# Product Requirements Document (PRD) - Atlanta FIFA Navigator

| **Key Detail** | **Value** |
| --- | --- |
| **Product Name** | Atlanta FIFA Navigator |
| **Version** | 1.0 Draft |
| **Author** | TyrannosaurusTech Consulting |
| **Date** | October 13, 2025 |
| **Status** | Draft for Review |

## 1. Introduction and Goal

### 1.1 Project Context

The FIFA Atlanta Games are expected to bring a massive influx of national and international visitors to the city. Atlanta is known for its complex road networks and significant traffic congestion, especially around major event venues. Visitors, unfamiliar with the MARTA transit system or local parking regulations, risk a frustrating and time-consuming experience, detracting from the overall enjoyment of the games.

### 1.2 Product Goal

To provide first-time visitors to Atlanta for the FIFA Games with a seamless, real-time, multi-modal navigation and parking application that minimizes travel stress and maximizes time spent enjoying the event.

### 1.3 Success Criteria

Success will be measured by:

1. **User Adoption:** of event-day attendees download and open the app at least once.
2. **Time Savings:** Average reported travel time to and from venues is reduced by compared to standard navigation apps during peak traffic periods (requires user feedback/telemetry).
3. **Parking Satisfaction:** User-rated satisfaction with the parking feature is or higher.

## 2. Target Users and Use Cases

### 2.1 Target User

The primary target user is the **FIFA Games Visitor**, defined as:

* A user unfamiliar with Atlanta’s infrastructure (MARTA, road layouts, parking zones).
* Focused on reaching a specific venue (stadium, fan zone) on time.
* Seeking the fastest and least stressful travel method (driving, ride-share, transit).

### 2.2 Key User Stories

| **ID** | **User Story** | **Priority** |
| --- | --- | --- |
| US-001 | As a visitor, I want to see the real-time traffic conditions on my driving route so I can decide if I should take transit instead. | P1 (Must Have) |
| US-002 | As a visitor, I want to find the nearest MARTA rail station to my current location and see exactly when the next train will arrive. | P1 (Must Have) |
| US-003 | As a visitor, I want to see the real-time location of the MARTA buses on my route so I don't miss the connection to the stadium. | P1 (Must Have) |
| US-004 | As a driver, I want to quickly find available parking spots near the venue, including their cost and walking distance. | P1 (Must Have) |
| US-005 | As a driver, I want to be able to pay for parking directly through the app (if feasible with ParkMobile API) so I don't need a separate application. | P2 (Should Have) |

## 3. Product Features and Requirements

### 3.1 Multi-Modal Navigation View (P1)

The application will feature a map-centric home screen displaying user location and primary points of interest (stadiums, fan zones).

* **R-3.1.1 Real-Time Traffic:** Overlay the Google Maps Traffic Layer on all driving routes to show congestion levels.
* **R-3.1.2 Transit Overlays:** Display the Google Maps Transit Layer, showing static MARTA routes (rail and bus lines).
* **R-3.1.3 Route Planning:** Allow users to input a start and end destination (defaulting the end to the nearest FIFA venue or a specified venue) and receive optimized routes for: Driving, Transit, and Walking.

### 3.2 MARTA Real-Time Data Integration (P1)

The app must consume and display live data from the MARTA APIs.

* **R-3.2.1 Rail Real-Time:** Integrate with the MARTA Rail Realtime API to show the next scheduled arrival time (ETA) for trains at specific stations.
* **R-3.2.2 Bus Real-Time:** Integrate with the MARTA Bus Realtime API to show the current location and direction of buses on selected routes.
* **R-3.2.3 Dynamic Route Update:** Transit routes must dynamically update ETA based on the real-time data from the MARTA APIs.

### 3.3 Parking and Payment (P1/P2)

The application must leverage the ParkMobile API for parking solutions.

* **R-3.3.1 Parking Spot Discovery (P1):** Use the ParkMobile API to locate nearby parking zones, display their zone ID, and current availability status (if available).
* **R-3.3.2 Pricing and Time Limits (P1):** Display associated costs and maximum time limits for each ParkMobile zone.
* **R-3.3.3 API Research Spike (P1 - Initial Phase):** A dedicated technical spike must be conducted to determine if the ParkMobile API supports **in-app payment transactions**.
  + *If Payment is supported:* Implement full in-app payment functionality (R-3.3.4).
  + *If Payment is not supported:* Provide a deep-link/URL to the official ParkMobile app or web checkout for the identified zone ID.

### 3.4 User Experience (UX) (P1)

* **R-3.4.1 Clear UI:** Use high-contrast colors and large, legible text suitable for quick viewing outdoors.
* **R-3.4.2 Language Support:** Support English and Spanish at minimum (FIFA required).

## 4. Technical Specifications and API Integration

### 4.1 Platform

* Native Mobile Application (iOS and Android)

### 4.2 Required APIs and Status

| **API** | **Purpose** | **Status** | **Notes** |
| --- | --- | --- | --- |
| Google Maps Traffic Layer | Real-time road congestion visualization. | Confirmed | Standard Google Maps SDK integration. |
| Google Maps Transit Layer | Static transit route visualization. | Confirmed | Standard Google Maps SDK integration. |
| MARTA Bus Real-Time API | Live bus locations and ETAs. | Confirmed | Requires secure endpoint access. |
| MARTA Rail Real-Time API | Live rail ETAs at stations. | Confirmed | Requires secure endpoint access. |
| ParkMobile API | Parking zone discovery, pricing, and availability. | **Research Required** | See R-3.3.3. Integration depends on feasibility of payment functionality. |

### 4.3 Data Handling

The app is heavily reliant on external APIs. The system must implement robust caching strategies (especially for MARTA and ParkMobile static data) and error handling to gracefully manage API downtime or slow response times, ensuring the user still receives the last known good data.

## 5. Next Steps

1. **Finalize Scope:** Review and approve this PRD with FIFA stakeholders.
2. **Access & Credentialing:** TyrannosaurusTech requires API keys/credentials for all listed services (MARTA, ParkMobile, Google Maps).
3. **ParkMobile Spike:** Initiate R-3.3.3 research spike to determine the feasibility of in-app parking payment.
4. **Wireframing:** Begin high-fidelity wireframing and design based on the approved feature set.
5. **Technical Architecture:** Design the serverless backend infrastructure required to securely proxy and manage the real-time MARTA API data.