Navegador externo

Abra o checkout e preserve o contexto

Escolha a interface

Para abrir uma página geral no navegador padrão, use Linking.openURL. Em um fluxo com URL de retorno, expo-web-browser oferece openAuthSessionAsync, que acompanha a sessão até o callback ou o encerramento. Teste a experiência no Android e no iOS com uma build que registre o esquema de retorno.

Antes de abrir

Crie a tentativa no backend, associe-a à reserva e guarde seu identificador no app. O backend retorna a URL do checkout. Configure a URL de retorno de acordo com as capacidades do provedor e valide a origem da URL de checkout antes de abrir.

Separe as responsabilidades

O navegador exibe o checkout. O app lida com a volta. O backend confirma o pagamento com o provedor e atualiza a reserva. O retorno do navegador é um bom momento para consultar o backend.

Navegador externo

Consulte a reserva quando o usuário voltar

Exemplo de lógica do aplicativo

```
async function voltar(abrir, consultar, reservaId) {
  const retorno = await abrir();
  if (retorno.type !== "success") return { status: "aguardando" };
  return consultar(reservaId);
}
```

Integração no Expo

Passe como abrir uma função que chama WebBrowser.openAuthSessionAsync(checkoutUrl, retornoUrl). Passe como consultar a chamada autenticada ao endpoint de estado. Use o identificador persistido antes de abrir a sessão. O estado aguardando indica que a interface ainda deve oferecer uma consulta manual.

Atualização assíncrona

O webhook pode chegar depois do retorno. Se a API responder pendente, mantenha esse estado e permita atualizar. Ao reabrir o app após encerramento do processo, recupere a tentativa persistida e consulte o backend. O endpoint deve verificar que a reserva pertence ao usuário.

Navegador externo

Exercício e verificação

Cenários

Simule retorno bem-sucedido com API ainda pendente. Depois simule retorno com a API já confirmada. Feche a sessão manualmente e consulte o estado ao reabrir o app. Repita em cada plataforma com a URL de retorno registrada.

Resultado esperado

A interface mostra pendente ou confirmado de acordo com o backend. Se a sessão for fechada, mostra aguardando e oferece uma nova consulta. Um pagamento confirmado enquanto o app estava fechado aparece na próxima consulta.

Pergunta

Por que a API pode mostrar pendente após o callback? Porque o evento do provedor pode chegar ou ser processado depois. O servidor deve autenticar o evento e processar entregas repetidas sem duplicar a confirmação.

Referências primárias

Expo WebBrowser

Stripe Checkout fulfillment

Navegador externo

Retome a consulta ao voltar do navegador

Persista a tentativa antes de sair

Ao iniciar o checkout, grave localmente o identificador da tentativa. Quando o aplicativo voltar ao primeiro plano ou reiniciar, consulte a API. Um fechamento manual do navegador pode ocorrer depois que a pessoa já pagou.

Exercício adicional

A pessoa paga, fecha o navegador e o aplicativo perde o callback. Como recuperar a tela?

Resolução

Recupere a tentativa persistida e consulte a API autenticada. Mostre o estado que o servidor devolver. Teste fechando o navegador sem seguir o redirecionamento e reabrindo o aplicativo.

Ao assistir ao vídeo

Use o exemplo de login para observar a saída para o navegador e o retorno ao aplicativo.

Vídeo complementar

Expo - How to integrate Google Sign-In with Expo (em inglês)