**Plano de teste**

Programa de Bolsas Compass - UOL

**API - Serverest**

João Ricardo Soares da Silva

Plano de teste

**Introdução**

Este documento apresenta o planejamento das atividades de testes da API Serverest, o qual será utilizado como base para as atividades de acompanhamento, revisão, verificação e validação do projeto, desde seu início até sua conclusão, a fim de garantir a análise comparativa do resultado real *versus* planejado. Desta forma, ações corretivas e preventivas poderão ser tomadas, sempre que resultados reais desviarem significativamente do planejado.

**Objetivos**

Os objetivos da API Serverest é que o usuário realize seu cadastro e conclua uma compra. A API irá realizar o cadastro do usuário, depois o usuário terá que realizar o seu login, então será possível a realização do cadastro de um produto que será adicionado a um carrinho e o será cancelado para efetivação de conclusão da compra.

**Escopo**

O escopo visa determinar os fluxos que compõem, ou, que constituirá a aplicação caso ela  esteja em processo de desenvolvimento. No caso da Uber o escopo é o seguinte:

~~CADASTRO/LOGIN~~

* ~~O usuário será autenticado no aplicativo através da confirmação dos seus dados por meio do Cadastro, quando for o primeiro acesso, ou quando já cadastrado, o Login na plataforma.~~

~~CORRIDA~~

* ~~Usuário especificará os locais de embarque (coleta do usuário) e destino (entrega do usuário).~~

~~CÁLCULO DE DISTÂNCIA E VALOR~~

* ~~Estimativa de tempo e preço da corrida.~~

~~START NA CORRIDA~~

* ~~Solicitação do motorista.~~

~~CANCELAMENTO DA CORRIDA~~

* ~~Usuário deve conseguir cancelar a corrida.~~

~~MOTORISTA~~

* ~~A Uber, unirá a solicitação do usuário a um Uber (motorista) próximo.~~
* ~~O motorista dará start na corri~~da,  buscando o cliente no local de embarque, e o levando até o seu destino.

**Mapa Mental**

**Suítes de Testes**

 Login:

Validar login e senha corretos.

Validar com login correto e senha incorreta.

Validar com login incorreto e senha correta.

 **Usuários**

Validar com email válido.

Validar com email inválido.

 **Produtos**

Validar forma de pagamento por Dinheiro

Validar forma de pagamento Cartão

Validar forma de pagamento por Pix

Validar Forma de pagamento Via Paypal

Validar forma de pagamento via Débito em conta

Validar forma de pagamento Negada

 **Carrinhos**

Validar endereço correto

Validar com endereço incorreto

**Estratégia de testes**

**Teste de Sistema**

Os testes de sistema avaliarão o funcionamento e o desempenho do sistema como um todo, verificando a eficácia e segurança, além de compatibilidade e integração do software em diferentes ambientes.

|  |  |
| --- | --- |
| Objetivo da Técnica: | Executar o sistema sob ponto de vista de seu usuário final, varrendo as funcionalidades em busca de falhas. |
| Técnica: | Os testes serão executados em condições similares de ambiente, interfaces sistêmicas e massa de dados, ou seja; aquela que um usuário utilizará no seu dia-a-dia de manipulação do sistema. De acordo com a politica de uma organização, podem ser utilizadas condições reais de ambiente, interfaces sistêmicas e massa de dados. |
| Ferramentas Necessárias: | Qualquer aparelho celular com sistema operacional Android ou IOS. |
| Critérios de Êxito: | O sistema se comportar como esperado (de acordo com o documento do requisitor). |
| Considerações Especiais: | Não se aplica. |

**Teste de Aceitação**

Os testes de aceitação apresentarão o produto Final para o usuário para validação e últimos ajustes.

|  |  |
| --- | --- |
| Objetivo da Técnica: | Validar o software pelo comprador, pelo usuário e por terceira parte, com o uso de dados ou cenários especificados ou reais. |
| Técnica: | O comprador (cliente)/Usuário utiliza o sistema como forma de validar se o sistema está de acordo com o requisitado. |
| Ferramentas Necessárias: | Qualquer aparelho celular com sistema operacional Android ou IOS. |
| Critérios de Êxito: | O cliente aceitar o sistema como foi implementado. |
| Considerações Especiais: | Não se aplica. |

**Teste de interface do usuário**

O teste de Interface do usuário (UI) verifica a interação do usuário com o software. O teste de UI tem por fim assegurar que a UI forneça ao usuário o acesso e a navegação adequados através das funções do objetivo do teste.

|  |  |
| --- | --- |
| Objetivo da Técnica: | Verificar se a interface da navegação reflete os requisitos e funções de negócios, incluindo a navegação entre os menus disponíveis campo a campo, e o uso de métodos de acesso. |
| Técnica: | Acesse cada tela da aplicação, para verificar se a navegação está de acordo com o esperado. |
| Ferramentas Necessárias: | Qualquer aparelho celular com sistema operacional Android ou IOS. |
| Critérios de Êxito: | As janelas estarem de acordo com as especificações. |
| Considerações Especiais: | Não se aplica. |

**Responsabilidades**

|  |  |
| --- | --- |
| **Papel** | **Responsabilidades** |
| Analista de Testes | Identifica e define os testes específicos a serem conduzidos.  Estas são as responsabilidades:   * Identificar ideias de teste * Definir detalhes dos testes * Determinar os resultados dos testes * Documentar solicitações de mudança |
| Testador | Implementa e executa os testes.  Estas são as responsabilidades:   * Implementar os teste e os conjuntos de testes * Executar os conjuntos de testes * Registrar os resultados * Analisar as falhas dos testes e possibilitar a recuperação posterior documentar incidentes. |

**Prioridades**

Considera-se prioridade todo e qualquer fluxo que na sua ausência, tornaria o objetivo do produto incapaz de ser atingido. Logo, os fluxos prioritários que tornaria o usuário incapaz de realizar a sua corrida é o cadastro e o login. Sem ele, o usuário não alcançaria o objetivo do aplicativo.

**Candidatos à automação**

 **Regressão**

**Validação 1**

Validar login

Validar endereço origem

Validar destino

Validar forma de pagamento com cartão

Validar aceitação do motorista.

Validar cancelamento da corrida pelo cliente

**Validação 2**

Validar acesso pelas redes sociais

Validar endereço

Validar destino

Validar distancia

Validar forma de pagamento com Pix

Validar motorista cadastrado

Validar cancelamento da corrida pelo motorista

**Ferramentas**

* Jira - Gerenciamento do projeto
* Swagger - Consulta da documentação
* Postman - Teste e consumo da API
* Dispositivos Mobile - Com sistemas operacionais Android e IOS.

**Cronograma**

|  |  |
| --- | --- |
| **Data** | **Tarefa** |
| 15/08/2022 | Criação dos testes |
| 22/08/2022 | Execução dos testes |
| 29/08/2022 | Testes de sistema |
| 05/09/2022 | Testes de aceitação e Interface (incluindo bugs) |
| 15/09/2022 | Ambiente Produção (Go Live) |