 caracteres, uma letra maiúscula, u minúscula e números;
- Sempre que o usuário inserir uma senha inválida ou e-mail deve receber uma mensagem de erro;
- Sempre que o usuário realizar seu cadastro de forma satisfatória, deve receber uma mensagem de confirmação de “Cadastro Realizado com sucesso”;
- Após o cadastro, o usuário deve ser redirecionado a página de login;
- O desempenho da aplicação precisa ser aceitável (tempos de resposta e carregamento dentro de um limite razoável).

- O sistema não apresentar problemas de **segurança** evidentes

Riscos

Nesse tópico serão listados os possíveis riscos para o planejamento de testes da aplicação.

- Falta de conexão com a internet via rede wi-fi ou cabo. Caso ocorra este evento, será necessário a conexão com o wi-fi via celular por roteamento 4G.
- Ausência da qualidade dos softwares de testes caso tenha alguma atualização imprevista que cause impacto.

Tipo de testes a serem feitos:

- Teste funcional.
- Teste de interface (UI).
- Teste de usabilidade.
- Teste de segurança.
- Testes de Compatibilidade
- Testes de Performance

Estratégias de teste:

• **Teste funcional:**

- Terá como objetivo verificar se o sistema está fazendo aquilo que deveria estar fazendo, com os resultados que já temos, o nosso esperado.

1. **Login** com credenciais **válidas ou inválidas**.

2. **Cadastro** com dados válidos ou inválidos.

3. Verificar **mensagens** de erro.

4. **Redirecionamento** de telas.

• *Ferramentas e ambientes a serem usados:*

- Será por meio manual, ou seja, através do navegador como um usuário.

• Teste de Interface(UI):

- Terá como objetivo verificar se os elementos de interface estejam corretamente visíveis.

1. Botões **responsivos**.

2. Mensagens **visíveis**.

3. **Responsividade** (mobile e desk).

4. **Validação** de renderização de componentes no Front-end.

5. Verificar o **Autocomplete** dos Inputs do formulário de Login e cadastro.

- *Ferramentas e ambientes a seres usados:*

- Será usado automação Cypress para verificar o JS.
- E verificar manualmente a interação que o usuário tem.

- Teste de Usabilidade:

- Se a experiência está fluída ou não.

1. Verificar se o usuário **entende com compreensão as mensagens exibidas**.

2. Verificar se causa **alguma confusão ou não**.

- *Ferramentas e ambientes a seres usados:*

- Teste manual de experiência visual nas mensagens e feedback de usuários.

- Teste de Segurança:

- Verificar se tem o mínimo de segurança aceito.

1. Verificar se o front consegue **esconder dados sensíveis**.

2. Verificar se um usuário **consegue se tornar admin**.

3. Verificar se está **cumprindo a LGPD**.

- *Ferramentas e ambientes a seres usados:*

- Automação com Postman e BurpSuite para verificar como está sendo a transferência de dados.

- Teste de Compatibilidade:

- Verificar como que o sistema se comporta em outros navegadores.

1. Verificar como o sistema **se comporta no navegador desk e para mobile**.

2. Testar vários navegadores **diferentes**.

- *Ferramentas e ambientes a seres usados:*

- Verificar em outros navegadores, como: Chrome, Edge, Brave, Safari, Firefox.

- Teste de Performance:

- Avaliar tempo de resposta do sistema e a sua fluidez utilizando o web Console para verificar o retorno da página em conexões via wi-fi, 4G e 3G.

1. Tempo de **carregamento ao clicar** para “Entrar” e para “Cadastrar”.

2. Verificar quanto tempo leva para o usuário **chegar até a tela principal do e-commerce**.
 3. Tempo de resposta de **feedback** ao digitar e clicar.
- *Ferramentas e ambientes a seres usados:*
 - Usar o Devtools para ajudar e BurpSuite.

Como Serão divulgados os resultados dos testes

Serão divulgados através de apresentação para as Squads.

Tipos de Priorização de Testes

Uma forma de caracterizar os tipos de testes que devem ser realizados.

- Fácil
- Médio (Relevante)
- Difícil (Complicada)
- Muito difícil (Urgência)

Cronograma

Horário	Teste	Módulo
13:30 - 14:15	Testes de Login (válido, inválido, campos vazios e mensagens)	Login
14:15 - 15:00	Testes de Cadastro (válido, e-mail duplicado/inválido e senha menor que 6)	Cadastro
15:00 - 15:15	Confirmação visual (mensagens de sucesso) e redirecionamentos	Login e Cadastro
15:15 - 15:30	Finalização	Conclusão