

시니어·아이 대상 택시 호출 앱 React Native 프론트엔드 개발

상세기획서 - 계약 전 검토용 공개 제출 자료

공개 데모	https://taxi-mobility-rn-frontend-demo.pages.dev/#pre-contract-review
포트폴리오 허브	https://nelydo-public-project-hub.pages.dev/
구현 기준	React Native iOS/Android 단일 코드베이스, Supabase 클라이언트 연동, 카카오맵 SDK, 토스페이먼츠, Firebase FCM
문서 성격	계약 전 요구사항 이해와 실행 범위 검토를 위한 제안 자료입니다. 실제 운영 API/결제/푸시/위치/개인정보 처리는 계약 후 제공자료와 승인 범위에서만 연결합니다.

본 자료는 발주처가 제공 예정인 Figma, Supabase 스키마/RLS/Edge Function/OpenAPI, 카카오맵·토스페이먼츠·Firebase 계정 및 Apple/Google 개발자 권한을 받은 뒤 실제 React Native 앱 구현과 내부 테스트 빌드까지 진행하기 위한 계약 전 검토 문서입니다.

1. 프로젝트 이해

본 프로젝트는 시니어 또는 아이가 복잡한 출발지·목적지 입력 없이 보호자가 미리 등록된 가상정류장 카드를 선택해 단 2번의 터치로 택시를 호출할 수 있게 하는 React Native 기반 iOS/Android 앱 프론트엔드 개발입니다.

핵심 사용자는 두 그룹입니다. 이용자는 큰 버튼, 낮은 인지부하, 호출 전 안전장치, SOS 접근성이 중요합니다. 보호자/결제자는 정류장 관리, 실시간 호출 상태 확인, 위치·경로 확인, 결제 상태 관리, SOS 대응이 중요합니다.

발주처는 기획, Figma, Supabase 백엔드, OpenAPI 명세, 외부 API 계정 및 개발자 계정을 제공합니다. 파트너사는 React Native 클라이언트 구현과 외부 SDK/API 연동, 내부 테스트 빌드 산출에 집중합니다.

계약 전 공개 데모 상태

현재 공개 데모는 승객 호출, 보호자 모니터링, 운영자 관제 흐름을 sandbox/staging 관점으로 시각화합니다. 실제 택시 호출, 실제 결제, SMS, Push, 백그라운드 위치추적, 실제 개인정보 처리는 연결하지 않았습니다.

2. 공고 요구사항 대응 매트릭스

영역	공고 요구	구현 접근
이용자 앱	2-Tap 호출, 가까운 가상정류장, 6단계 상태 UI, 4초 카운트다운, SOS 오버레이	역할 기반 라우팅, 상태머신, 접근성 UI, 보호자 등록 정류장 카드 우선 노출
결제자 앱	정류장 관리, 실시간 모니터링, SOS 수신, 결제 관리, 초대 기능	Supabase Realtime 구독, 지도/경로 표시, FCM 수신, Universal/App Link 초대
Supabase	Auth, DB, Realtime, Storage, SMS 인증, RLS 기반 접근	발주처 DDL/RLS/OpenAPI 기준 client adapter와 error/fallback UI 구현
카카오맵	RN 브릿지, 정류장/차량 위치, 이동 경로 지도 표시	iOS/Android native bridge wrapper, marker/path update, 권한/fallback 처리
토스페이먼츠	빌링키 정기결제, 건당 결제, 실패 재시도	test/sandbox 우선, production key 미사용, 결제 결과 callback/deeplink 처리
Firebase FCM	상태별 푸시, SOS 즉시 알림	foreground/background/terminated 수신 테스트와 token lifecycle 관리
빌드 산출	TestFlight, Google Play Internal Testing 진입용 빌드	iOS/Android signing, env 분리, QA 후 내부 테스트 제출 준비

3. 이용자 앱 구현 방향

홈 화면

다크 모드 기반의 큰 2-Tap 호출 CTA, SOS 버튼, 보호자 등록 가상정류장 카드 우선 노출을 적용합니다. 시니어/아이 사용자가 길게 읽지 않아도 다음 행동을 알 수 있게 버튼과 상태 문구를 분리합니다.

정류장 선택

GPS 기반 가까운 가상정류장을 우선 정렬하고, 즐겨찾기·최근 이용 정류장을 함께 노출합니다. 위치 권한 거부 또는 위치 값 없음 상태에서는 보호자 등록 정류장 기준 fallback을 제공합니다.

호출 진행

6단계 상태머신에 따라 UI를 갱신합니다. matching, driver_assigned, en_route_pickup, riding, completed, cancelled/failed 상태별로 문구와 색상, 버튼 활성 상태를 분리합니다.

