Project Proposal for the Advanced Mobile Programming course:
ParkirajBa – A Smart Urban Parking Assistant by Faris Selimović

Urban parking in cities like Sarajevo is becoming a larger issue by the day both for residents and tourists. Traffic congestion, lack of real-time parking information, and inefficient use of parking spaces often result in wasted time, increased fuel consumption, and it is bad for the environment. While many cities are slowly adopting digital parking solutions, Sarajevo currently lacks a centralized, real-time, user-friendly mobile platform that allows individuals to locate and reserve parking spots. To address this problem, this project proposes the development of *ParkirajBa (parkiraj.ba, domain available :))*, a mobile application built with React Native, Firebase, and the Google Maps API. The application aims to offer real-time information on parking availability across Sarajevo, allow users to reserve parking spaces, and ultimately reduce congestion and inefficiencies caused by the lack of a modern parking systems.

The concept behind ParkirajBa is the combination of geolocation services, a searchable interactive map, and a reservation system that promotes better planning and space utilization. When a user opens the application, a Google Map is automatically displayed, centered around the user's current GPS location. Nearby parking facilities - defined as garages, private lots, or designated street parking zones - are shown as clickable pins on the map. Each of these parking locations is treated as a separate data entity in the system, containing comprehensive information such as the facility's name, a photograph or logo, the street address, working hours, contact details, and live parking statistics. Users are also given the ability to search for any address within Sarajevo via a search bar. When a new location is entered, the map navigates to that area and refreshes to show the closest available parking facilities within the vicinity.

What sets ParkirajBa apart from other basic parking map tools is the rich level of detail provided for each location. Tapping on a parking pin expands a dedicated information panel that shows the current total capacity of the facility, as well as a breakdown of available, occupied, and reserved parking spots. Furthermore, parking spaces within each location are categorized by type to support different vehicle and user needs. Categories include regular spots for standard vehicles, designated spaces for persons with disabilities, electric vehicle (EV) charging points, and spaces for rideshare vehicles such as e-GO in Sarajevo. Additional types supported include motorcycle parking, delivery and pickup zones and much more. This categorization supports accessibility and improves user satisfaction by ensuring the right type of space is available for each user scenario.

A major functional component of the app is its smart reservation system. Users who wish to secure a parking space in advance can do so by selecting the desired time and the type

of parking spot they need. The system then automatically assigns the user an appropriate available spot, updating the parking facility's database in real time. A timer begins from the moment of reservation, and the user is expected to arrive at the facility and occupy their spot within the assigned window. If the user fails to arrive on time, the reserved spot is released back into the pool of available spaces, and the user receives a penalty. This penalty system is designed to discourage misuse of the reservation feature and to ensure the integrity of the availability data shown to all users.

Authentication will be handled via Firebase Authentication, allowing users to sign in securely using their email and password or via integrated Google Sign-In. Each user has a profile that stores reservation history, penalty status, and preferences. The backend logic and data storage are powered by Firebase Firestore.

From a technical perspective, the app is built using the React Native framework and Expo for cross-platform development on Android and iOS. It employs Google Maps API for real-time map rendering and geolocation services, and the Places API for the search functionality.

The Firestore database is structured around several collections. The primary ones include users, parkings, spots, and reservations. Each user document stores personal information, penalty count, and a history of reservations. Each parking document stores metadata such as name, location, working hours, capacity, and a categorized count of different spot types. Under each parking, a subcollection of spots defines each individual parking space, its type, status (available, reserved, occupied), and any reservation currently attached to it. The reservations collection is used to track and log each booking attempt, with fields such as reservation time, user ID, and status.

In addition to core functionality, there are opportunities to expand ParkirajBa with advanced features in the future. For example, payment processing can be integrated to allow paid reservations, or dynamic pricing models can be introduced based on time of day and demand. Push notifications can be added to remind users of upcoming reservations or to notify them of penalty status changes. With appropriate partnerships, the app can also integrate live camera feeds or gate access control systems for automated entry. Furthermore, an administrative interface can be developed for parking facility managers to manage their listings, adjust working hours, and monitor real-time usage.

In conclusion, ParkirajBa offers a practical, real-world solution to one of the most common urban issues in Sarajevo: parking. Not only will it demonstrate technical proficiency, but it also addresses a real community need - making it a project that is both academically rigorous and socially meaningful.