



Identificação de Fluxos Importantes

◆ Fluxo 1: Seleção de Localização e Restaurante

A seleção de localização e restaurante é a etapa principal da experiência do cliente na Coco Bambu, uma das primeiras impressões da empresa. Este é o primeiro contato do usuário com a plataforma e, qualquer erro ou dificuldade nesta parte pode levar à desistência de prosseguir, frustrando o cliente e inviabilizando a conclusão do pedido. Essa parte conecta o cliente ao cardápio disponível da loja selecionada e define a base de toda a experiência de compra.

Uma busca bem projetada permite que os clientes insiram rapidamente sua localização e encontrem as lojas disponíveis de forma assertiva. A funcionalidade precisa ser ágil e trazer resultados relevantes; caso contrário, a experiência do usuário será prejudicada, causando insatisfação e abandono do processo. Dificuldades, falta de clareza ou falhas no redirecionamento podem impactar a percepção da qualidade do serviço, comprometendo a confiança do cliente e impedindo-o de avançar para explorar o menu e concluir o pedido. Além disso, se a loja estiver fechada, não há informações claras sobre horários de funcionamento ou opções alternativas, o que pode gerar frustração e afastar o cliente da plataforma.

◆ Fluxo 2: Exploração do Cardápio e Adição ao Carrinho

A exploração do cardápio é um dos momentos mais importantes na jornada de compra do cliente, pois é quando ele decide o que deseja consumir. Esta etapa precisa ser rica em informações, intuitiva e livre de qualquer falha, para garantir uma experiência agradável e funcional.

O cardápio deve ser exibido com clareza, apresentando as opções de categorias e de itens disponíveis, acompanhados de descrições detalhadas, preços e opções de fazer alguma observação, caso aplicável. Caso as informações estejam ausentes ou inconsistentes, a percepção do cliente sobre



o serviço pode ser negativamente impactada, resultando no abandono do processo.

Além disso, a funcionalidade de adicionar itens ao carrinho deve ser intuitiva e confiável. Personalizações, como a escolha de sabores, precisam ser fáceis de realizar, e o carrinho deve refletir as escolhas do cliente em tempo real. Qualquer dificuldade em editar itens ou remover produtos pode gerar frustração, comprometendo o fluxo de compra e a satisfação final do cliente.

◆ Fluxo 3: Compra de Vouchers

A compra de vouchers é um importante diferencial competitivo, que pode influenciar diretamente a decisão de compra. Sendo esse, um baita diferencial da plataforma Coco Bambu, permitindo que os clientes adquiram créditos para serem utilizados em restaurantes, mas sem a opção de delivery. Este fluxo deve ser intuitivo e bem estruturado, garantindo que o cliente compre e utilize os vouchers de maneira eficiente e sem surpresas desagradáveis.

Uma experiência positiva começa com a seleção do estado desejado pelo cliente, etapa essencial para alinhar a experiência com a unidade de uso pretendida. A plataforma deve oferecer mensagens claras, como "Para uma melhor experiência, por favor, selecione o estado desejado. Caso prefira, ative a sua localização", para facilitar essa interação.

Os vouchers possuem regras específicas que devem ser explicitadas durante a compra para evitar confusões. Esta funcionalidade precisa ser simples e confiável para proporcionar ao cliente uma experiência positiva para evitar possíveis insatisfações.

◆ Fluxo 4: Consultar Promoções Ativas

A funcionalidade de promoções ativas no aplicativo representa um elemento estratégico para engajar o cliente e incentivar pedidos, oferecendo



COCO BAMBU

incentivos claros e acessíveis diretamente no cardápio. É um fluxo importante, pois integra as promoções de forma natural à navegação do cliente, aumentando as chances de conversão.

No menu, as promoções estão destacadas ao em banners em cima e também ao lado de categorias como lançamentos e entradas, facilitando o acesso do cliente. Ao clicar em uma promoção, o cliente é direcionado para uma seção que lista os pratos em promoção, com detalhes como preços ajustados. Essa abordagem intuitiva permite que o cliente explore as ofertas sem interromper o fluxo de navegação pelo cardápio.

Escolha de Fluxo para Realizar os Testes

Escolhi o **Fluxo 2: Exploração do Cardápio e Adição ao Carrinho**, por ser um dos pontos mais críticos da jornada do usuário. Mesmo sem realizar o cadastro ou login, é nessa parte que os clientes exploram o cardápio, conhecem os diferentes pratos, verificam promoções e personalizam seus pedidos. Além disso, esse fluxo tem um impacto direto na decisão final de compra. Problemas nesse processo podem resultar em desistência.

Comentários Sobre Testes

No teste “explore_pizza_without_login.cy.js”, se o cliente acessar a seção de compra, por exemplo, de uma pizza sem estar logado, não será possível



COCO BAMBU



selecionar os sabores nem visualizar o checkbox da opção Cobertura Extra. No entanto, ainda é possível selecioná-la apenas clicando.