Safe-guard

호출 직전 4초 카운트다운과 취소 버튼을 제공합니다. 오작동 방지를 위해 호출 전 정류장명, 보호자명, 취소 가능 시간을 명확히 보여줍니다.

SOS 오버레이

가족, 기사, 운영자 3단 액션 카드를 제공하고, FCM/Realtime 이벤트와 연결해 보호자 앱에 즉시 동기화되도록 구현합니다.

4. 결제자/보호자 앱 구현 방향

가상정류장 관리

이용자별 정류장 등록/수정/삭제, 지도 기반 위치 확인, 별칭 설정, 사용 가능 시간대 확장을 고려합니다.

실시간 모니터링

Supabase Realtime으로 호출 상태를 구독하고, 카카오맵 marker/path로 이용자 위치 및 이동 경로를 표시합니다.

SOS 수신

SOS 발생 시 푸시 알림과 앱 내부 배너/모달을 함께 제공합니다. 가족/기사/운영자 대응 버튼을 한 화면에서 선택할 수 있게 구성합니다.

결제 관리

Premium 안내 모달, 결제 실패 안내, 카드 변경 유도, 현재 요금제 상태 표시를 제공합니다. 실제 결제 실행은 토스페이먼츠 test/sandbox부터 검증합니다.

초대 기능

Universal Link/App Link 기반 이용자 초대 흐름을 구현하고, 초대 수락 이후 보호자-이용자 연결 상태를 표시합니다.

5. 권장 앱 구조

React Native 단일 코드베이스 안에서 이용자 앱과 결제자 앱을 역할 기반 라우팅으로 분리합니다. 역할이 명확히 분리되면 앱 배포 단위가 하나여도 화면/권한/상태 흐름을 안정적으로 관리할 수 있습니다.

모듈	역할
app shell	인증 상태, 권한 상태, 네비게이션, 테마 전환
features/passenger	호출 홈, 정류장 선택, 호출 진행, SOS
features/payer	이용자 관리, 정류장 관리, 모니터링, 결제, 초대
integrations/supabase	auth, database, realtime channel, storage
integrations/maps	Kakao Map bridge wrapper, marker/path API
integrations/payments	Toss payment client, billing key flow, result callback
integrations/push	FCM token, notification permission, message handler
shared/state	server state, 상태머신, idempotency guard
shared/ui	공통 버튼, 카드, 모달, 접근성 컴포넌트

6. 상태머신 설계

호출 흐름은 상태머신으로 제어합니다. 프론트는 Supabase Realtime 또는 OpenAPI 응답을 기준으로 상태를 갱신하고, 중복 호출 방지를 위해 client-side idempotency key를 함께 사용합니다.

상태	이용자 UI	보호자 UI
ready	호출 가능, 정류장 카드 선택	이용자 상태 정상
confirming	4초 카운트다운, 취소 가능	호출 준비 알림
matching	기사 매칭 중	매칭 상태 실시간 표시
driver_assigned	기사 배경 완료	기사/차량 정보 표시
en_route_pickup	픽업지 이동 중	차량 위치/경로 표시
riding	탑승/이동 중	이동 경로 모니터링
completed	도착/완료	완료 및 결제 상태 표시
cancelled/failed	취소/실패 안내	원인 및 재호출 안내

7. Supabase 연동 계획

Auth

이용자/보호자 인증, SMS 인증, 세션 복원, 로그아웃을 구현합니다.

DB

이용자, 보호자, 가상정류장, 호출, 위치, SOS, 결제 상태 조회/변경을 발주처 스키마 기준으로 구현합니다.

Realtime

호출 상태, SOS 상태, 위치 상태 변경 구독과 끊김/복구 동작을 테스트합니다.

Storage

필요 시 프로필, 약관, 정류장 관련 파일 접근을 구현합니다.

RLS

클라이언트 권한 범위 오류를 안전하게 처리하고, 비허용 데이터 접근을 UI에서 차단합니다.

Edge Function

호출 생성, 결제 콜백, 초대 수락 등 서버 검증이 필요한 작업을 명세 기준으로 호출합니다.

8. 카카오맵·토스페이먼츠·Firebase 연동 계획

연동	구현 포인트	검증 포인트
카카오맵 SDK	정류장 marker, 차량 marker, 경로 polyline, 위치 권한, fallback UI	iOS/Android bridge 초기화, 지도 lifecycle, 권한 거부/위치 없음 처리
토스페이먼츠	빌링키, 정기결제, 건당결제, 실패 재시도, 카드 변경 유도	test/sandbox 우선, production key 미사용, callback/deeplink 결과 처리
Firebase FCM	SOS, 기사 배정, 이동 상태, 결제 실패, 초대 알림	foreground/background/terminated 수신, token refresh, 로그아웃 정리
Universal/App Link	초대 링크, 결제 결과 복귀, 딥링크 라우팅	iOS associated domains, Android assetlinks, 잘못된 링크 fallback

