Requirement Document for Tourist Information Application

1. Overview

The Tourist Information Application allows users to log in using a QR code, view their current location on a map, and access information about nearby tourist places. Users can either read the information or listen to it.

2. Features

2.1 User Login

- **QR Code Login**: Users log in to the application by scanning a QR code. This QR code will be unique to each user session.

2.2 Map View

- **Current Location**: Once logged in, the application displays the user's current location on a map.
- **Nearby Tourist Places**: The map highlights and pins nearby tourist places of interest.
- **Pin Interaction**: Users can click on a pin to access more information about the place.

2.3 Tourist Place Information

- **Read Option**:
    - **Icon**: Book icon.
    - **Action**: Opens a new page displaying the summary of the place sourced from Wikipedia.
    - **Navigation**: The new page includes a back option to return to the main app.
- **Hear Option**:
    - **Icon**: Ear icon.
    - **Action**: Initiates an audio playback of the summary of the place from Wikipedia.
    - **Controls**: Includes options for play, pause, and stop.

3. Functional Requirements

3.1 QR Code Login

- **QR Code Scanning**: Implement QR code scanning functionality using the device camera.
- **Session Management**: Ensure each QR code is tied to a unique user session.
- **Authentication**: Verify user credentials upon scanning the QR code.

3.2 Map Integration

- **Location Services**: Integrate with the device's location services to get the current location.
- **Map API**: Use a map service API (e.g., Google Maps, Mapbox) to display the map and pins.
- **Pin Management**: Dynamically place pins on the map for nearby tourist places.

3.3 Information Display

- **Wikipedia API Integration**: Fetch summaries of tourist places from Wikipedia.
- **Read Option**:
    - **UI Design**: Design a new page layout to display the summary text.
    - **Navigation**: Implement a back button for returning to the main app.
- **Hear Option**:
    - **Text-to-Speech**: Utilize text-to-speech functionality to read out the summary.
    - **Audio Controls**: Provide play, pause, and stop controls for the audio.

4. Non-Functional Requirements

4.1 Performance

- **Response Time**: The app should respond to user actions within 2 seconds.
- **Load Handling**: The app should handle at least 1000 concurrent users without performance degradation.

4.2 Usability

- **Intuitive Interface**: Ensure the interface is user-friendly and intuitive.
- **Accessibility**: The app should be accessible to users with disabilities, including support for screen readers.

4.3 Security

- **Data Protection**: Ensure user data, including location information, is securely stored and transmitted.
- **Authentication**: Implement secure authentication mechanisms for QR code login.

5. Technical Stack

5.1 Frontend

- **Framework**: React Native for cross-platform mobile application development.
- **UI Library**: Use native components and libraries for map integration and QR code scanning.

5.2 Backend

- **Server**: Node.js with Express for handling API requests.
- **Database**: MongoDB for storing user session data and other necessary information.
- **APIs**: Integration with third-party APIs like Google Maps and Wikipedia.

6. Milestones

6.1 Phase 1: QR Code Login

- Implement QR code scanning and authentication.
- Develop session management and user verification.

6.2 Phase 2: Map and Location

- Integrate map service and display current location.
- Highlight and pin nearby tourist places.

6.3 Phase 3: Information Access

- Integrate Wikipedia API for fetching summaries.
- Implement read option with navigation.
- Implement hear option with audio controls.

7. Acceptance Criteria

- Users can log in using a QR code.
- The map displays the user's current location and nearby tourist places.
- Clicking a pin shows options to read or hear about the place.
- The read option displays the summary in a new page.
- The hear option provides audio playback with controls.

8. Glossary

- **QR Code**: A machine-readable code consisting of an array of black and white squares, typically used for storing URLs or other information.
- **Text-to-Speech**: Technology that converts written text into spoken voice output.

Feel free to modify this document as per your specific needs.