

BusTrackNow — Complete Project Blueprint

React Native + Firebase Zero-Cost Architecture

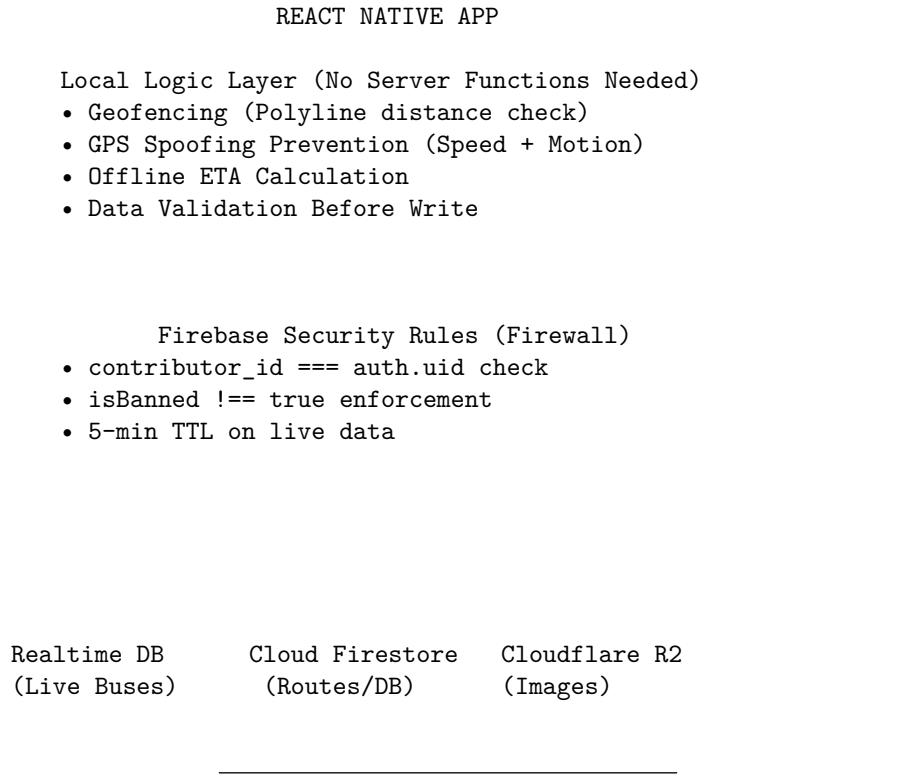
Table of Contents

1. Tech Stack & Architecture
 2. Complete User Workflow (15 Screens)
 3. Complete Developer Workflow (Setup → Deploy)
 4. Screen-by-Screen Breakdown (All 28 Screens)
 5. Data Structure & Security Rules
 6. Points Ecosystem & Revenue Streams
-

Tech Stack (Free Tier Optimized)

Component	Technology	Why This Choice
Frontend	React Native 0.73+	Single codebase for Android (iOS-ready)
Styling	NativeWind (Tailwind for RN)	Rapid UI dev, consistent design tokens, JIT
Navigation	React Navigation 6	Stack + Bottom Tab + Modal patterns
State Management	React Context + useReducer	Lightweight, no Redux overhead
Auth	Firebase Auth	Unlimited free OTP logins
Live Data	Firebase Realtime DB	High-frequency GPS updates, 100 concurrent free
Static Data	Cloud Firestore	Routes/stops/users, 50k reads/day free
Storage	Cloudflare R2	10GB free + \$0 egress (replaces Firebase Storage)
Maps	react-native-maps	Google Maps SDK (free credit) or OpenStreetMap
Local Storage	@react-native-async-storage/async-storage	Cache routes/stops, save quota
Animations	react-native-reanimated 3	Smooth 60 FPS animations
Icons	react-native-vector-icons	Lightweight, offline-ready
Build	EAS Build (Expo)	Free tier for Android APK

Architecture: Smart Client Model



Complete User Workflow (15 Core Screens)

