Uber

Planejamento de Testes

Por: Maria Gabriella Silva Menezes

Versão 1.0

2024

1. Introdução:

Este plano de testes visa garantir que os principais fluxos de solicitação de corrida e funcionalidades associadas no aplicativo Uber sejam testados com sucesso.

O foco será nos requisitos funcionais, não funcionais e regras de negócios descritos na Especificação Funcional para garantir que o sistema atenda às expectativas dos usuários, mantenha a qualidade e a segurança exigidas, e esteja conforme os requisitos de desempenho.

1. Escopo de Testes:

Os testes cobrirão o processo de solicitação de corrida, desde a seleção do ponto de partida e destino até a conclusão da corrida, com ênfase na funcionalidade do cálculo de preço, métodos de pagamento, comunicação entre motorista e passageiro, avaliações e compatibilidade do aplicativo.

1. Funcionalidades a Serem Testadas:

3.1 - Solicitação de Corrida:

* Testar se o usuário pode selecionar corretamente o ponto de partida e destino.
* Verificar o cálculo do preço estimado baseado na distância entre ponto de partida e destino.
* Garantir que o usuário possa selecionar e alterar o método de pagamento.
* Verificar se o motorista é notificado sobre a corrida e pode aceitar ou recusar.

3.2 - Interações:

* Testar a exibição do status da corrida em tempo real, garantindo a atualização correta (aceita, em rota, concluída).
* Verificar a funcionalidade do chat entre passageiro e motorista após a corrida ser aceita.
* Testar o processo de avaliação do motorista pelo passageiro e vice-versa após a corrida.

3.3 - Pagamentos:

* Testar a funcionalidade de pagamento, incluindo verificação de saldo e autorização do método de pagamento.
* Garantir que o usuário possa oferecer gorjeta após a corrida, se aplicável (cartão de crédito ou saldo no aplicativo).

3.4 - Cancelamento:

* Verificar a funcionalidade de cancelamento sem taxas dentro do limite de 5 minutos.
* Testar o cálculo e cobrança da taxa de cancelamento, se o motorista já tiver percorrido mais de 2 km.

1. Critérios de Aceitação:

* O usuário deve ser capaz de solicitar uma corrida e acompanhar seu status em tempo real.
* O motorista deve receber notificações adequadas e ter a opção de aceitar ou recusar a corrida.
* O cálculo do preço estimado deve ser preciso e rápido (máximo de 3 segundos).
* O método de pagamento deve ser validado antes da confirmação da corrida.
* O chat entre usuário e motorista deve funcionar corretamente após a corrida ser aceita.
* Avaliações de passageiro e motorista devem ser registradas corretamente após a corrida.
* O tempo de resposta para atualizações em tempo real não deve exceder 2 segundos.
* O sistema deve ser capaz de lidar com 100 mil solicitações simultâneas em picos de uso.
* O aplicativo deve funcionar conforme o esperado em diferentes dispositivos e versões de sistemas operacionais.

1. Casos de Teste

5.1 - Caso de Teste 1 - Solicitação de Corrida:

* Descrição: O usuário deve ser capaz de selecionar ponto de partida, destino e solicitar uma corrida.
* Pré-Requisitos: O usuário deve estar logado e com a localização do dispositivo ativada.
* Passos:

1. Abrir aplicativo e selecionar ponto de partida e destino.

* Resultado Esperado: O motorista é notificado e pode aceitar ou recusar a corrida.

5.2 - Caso de Teste 2: Cálculo do Preço Estimado

* Descrição: O preço estimado deve ser calculado de acordo com a distância entre o ponto de partida e o destino.
* Pré-Requisitos: Ponto de partida e destino definidos.
* Passos:

1. Inserir ponto de partida e destino.
2. Verificar o preço estimado exibido.

* Resultado Esperado: O preço exibido é calculado corretamente com base na distância.