Após realizar o login, a opção de selecionar o sabor e finalizar a compra aparece. Se a finalização for feita rapidamente, ocorre um erro: "Não foi possível começar um pedido" e, todo o processo de escolha da pizza é reiniciado, incluindo as observações e a seleção de Cobertura Extra, retornando à tela anterior. Caso o cliente aguarde alguns instantes, ele é redirecionado automaticamente, sem aviso prévio. Em alguns casos, mesmo logado, ele não mostra o ícone para adicionar os Sabores de Pizza, igual nas imagens abaixo:

Figura 01 – Sem a Opção de Selecionar Sabores.

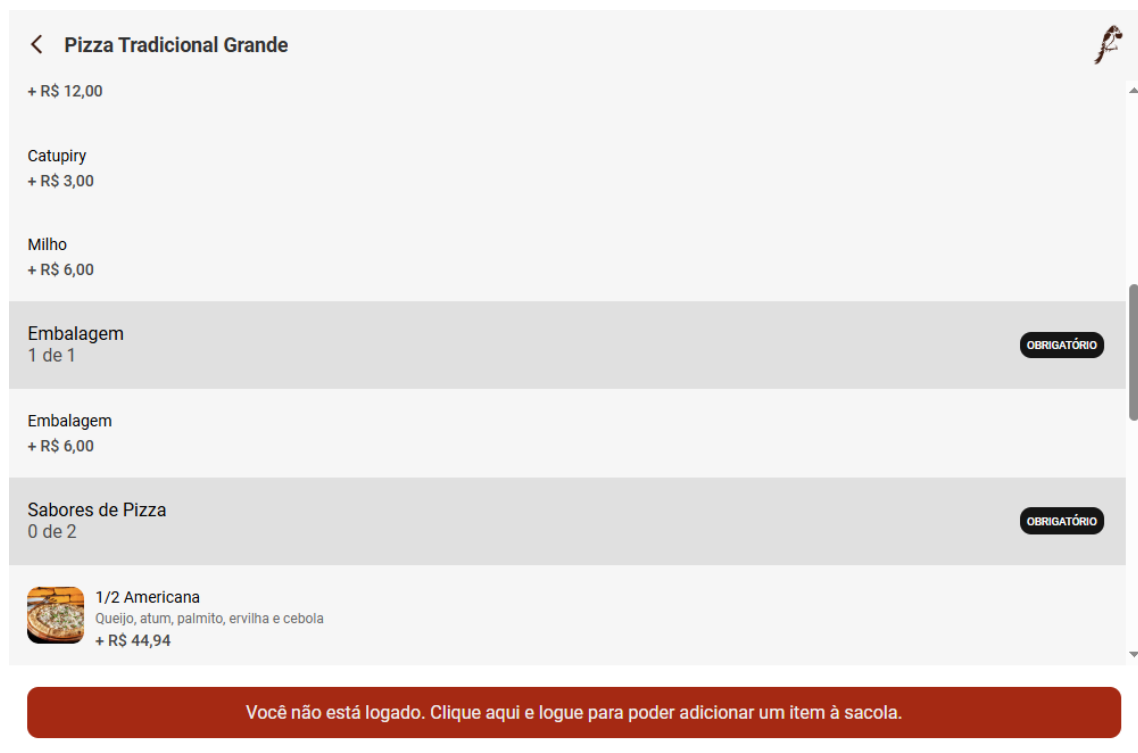


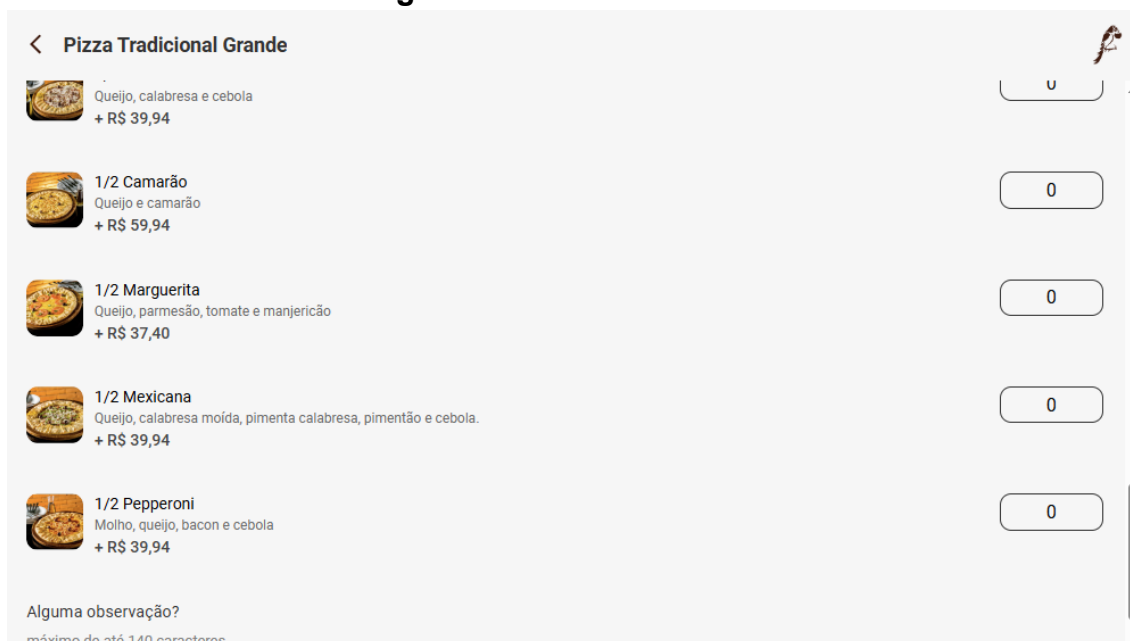
Figura 02 – Não foi Possível Começar um Pedido.