Phase 1: First Launch & Onboarding

Step	Screen	Key Actions	Data Flow
1	Splash Screen	Auto-redirect after 1.2s	Check AsyncStorage for <code>hasOnboarded</code>
2	Onboarding (4 slides)	Skip / Get Started	Set <code>hasOnboarded: true</code>
3	Phone Auth	Enter number → Request OTP	Firebase Auth → <code>auth.uid</code>
4	OTP Verify	Enter 6-digit code	Verify → Create user doc in Firestore
5	Profile Setup	Name, Language, UPI ID	Write to <code>users/{uid}</code>

Phase 2: Core Experience

Step	Screen	Key Actions	Data Flow
6	Home Map	Search route, View live buses	Subscribe to <code>active_buses/{routeID}</code>
7	Route Search	Type → Local search from cached JSON	Filter routes from AsyncStorage
8	Route Detail	View stops, ETA, bookmark	Read from cached routes JSON
9	Contribution Flow	Tap FAB → Confirm bus → Share location	Write to RTDB every 3s
10	Active Sharing State	“Stop Sharing” button	Monitor accelerometer + GPS
11	Service Alert	Report “Full”/“Late”/“Not Running”	Write to Firestore <code>alerts</code>

Phase 3: Rewards & Gamification

Step	Screen	Key Actions	Data Flow
12	Wallet Home	View points, Progress ring	Read <code>users/{uid}/points</code>
13	Points History	Timeline of earnings	Query Firestore <code>transactions</code>
14	Redeem Flow	Enter amount → Verify UPI → Request	Write to <code>redemptions</code>
15	UPI Verification	1 micro-transaction test	External UPI intent → Confirm

Phase 4: Discovery & Settings

Step	Screen	Key Actions	Data Flow
16	Saved Routes	Quick access to favorites	Read <code>users/{uid}/bookmarks</code>
17	Leaderboard	View local rankings	Query Firestore <code>leaderboard</code>
18	Badges & Levels	View unlocked achievements	Read <code>users/{uid}/badges</code>

Step	Screen	Key Actions	Data Flow
19	Community Alerts	Browse reports from others	Subscribe to <code>alerts</code> collection
20	Notifications	System updates, redemption status	Firebase Cloud Messaging
21	Settings	Language, UPI edit, Privacy	Update <code>users/{uid}</code>

Complete Developer Workflow

Phase 1: Setup (Day 1-2)

1. Initialize React Native Project

```
npx create-expo-app BusTrackNow
cd BusTrackNow
```

2. Install Dependencies

```
npm install @react-navigation/native @react-navigation/bottom-tabs @react-navigation/stack
npm install react-native-screens react-native-safe-area-context
npm install @react-native-async-storage/async-storage
npm install react-native-maps
npm install react-native-reanimated
npm install react-native-vector-icons
npm install nativewind
npm install tailwindcss
```

3. Firebase Setup

```
npm install @react-native-firebase/app
npm install @react-native-firebase/auth
npm install @react-native-firebase/database
npm install @react-native-firebase/firestore
npm install @react-native-firebase/messaging
```

4. Cloudflare R2 (S3-compatible)

```
npm install @aws-sdk/client-s3
```

Phase 2: Firebase Configuration (Day 2-3)

```
// firebase/config.js
import { initializeApp } from '@react-native-firebase/app';
import auth from '@react-native-firebase/auth';
import database from '@react-native-firebase/database';
```

```

import firestore from '@react-native-firebase/firestore';

const firebaseConfig = {
  apiKey: "YOUR_API_KEY",
  authDomain: "bustracknow.firebaseio.com",
  databaseURL: "https://bustracknow-default-rtdb.firebaseio.com",
  projectId: "bustracknow",
  storageBucket: "bustracknow.appspot.com",
  messagingSenderId: "123456789",
  appId: "1:123456789:android:abcdef"
};

const app = initializeApp(firebaseConfig);

export { auth, database, firestore };

```

Phase 3: NativeWind Setup (Day 3)

```

// tailwind.config.js
module.exports = {
  content: ["./App.{js,jsx,ts,tsx}", "./src/**/*.{js,jsx,ts,tsx}"],
  presets: [require("nativewind/preset")],
  theme: {
    extend: {
      colors: {
        primary: '#0D9488', // Teal
        secondary: '#2563EB', // Blue
        success: '#10B981', // Green
        warning: '#F59E0B', // Amber
        danger: '#EF4444', // Red
      }
    }
  }
}

// babel.config.js
module.exports = function(api) {
  api.cache(true);
  return {
    presets: ['babel-preset-expo'],
    plugins: ['nativewind/babel'],
  };
};