5.3 - Caso de Teste 3 - Seleção do Método de Pagamento

* Descrição: Verificar se o usuário pode selecionar corretamente um método de pagamento disponível para a corrida.
* Pré-Requisitos: O usuário deve ter ao menos um método de pagamento registrado no aplicativo.
* Passos:

1. Solicitar uma corrida.
2. Selecionar um método de pagamento da lista (pix, cartão de crédito, débito ou crédito no aplicativo).
3. Confirmar o método de pagamento.

* Resultado Esperado: O método de pagamento é selecionado com sucesso, e a corrida é confirmada após a validação do método.

5.4 - Caso de Teste 4 - Atualização de Status da Corrida em Tempo Real

* Descrição: Verificar se o status da corrida é atualizado em tempo real (aceita, em rota, concluída) com latência máxima de 2 segundos.
* Pré-Requisitos: O motorista aceitou a corrida.
* Passos:

1. Solicitar uma corrida.
2. Aguardar o motorista aceitar a corrida.
3. Acompanhar o status da corrida (aceita, em rota, concluída).

* Resultado Esperado: O status da corrida é atualizado em tempo real, e a latência entre cada atualização não excede 2 segundos.

5.5 - Caso de Teste 5 - Avaliação do Motorista

* Descrição: Verificar se o usuário pode avaliar o motorista e oferecer uma gorjeta após a conclusão da corrida, se aplicável.
* Pré-Requisitos: A corrida foi realizada.
* Passos:

1. Concluir a corrida.
2. Avaliar o motorista atribuindo uma nota de 1 a 5 estrelas e adicionar um comentário (opcional).
3. Oferecer uma gorjeta ao motorista (opcional).

* Resultado Esperado: O usuário consegue avaliar o motorista com sucesso e, caso selecionado, a gorjeta é oferecida e debitada corretamente.

5.6 - Caso de Teste 6 - Suporte a Várias Solicitações Simultâneas

* Descrição: Verificar se o sistema é capaz de lidar com várias solicitações simultâneas de corrida durante os picos de uso.
* Pré-Requisitos: O sistema está em condições normais de operação, com vários usuários simultâneos simulados.
* Passos:

1. Simular uma sobrecarga de servidor utilizando JMeter.
2. Monitorar o desempenho do sistema, tempo de resposta e estabilidade durante o teste de carga.

* Resultado Esperado: O sistema deve ser capaz de processar todas as solicitações simultâneas sem falhas. O tempo de resposta para cada solicitação de corrida não deve ser afetado significativamente, e não deve haver perda de desempenho.

5.7 - Caso de Teste 7 - Verificação de Compatibilidade

* Descrição: Verificar se o aplicativo funciona corretamente em dispositivos com diferentes resoluções de tela e versões de sistemas operacionais Android e iOS.
* Pré-Requisitos: O aplicativo deve estar disponível para múltiplos dispositivos, com diferentes versões dos sistemas operacionais Android (versões mais recentes e as três anteriores) e iOS.
* Passos:

1. Instalar o aplicativo em dispositivos com diferentes resoluções de tela (smartphones e tablets).
2. Testar o aplicativo em dispositivos Android e iOS com versões mais recentes e nas três versões anteriores de ambos.
3. Realizar fluxos principais (solicitação de corrida, seleção de pagamento, visualização de status em tempo real, etc.) em cada dispositivo.

* Resultado Esperado: O aplicativo deve funcionar corretamente em todas as versões testadas e em dispositivos com diferentes resoluções. Não devem ocorrer problemas de layout, desempenho ou funcionalidade devido à compatibilidade de sistema operacional ou de dispositivo.

6. Ferramentas de Teste:

* Ferramentas de automação para testes de interface;
* JMeter para simulação de picos de uso;
* Ferramentas de segurança para validação de criptografias;
* Ferramentas de compatibilidade para teste em múltiplas plataformas.

7. Considerações Finais:

O principal objetivo desse Plano de Testes é garantir que a funcionalidade, segurança e usabilidade do aplicativo Uber estão preservadas, tornando a experiência do usuário satisfatória no processo de solicitação de corrida.