9. 60일 실행 계획

기간	주요 작업	완료 기준
1주차	제공자료 확인, Figma 분석, Supabase/OpenAPI 확인, RN 프로젝트 세팅	환경/라우팅/빌드 flavor 기본 구조
2주차	공통 UI, 네비게이션, 인증, 역할 기반 app shell	이용자/보호자 진입 flow smoke
3주차	이용자 홈, 정류장 선택, 카운트다운, 호출 상태 UI, SOS	이용자 핵심 flow smoke PASS
4주차	보호자 정류장 관리, 이용자 연결, 실시간 모니터링 기본 화면	보호자 핵심 flow smoke PASS
5주차	Supabase Realtime, 위치 polling, idempotency, SOS/호출 동기화	상태 동기화·끊김 복구 확인
6주차	카카오맵 브릿지, marker/path, 위치 권한, 지도 fallback	iOS/Android 지도 smoke PASS
7주차	토스페이먼츠, FCM, Universal/App Link 연동 및 회귀 테스트	결제/푸시/딥링크 test PASS
8주차	TestFlight/Internal Testing 빌드 준비, QA, 문서화, 소스 이관 정리	내부 테스트 빌드 및 인계 패키지

10. 산출물

- React Native iOS/Android 단일 코드베이스
- 이용자 앱 / 결제자 앱 화면 및 기능 구현
- Supabase, 카카오맵, 토스페이먼츠, Firebase FCM 클라이언트 연동 코드
- IPA 또는 TestFlight 진입용 iOS 빌드 산출
- AAB 또는 Google Play Internal Testing 진입용 Android 빌드 산출
- 클라이언트 연동 문서
- 빌드/배포 절차 문서
- GitHub 저장소 이관 체크리스트

11. 착수 전 요청자료

자료	필요 이유
Figma 파일 및 디자인 토큰	화면 1:1 구현, 상태별 UI, 다크/라이트 모드 기준 확정
Supabase URL, anon key, DDL, RLS, Edge Function 자료	Auth/DB/Realtime/Storage 클라이언트 연동 및 권한 오류 처리
OpenAPI 3.0 명세서	호출 생성, 결제 상태, 초대, SOS 등 서버 API 계약 확정
카카오맵 SDK 키 및 iOS/Android 설정 정보	지도 브릿지 초기화, marker/path, 위치 권한 설정
토스페이먼츠 test key 및 callback/deep link 정책	빌링키/건당결제 test flow와 결제 결과 복귀 처리
Firebase 프로젝트 권한 및 설정 파일	FCM token, push 수신, foreground/background 테스트
Apple/Google 개발자 계정 초대 권한	TestFlight 및 Google Play Internal Testing 진입용 빌드
약관/개인정보/위치 권한 문구, 테스트 계정-데이터	법적 고지 UI, QA, 클로즈드 베타 준비

12. 품질 게이트

- Figma 주요 화면 대응 확인
- iOS/Android 네이티브 빌드 성공
- 핵심 사용자 flow smoke PASS
- Supabase 권한/RLS 오류 처리 확인
- Realtime 끊김/복구 동작 확인
- 지도 권한 거부/위치 없음/SDK 실패 fallback 확인
- 결제 성공/실패/재시도/카드 변경 flow 확인
- 푸시 foreground/background/terminated 수신 확인
- 개인정보/결제/위치 관련 production key 미사용 확인
- TestFlight/Internal Testing 진입 가능 상태 확인

13. 미팅 확인 질문

- 사용자 앱과 결제자 앱을 하나의 앱에서 역할로 분리할지, 별도 앱처럼 배포할지 확인이 필요합니다.
- 택시 실제 배차 API는 발주처 백엔드가 추상화해서 제공하는지, 앱이 직접 외부 API를 호출하는지 확인이 필요합니다.
- 위치 polling 15초 주기의 정확한 주체가 앱인지 서버인지 확인이 필요합니다.
- 결제 실패 재시도 정책과 Premium 전환 조건을 확인해야 합니다.
- SOS 발생 시 가족/기사/운전자 중 어느 채널이 우선인지 확인해야 합니다.
- 클로즈드 베타 범위, 테스트 계정 수, 테스트 단말 기준을 확인해야 합니다.

14. 결론

이 프로젝트의 핵심은 단순 화면 구현보다 안전한 호출 경험, 보호자 모니터링, 실시간 상태 동기화, 지도·결제·푸시 연동 안정성입니다. 계약 전 공개 데모는 요구사항 이해와 구조를 검토하기 위한 자료이며, 실제 납품 구현은 발주처 제공자료를 기준으로 React Native 실앱과 내부 테스트 빌드 산출까지 진행합니다.