```

Phase 4: Security Rules (Day 4)

```
// Firestore Rules
rules_version = '2';
service cloud.firestore {
  match /databases/{database}/documents {
    // Users can only read/write their own data
    match /users/{userId} {
      allow read: if true;
      allow write: if request.auth != null && request.auth.uid == userId;
    }

    // Routes are public read, admin write
    match /routes/{routeId} {
      allow read: if true;
      allow write: if get(/databases/$(database)/documents/users/$(request.auth.uid)).data.r
    }

    // Alerts: anyone can write, public read
    match /alerts/{alertId} {
      allow read: if true;
      allow create: if request.auth != null;
      allow update, delete: if request.auth.uid == resource.data.contributor_id;
    }

    // Redemptions: user owns their requests
    match /redemptions/{requestId} {
      allow read: if request.auth.uid == resource.data.userId;
      allow create: if request.auth != null;
    }
  }
}

// Realtime Database Rules
{
  "rules": {
    "active_buses": {
      "$routeId": {
        "$sessionId": {
          ".write": "auth != null &&
            (data.child('contributor_id').val() === auth.uid ||
              root.child('admins').child(auth.uid).exists())",
          ".read": "true",
        }
      }
    }
  }
}
```

```

    // Validate structure
    ".validate": "newData.hasChildren(['lat', 'lng', 'timestamp', 'contributor_id']) &
      newData.child('timestamp').val() > (now - 300000)" // 5-min TTL
    }
  }
},
"admins": {
  ".read": "root.child('admins').child(auth.uid).exists()", 
  ".write": "false" // Only set via Firebase Console
}
}
}

```

Phase 5: Core Feature Implementation (Day 5-14)

Day 5-6: Auth Flow

- Phone input screen → OTP screen → Profile setup
- Firebase Auth integration
- User document creation in Firestore

Day 7-8: Map & Routing

- Google Maps integration with `react-native-maps`
- Route polyline rendering
- Bus marker clustering
- Search bar with local filtering

Day 9-10: Contribution System

- Geofencing logic (distance to polyline < 50m)
- GPS broadcast every 3s to RTDB
- Accelerometer monitoring for auto-stop
- “I’m on this bus” FAB flow

Day 11-12: Points & Wallet

- Points calculation logic
- Wallet screen with tabs
- UPI verification modal
- Redemption flow

Day 13-14: Gamification & Alerts

- Badge system

- Leaderboard (opt-in)
 - Service alert reporting
 - Notifications via FCM
-

Phase 6: Testing & Deployment (Day 15-21)

```
# Test on Android device
npm run android

# Build APK with EAS
eas build --platform android --profile preview

# Submit to Play Store
eas submit --platform android
```

Complete Screen Inventory (28 Screens)

Authentication Flow (5 Screens)

#	Screen Name	Key Components	States
1	Splash	Logo + animated bus route line	Loading → Auto-redirect
2	Onboarding Slide 1	Illustration + “Tired of waiting?” headline	Swipe / Skip
3	Onboarding Slide 2	Map mockup + “See buses live”	Swipe / Skip
4	Onboarding Slide 3	Community graphic + “Everyone contributes”	Swipe / Skip
5	Onboarding Slide 4	Coin animation + “Earn cash rewards”	Get Started

#	Screen Name	Key Components	States
6	Phone Auth	Country code dropdown, Phone input, “Send OTP” button	Idle / Sending / Error
7	OTP Verify	6-digit input boxes, Resend timer, Auto-read OTP	Idle / Verifying / Success / Failed

#	Screen Name	Key Components	States
8	Profile Setup	Name input, Language picker, UPI field with info card	Idle / Saving / Success
9	UPI Info Modal	“Why we ask for UPI” explanation, Security reassurance	Open / Closed

Main Navigation (5 Tabs = 15+ Screens)

