

TILEBRIGHT APPS — DESARROLLO DE
APLICACIONES MÓVILES

Especificación de Requisitos

Una aplicación nativa de viajes compartidos para iOS y Android,
construida y lanzada antes de la ventana de lanzamiento de
Ripple Ride en Austin.

Contenido

01	Control de documentos	2
02	Introducción y alcance	3
03	Propósito y público	3
04	Requisitos funcionales	3
05	Requisitos no funcionales	5
06	Módulos del producto	5
07	Especificación de componentes	6
08	Supuestos, restricciones y dependencias	8
09	Fases de lanzamiento	8

DE UN VISTAZO

Control de documentos

ESPECIFICACIÓN

REQ-2026-0611

Emitido el 16 de marzo de 2026

PREPARADO POR

Tilebright Apps

Austin, TX 78701

PROPIETARIA

Priya Nair

Responsable de Operaciones,
Ripple Ride Carpooling

ESTADO

Para revisión

Línea base fijada en la aprobación

REQUISITOS

8

Funcionales y no funcionales

01

Introducción y alcance

Ripple Ride abrirá su mercado de Austin este otoño, coincidiendo con el inicio del semestre, cuando la demanda de viajes compartidos de UT Austin y del sector tecnológico en general alcanza su punto máximo. El equipo ya tiene asegurada una alianza de reclutamiento de conductores y una fecha de lanzamiento que no se moverá; lo que aún no tienen es una aplicación nativa capaz de soportar ese tráfico desde el primer día.

Propósito y público

Ripple Ride ha construido su red de conductores y asegurado una alianza de lanzamiento programada para el semestre de otoño, pero el producto en sí sigue siendo un conjunto de flujos en Figma y un backend prototipo. La empresa tiene una sola oportunidad de causar una primera impresión en un mercado donde los usuarios ya recurren por hábito a las aplicaciones de transporte compartido, y una aplicación lenta, con errores o a medio terminar el día del lanzamiento echaría por tierra meses de reclutamiento de conductores.

Tilebright Apps propone una contratación de dieciséis semanas para diseñar y construir aplicaciones nativas de iOS y Android en paralelo con un backend de emparejamiento de viajes, de modo que Ripple Ride se lance con una aplicación que se sienta tan rápida y confiable como la red que la respalda. Manejamos el diseño, ambas construcciones nativas y el control de calidad como una sola vía coordinada en lugar de una carrera de relevos, de modo que nada espere más de lo estrictamente necesario. Las siguientes secciones exponen el enfoque, el alcance, el cronograma y la inversión.

Requisitos funcionales

ID	REQUISITO	PRIORIDAD
R-01	La aplicación debe ser compatible de forma nativa con dispositivos iOS 16+ y Android 12+	OBLIGATORIO

R-02	La búsqueda de coincidencias de viaje debe devolver candidatos en un plazo de 2 segundos	OBLIGATORIO
R-03	Los perfiles de conductor y pasajero deben mostrar el estado actual de verificación de identidad	OBLIGATORIO
R-04	La mensajería dentro de la app debe estar disponible entre el conductor y el pasajero emparejados	OBLIGATORIO
R-05	La app debe seguir siendo utilizable en condiciones de red 3G degradada, comunes en zonas muertas de wifi del campus	DESEABLE
R-06	Se requieren notificaciones push para confirmación, cancelación y llegada del viaje	OBLIGATORIO
R-07	Los flujos principales deben superar las pruebas con lector de pantalla VoiceOver y TalkBack	DESEABLE
R-08	Eventos de analítica instrumentados para los embudos de incorporación y finalización de viaje	PODRÍA

ID	PRIORIDAD	REQUISITO
R-01	Obligatorio	La aplicación debe ser compatible de forma nativa con dispositivos iOS 16+ y Android 12+
R-02	Obligatorio	La búsqueda de coincidencias de viaje debe devolver candidatos en un plazo de 2 segundos
R-03	Obligatorio	Los perfiles de conductor y pasajero deben mostrar el estado actual de verificación de identidad
R-04	Obligatorio	La mensajería dentro de la app debe estar disponible entre el conductor y el pasajero emparejados
R-05	Deseable	La app debe seguir siendo utilizable en condiciones de red 3G degradada, comunes en zonas muertas de wifi del campus

Diseño de producto y UX

Mapeo de flujos, wireframes y el sistema de UI plano, validados con sesiones de usabilidad de conductores y pasajeros.

