

TILEBRIGHT APPS — DESENVOLVIMENTO DE
APLICATIVO MÓVEL

Especificação de Requisitos

Um aplicativo nativo de caronas compartilhadas para iOS e Android, construído e lançado antes da janela de lançamento do Ripple Ride em Austin.

Sumário

01	Controle de documentos	2
02	Introdução e escopo	3
03	Objetivo e público-alvo	3
04	Requisitos funcionais	3
05	Requisitos não funcionais	5
06	Módulos do produto	5
07	Especificação de componentes	6
08	Premissas, restrições e dependências	7
09	Fases de lançamento	8

RESUMO GERAL

Controle de documentos

ESPECIFICAÇÃO

REQ-2026-0611

Emitido em 16 de março de 2026

PREPARADO POR

Tilebright Apps

Austin, TX 78701

PROPRIETÁRIA

Priya Nair

Chefe de Operações, Ripple
Ride Carpooling

STATUS

Para revisão

Linha de base definida
na aprovação

REQUISITOS

8

Funcionais e não funcionais

01

Introdução e escopo

A Ripple Ride está abrindo seu mercado em Austin neste outono, sincronizado com o início do semestre, quando a demanda por caronas compartilhadas da UT Austin e da força de trabalho de tecnologia em geral atinge o pico. A equipe já garantiu uma parceria de recrutamento de motoristas e uma data de lançamento que não vai mudar — o que ainda falta é um aplicativo nativo capaz de suportar esse tráfego já no primeiro dia.

Objetivo e público-alvo

O Ripple Ride construiu sua rede de motoristas e garantiu uma parceria de lançamento programada para o semestre de outono, mas o produto em si ainda é um conjunto de fluxos no Figma e um backend de protótipo. A empresa tem uma única chance de causar uma primeira impressão em um mercado onde os passageiros já recorrem por hábito a aplicativos de caronas, e um app lento, cheio de falhas ou inacabado no dia do lançamento anularia meses de recrutamento de motoristas.

A Tilebright Apps propõe um projeto de dezesseis semanas para projetar e construir aplicativos nativos para iOS e Android em paralelo com um backend de pareamento de caronas, para que o Ripple Ride seja lançado com um app tão rápido e confiável quanto a rede por trás dele. Conduzimos design, os dois desenvolvimentos nativos e o QA como uma única trilha coordenada em vez de um revezamento, para que nada espere mais do que o necessário. As seções a seguir apresentam a abordagem, o escopo, o cronograma e o investimento.

Requisitos funcionais

ID	REQUISITO	PRIORIDADE
R-01	O aplicativo deve oferecer suporte nativo a dispositivos iOS 16+ e Android 12+	OBRIGATÓRIO

R-02	A correspondência de corridas deve retornar candidatos em até 2 segundos	OBRIGATÓRIO
R-03	Os perfis de motorista e passageiro devem exibir o status atual de verificação de identidade	OBRIGATÓRIO
R-04	Mensagens no aplicativo devem estar disponíveis entre motorista e passageiro correspondidos	OBRIGATÓRIO
R-05	O aplicativo deve permanecer utilizável em condições de rede 3G degradada, comuns em zonas mortas de wifi no campus	DESEJÁVEL
R-06	Notificações push obrigatórias para confirmação, cancelamento e chegada da corrida	OBRIGATÓRIO
R-07	Os fluxos principais devem passar em testes de leitor de tela VoiceOver e TalkBack	DESEJÁVEL
R-08	Eventos de analytics instrumentados para os funis de onboarding e conclusão de corrida	PODERIA

ID	PRIORIDADE	REQUISITO
R-01	Obrigatório	O aplicativo deve oferecer suporte nativo a dispositivos iOS 16+ e Android 12+
R-02	Obrigatório	A correspondência de corridas deve retornar candidatos em até 2 segundos
R-03	Obrigatório	Os perfis de motorista e passageiro devem exibir o status atual de verificação de identidade
R-04	Obrigatório	Mensagens no aplicativo devem estar disponíveis entre motorista e passageiro correspondidos
R-05	Desejável	O aplicativo deve permanecer utilizável em condições de rede 3G degradada, comuns em zonas mortas de wifi no campus

Design de produto e UX

Mapeamento de fluxos, wireframes e o sistema de UI flat, validados com sessões de usabilidade com motoristas e passageiros.

Construção para iOS

Desenvolvimento nativo em Swift/SwiftUI do aplicativo completo, do onboarding até a conclusão da corrida.

Construção para Android

Desenvolvimento nativo em Kotlin do aplicativo completo, construído em paralelo ao iOS a partir de um sistema de design compartilhado.

Backend de correspondência de corridas

Serviço de correspondência baseado em rota e horário, com atualizações de status em tempo real para ambos os aplicativos.

QA e testes em dispositivos

Testes entre dispositivos e em condições de rede degradada antes de qualquer um dos aplicativos chegar à loja.

Lançamento e envio às lojas

Listagens de loja, pacotes de revisão e apoio à submissão tanto para a App Store quanto para o Google Play.

ESPECIFICAÇÃO

Especificação de componentes

REF.	ITEM	ESPECIFICAÇÃO	UNIDADE	PREÇO	PRazo	UNITÁRIO
SPC-01	Design de produto	Sistema de design UI flat, mapeamento de fluxos e validação de usabilidade	3	3 sem.	\$7,000 / wk	
SPC-02	Desenvolvimento iOS	Build nativo em Swift/SwiftUI da experiência completa de passageiro e motorista	8	8 sem.	\$9,000 / wk	

SPC-04	Backend e API de pareamento de caronas	Serviço de pareamento baseado em rota e horário, com atualizações de status em tempo real	6	6 sem.	\$8,500 / wk
SPC-05	QA e testes em dispositivos	Etapa de QA multi-dispositivo e em rede degradada antes do envio	2	2 sem.	\$6,000 / wk
SPC-06	Lançamento e envio às lojas	Listagens de loja, capturas de tela e pacotes de envio para ambas as plataformas	1	2 de setembro de 2026	\$8,000

As especificações são fixadas no momento da aceitação; qualquer alteração é tratada por meio do processo de controle de mudanças definido neste documento.

Interface plana e legível

Um sistema de design flat contido, com blocos de cor de alto contraste, feito para uso com uma mão só em um ponto de ônibus ou na calçada.

Desempenho nativo

Bases de código Swift e Kotlin separadas, não um wrapper multiplataforma, para que correspondências e mapas continuem rápidos em celulares mais antigos.

QA pronto para lançamento

Uma rodada dedicada de QA em dispositivos e redes antes do envio, para que a primeira avaliação da App Store também seja a última.

01 SEMANAS 1-3 · DESCOBERTA E DESIGN

Mapeamento de fluxos, wireframes e o sistema de design UI flat, validados em sessões de usabilidade com motoristas e passageiros.

02 SEMANAS 4-9 · CONSTRUÇÃO PRINCIPAL

iOS, Android e o backend de pareamento de caronas desenvolvidos em paralelo a partir de um sistema de design compartilhado.

03 SEMANAS 10-12 · RECURSOS DE CONFIANÇA E SEGURANÇA

Status de verificação, avaliações e mensagens no aplicativo integrados ao fluxo principal de pareamento.

04 SEMANAS 13-14 · QA E TESTES EM DISPOSITIVOS

Testes multi-dispositivo e em rede degradada em uma matriz de celulares atuais e mais antigos.

Tilebright Apps

600 Congress Ave, Austin, TX 78701, Estados Unidos

build@tilebrightapps.com · +1 512 555 0164

· tilebrightapps.com

Tilebright Apps LLC · EIN 47-2810366