Tab 1: Map (Home) - 4 Screens

#	Screen Name	Key Components	States
10	Home Map	Map view, Search bar, Wallet chip, FAB, Bottom sheet	Default / Loading routes / Offline
11	Bottom Sheet (Collapsed)	Recent routes, Suggested routes, ETA previews	Collapsed / Expanding
12	Bottom Sheet (Expanded)	Route cards list, Filter chips (Nearest/Fastest)	Expanded / Filtering
13	Route Search Modal	Large search input, Smart suggestions list	Typing / Results / No results

Tab 2: Routes - 3 Screens

#	Screen Name	Key Components	States
14	Routes List	Search bar, Route cards grid, Filter dropdown	Loading / Default / Empty
15	Route Detail	Route header, Mini map, Stop list with ETA, Bookmark button	Loading / Default / Error
16	Stop Detail (Modal)	Stop name, Next bus ETA, “Confirm next stop” CTA	Default / Confirming / Confirmed

Tab 3: Wallet - 5 Screens

#	Screen Name	Key Components	States
17	Wallet Home	Points total, Cash equivalent, Progress ring, Tabs	Default / Loading
18	Points History	Timeline list, Entry type badges, Status pills	Loading / Default / Empty / Offline
19	Redeem Screen	Available amount, Minimum threshold hint, UPI preview, CTA	Disabled (< 100) / Eligible / Processing
20	UPI Verification	1 micro-transaction explanation, “Verify Now” button	Idle / Verifying / Success / Failed
21	Redemption Receipt	Amount, Date, Transaction ID, Share button	Success / Failed

Tab 4: Alerts - 3 Screens

#	Screen Name	Key Components	States
22	Community Alerts	Alert cards list, Filter by type (Full/Late/Not Running)	Loading / Default / Empty
23	Report Alert	Large buttons: “Full” / “Late” / “Not Running”, Optional comment	Default / Submitting / Success
24	Alert Detail	Alert info, Contributor badge, Timestamp, Upvote button	Default / Voted

Tab 5: Settings - 4 Screens

#	Screen Name	Key Components	States
25	Settings Home	Profile section, UPI management, Language picker, Notifications toggle	Default
26	Saved Routes	Favorited routes list, Quick open buttons, Unfavorite option	Loading / Default / Empty
27	Badges & Levels	Badge grid, Unlock criteria, Level progress bar	Default
28	Leaderboard	Rank list, Points display, Opt-in toggle, Privacy notice	Loading / Default / Opt-out

Overlay Components (Modals & Sheets)

Component	Trigger	Content
Active Sharing Badge	User taps FAB	“Sharing Live Location” + “Stop Sharing” button + Battery warning
Points Earned Toast	Contribution confirmed	Coin burst animation + “+10 Points” + Breakdown link
Offline Warning Banner	Network disconnected	“Showing cached data” + Sync status
Permission Request Modal	First location access	“Allow location for live tracking” + Benefits explanation

Data Structure

Cloud Firestore (Static Data)

```
// Routes Collection
routes/{routeId}
  routeNumber: "45"
  origin: "Gandhi Chowk"
  destination: "Railway Station"
  stops: ["stop1", "stop2", "stop3"]
  polyline: "encoded_polyline_string"
```

```

averageSpeed: 25 // km/h
isActive: true

// Stops Collection
stops/{stopId}
  name: "College Road"
  lat: 11.0168
  lng: 76.9558
  routes: ["route45", "route12"]
  landmark: "Near Post Office"

// Users Collection
users/{userId}
  name: "Rahul"
  phone: "+919876543210"
  language: "hi" // hi, ta, te, en
  upiId: "rahul@paytm" // optional
  points: 8500
  trustScore: 95
  isBanned: false
  role: "user" // or "admin"
  streakDays: 7
  lastContribution: timestamp
  bookmarks: ["route45", "route12"]

// Alerts Collection
alerts/{alertId}
  routeId: "route45"
  type: "full" // full, late, not-running
  contributorId: "userId"
  location: { lat, lng }
  comment: "Very crowded"
  upvotes: 12
  timestamp: serverTimestamp()
  status: "active" // active, resolved

// Redemptions Collection
redemptions/{requestId}
  userId: "userId"
  amount: 150 // INR
  pointsUsed: 15000
  upiId: "rahul@paytm"
  status: "pending" // pending, processing, completed, failed
  transactionId: "TXN123456"
  requestedAt: timestamp
  completedAt: timestamp