Desarrollo para iOS

Desarrollo nativo en Swift/SwiftUI de la aplicación completa, desde la incorporación hasta la finalización del viaje.

Desarrollo para Android

Desarrollo nativo en Kotlin de la aplicación completa, construido en paralelo con iOS a partir de un sistema de diseño compartido.

Backend de emparejamiento de viajes

Servicio de emparejamiento basado en ruta y horario, con actualizaciones de estado en tiempo real para ambas aplicaciones.

Control de calidad y pruebas en dispositivos

Pruebas multidispositivo y de red degradada antes de que cualquiera de las aplicaciones llegue a la tienda.

Lanzamiento y envío a tiendas

Fichas de tienda, paquetes de revisión y apoyo de envío tanto para App Store como para Google Play.

ESPECIFICACIÓN

Especificación de componentes

REF.	ÍTEM	ESPECIFICACIÓN	UNIDADES	TERMINO	PRECIO UNITARIO
SPC-01	Diseño de producto	Sistema de diseño de interfaz plana, mapeo de flujos y validación de usabilidad	3	3 sem.	\$7,000 / wk
SPC-02	Desarrollo iOS	Compilación nativa en Swift/SwiftUI de la experiencia completa de pasajero y conductor	8	8 semanas	\$9,000 / wk

SPC-03	Desarrollo Android	Compilación nativa en Kotlin de la experiencia completa de pasajero y conductor	8	8 semanas	\$9,000 / wk
SPC-04	Backend y API de emparejamiento de viajes	Servicio de emparejamiento basado en ruta y horario con actualizaciones de estado en tiempo real	6	6 sem.	\$8,500 / wk
SPC-05	Control de calidad y pruebas en dispositivos	Fase de control de calidad multiplataforma y en red degradada previa al envío	2	2 sem.	\$6,000 / wk
SPC-06	Lanzamiento y envío a tiendas	Fichas de tienda, capturas de pantalla y paquetes de envío para ambas plataformas	1	2 de septiembre de 2026	\$8,000

Las especificaciones quedan fijadas en el momento de la aceptación; cualquier cambio se gestiona mediante el procedimiento de control de cambios establecido en este documento.

Interfaz plana y legible

Un sistema de diseño plano y sobrio, con bloques de color de alto contraste, pensado para usarse con una sola mano en una parada de autobús o en la acera.

01

Rendimiento nativo

Bases de código independientes en Swift y Kotlin, no un envoltorio multiplataforma, para que la coincidencia de datos y los mapas se mantengan ágiles incluso en teléfonos antiguos.

02

Control de calidad listo para el lanzamiento

Una ronda de pruebas dedicada de dispositivos y red antes del envío, de modo que la primera revisión de la App Store sea también la última.

03

01 SEMANAS 1-3 · DESCUBRIMIENTO Y DISEÑO

Mapeo de flujos, wireframes y el sistema de diseño de interfaz plana, validados en sesiones de usabilidad con conductores y pasajeros.

02 SEMANAS 4-9 · DESARROLLO PRINCIPAL

iOS, Android y el backend de emparejamiento de viajes desarrollados en paralelo a partir de un sistema de diseño compartido.

03 SEMANAS 10-12 · FUNCIONES DE CONFIANZA Y SEGURIDAD

Estado de verificación, calificaciones y mensajería dentro de la app añadidos sobre el flujo principal de emparejamiento.

04 SEMANAS 13-14 · CONTROL DE CALIDAD Y PRUEBAS EN DISPOSITIVOS

Pruebas multiplataforma y en condiciones de red degradada en una matriz de teléfonos actuales y antiguos.



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