Não foi possível começar um pedido

Para efetuar algum pedido, é preciso escolher uma loja antes de criar a sacola

OK

Figura 03 – Ícones Faltando.



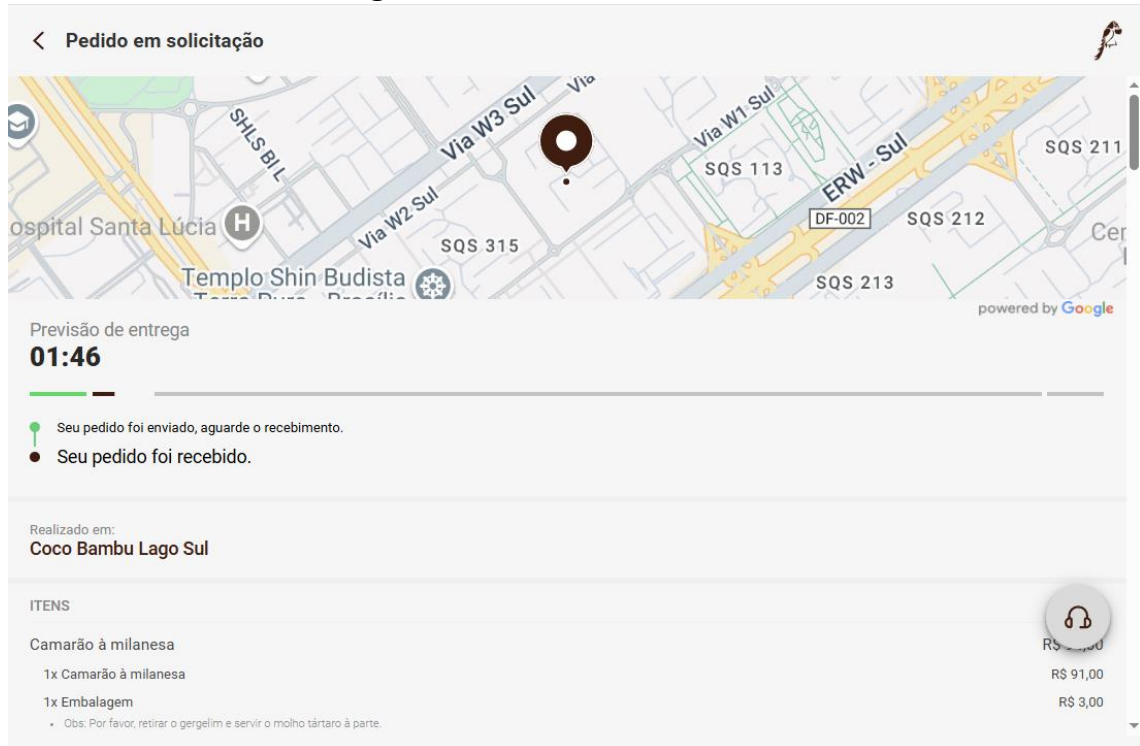
Automatizei até o final do pedido para ver como a aplicação se comportava.



COCO BAMBU



Figura 04 – Pedido Finalizado



Destaques básicos sobre a API:

- AAPI não precisa de um userId para excluir um carrinho, como mostrado no teste de delete: delete_cart_userid.cy.js. Como a API DummyJSON é simples, focada mais para testes, não há problemas.
- A API ignora produtos sem o id, mas não rejeita, criando um carrinho vazio, em: add_product_without_productid.cy.js.
- Em add_product_without_data.cy.js, se a API está retornando 201 mesmo sem quantity ou price, significa que ela não exige esses campos para processar a requisição com sucesso. Porém sem id ou com id inexistente, e usando apenas o título, ela não adiciona no carrinho e nem da mensagem de erro, apenas ignora.



COCO BAMBU