```

```
// Global Config
global/config
  dataVersion: 12
  minRedemption: 100 // 100
  pointsPerMinute: 5
  maintenanceMode: false
```

Realtime Database (Live Data)

```
active_buses/
  route45/
    session_abc123/
      lat: 11.0168
      lng: 76.9558
      speed: 22.5
      heading: 45
      contributorId: "userId"
      startedAt: 1704067200000
      timestamp: 1704067230000
    session_def456/
      lat: 11.0180
      lng: 76.9570
      contributorId: "userId2"
      timestamp: 1704067235000
  route12/
    session_xyz789/
      lat: 11.0200
      lng: 76.9600
      timestamp: 1704067240000
```

Cloudflare R2 (Media Storage)

```
bustracknow-bucket/
  profiles/
    userId1.jpg
    userId2.jpg
  routes/
    route45.jpg
    route12.jpg
  alerts/
    alert123_photo.jpg
    alert456_photo.jpg
```

Points Ecosystem (Detailed)

Earning Table

Action	Points	Frequency Limit	Conditions
Share location	+5/min	Max 30/min per trip	Distance to route < 50m
Confirm next stop	+2	Once per stop	After 2 stops minimum
Report “Full” bus	+10	Once per route/day	Verified by 2 users
Report “Late” bus	+8	Once per route/day	ETA deviation > 10 mins
Report “Not Running”	+15	Once per route/day	Verified by admin
3-day streak	+30	Weekly	Consecutive contribution days
7-day streak	+50	Monthly	Consecutive contribution days
30-day streak	+200	Quarterly	Consecutive contribution days
Refer friend	+100	Unlimited	Friend completes first trip
First contribution	+50	One-time	Welcome bonus

Redemption Logic

```
// In Wallet Screen
const minPoints = 10000; // 100
const userPoints = userData.points;
const redeemable = userPoints >= minPoints;

// Conversion
const cashAmount = Math.floor(userPoints / 100); // 100 points = 1

// UPI Verification Check
const upiVerified = userData.upiVerified || false;

// CTA State
if (!upiVerified) {
```

```

    // Show "Verify UPI First" button
} else if (userPoints < minPoints) {
    // Show disabled button + "Earn X more points"
} else {
    // Show active "Redeem X" button
}

```

Revenue Streams (From PDF)

Stream	Implementation	Target Revenue (Monthly)
Freemium	25/month Premium: Ad-free, advanced ETAs, offline maps	50k (2,000 users @ 2.5% conversion)
B2B	Dashboard access for transport departments	1.2L (8 cities @ 15k each)
Licensing		
Hyperlocal Ads	Banner ads near bus stops (geo-fenced)	30k (10 CPM, 3M impressions)
Data Insights	Anonymized movement patterns for urban planning	50k (project-based)
Total Potential		2.5L/month

Go-to-Market Strategy

Phase 1: Pilot (Month 1-2)

- City: Coimbatore
- Target: 5 engineering colleges
- Seed: 500 student ambassadors (2x points incentive)
- Marketing: QR code posters at 50 bus stops

Phase 2: Scale (Month 3-6)

- Add 3 more cities (Nagpur, Indore, Vijayawada)
- Partner with 2 municipal transport boards
- Launch referral program

Phase 3: Monetize (Month 7-12)

- Introduce Premium tier
- Pitch B2B dashboard to 10 cities

- Onboard local advertisers
-

Success Metrics

Metric	Target (6 Months)	Target (12 Months)
Active Users	15,000	50,000
Daily Contributions	3,000	10,000
Avg. Session Time	8 mins	12 mins
Redemption Rate	12%	18%
Cities Covered	4	12
Revenue	80k/month	2.5L/month

Final Checklist

- React Native project initialized
 - Firebase project created + SDKs integrated
 - Cloudflare R2 bucket configured
 - NativeWind + Tailwind setup complete
 - All 28 screens designed in Figma/Proto
 - Security rules deployed
 - Test users created (viewer + admin)
 - Route data seeded for pilot city
 - APK built and tested on low-end device
 - Play Store listing prepared
-

Total Development Time: 3-4 weeks

Total Screens: 28 (including modals)

Tech Stack Cost: 0/month (free tiers)

Scalability: 10,000+ users on free tier