**(Phase 4)**

**IOT- Noise Pollution Monitoring**

**System**

**by**

**RAJARAJAN R (aut2271360229)**

**A Real-Time Mobile Application for Noise Monitoring**

**Assignment Overview:**

The goal of this Assignment is to design and develop mobile applications for iOS and Android platforms that allow users to access real-time noise level updates. These apps will provide users with valuable information to help them monitor noise pollution in their surroundings.

**Key Features and Functionalities:**

1. **Real-Time Noise Monitoring**: The app will use the device's microphone to capture noise levels and provide real-time updates.
2. **Noise Level Alerts**: Users can set noise level thresholds and receive notifications when noise levels exceed those thresholds.
3. **Historical Data**: The app will store historical noise level data and allow users to view charts and trends over time.
4. **Location-Based Data**: Users can view noise levels in their current location or choose different locations on a map.
5. **User Profiles**: Users can create profiles to customize settings and track their personal noise exposure over time.
6. **Community Reporting**: Users can report noise disturbances, which will be shared with other users in the area.
7. **Data Privacy and Permissions**: Ensure that the app requests user permissions for microphone access and location data and complies with data privacy regulations.

**Backend:**

* Develop a backend server to store historical data and facilitate user notifications.
* Use RESTful APIs for communication between the mobile apps and the server.
* Ensure data security and user authentication.

**Database:**

* Utilize a database system (e.g., MySQL, PostgreSQL) to store historical noise level data.

**Real-Time Updates:**

* Implement Web Sockets or other real-time communication protocols to update noise levels in real-time.

**Mapping:**

* Integrate mapping and geolocation services (e.g., Google Maps API) for location-based noise data.

**Notifications:**

* Implement push notifications to alert users when noise levels exceed their defined thresholds.

**Data Privacy and Security:**

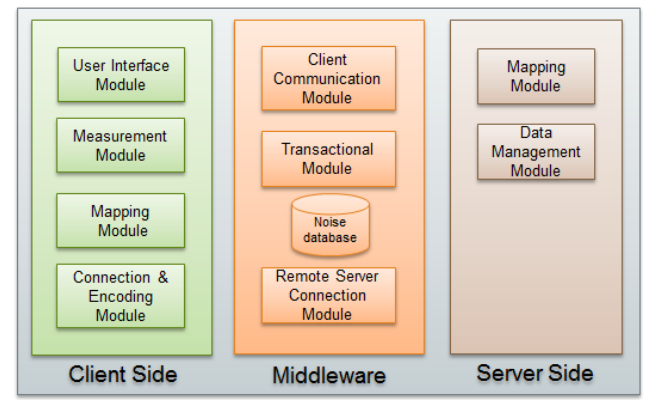
* Comply with GDPR and other relevant data privacy regulations.
* Encrypt data transmission and storage.
* Regularly update security protocols.

**Architecture and Prototypes:**

**General Architecture,**

This section describes in detail the conceptual architecture we used. the general architecture of the project. The prototypes in-development for this project follows the same schema. As seen, the modules are divided in three parts: Mobile Client-side application, Middleware Layer and Remote Server-side that are described below these lines.

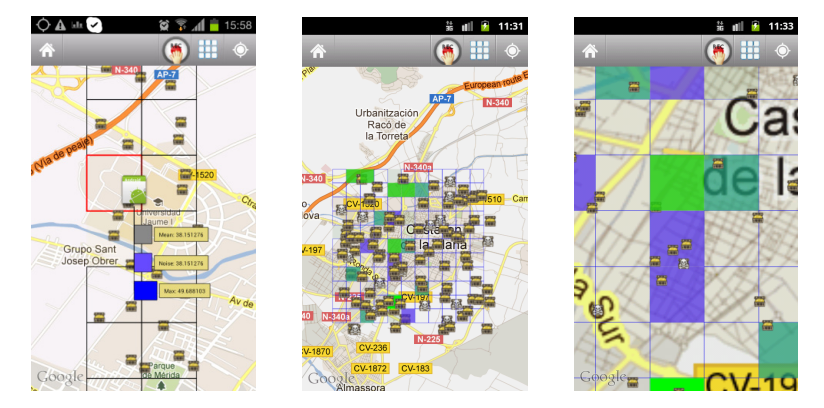
The Mobile application provides the functionality that allows user to take measurements, send them to the middleware and check the general progress of the battle in course. We suggest that the application should be divided in the following modules: User Interface, Measurement, Mapping and Connection & Encoding. For the User Interface we used Android SDK because it is a free and growing development platform and provides a good integration with Google Maps, the engine we used for our Mapping module.

****

**General architecture of the proposed solution**

**Noise Battle Prototype**

Noise Battle is a gamified mobile application developed for Android platform where users have to move around a city taking noise samples. Currently the game is on development and it is not entirely functional or available to download. The final goal of this game from the user’s point of view is conquering the city by taking noise measurements. The city is split into cells of a grid, so the user can conquer the cells by taking more and better measurements than other users in the area. During the game, the users are rewarded by different means: either by measuring in the proximity of one of the rewards placed in the grid or by conquering some cell of it. The rewards might include the possibility of sending noises or sounds to the foes. The sounds that can be sent to the rivals are used to show the power obtained by the sender of the noise. The rivals have the option of re-conquering previously conquered cells by performing better quality measurements or more recent measurements. These mechanisms should encourage the players to provide more accurate (based on a criterion taken from the API used) or updated noise data. Regarding the rewards, we have considered awarding measurements dispersion and data quality (based on Noise Droid quality assessment) in order to assure a regular number of observations taken in the entire city area. Besides, we also took into account to place rewards (or having higher density of them) in zones where there is more interest about noise pollution conditions. The observations gathered are immediately uploaded to the middleware server, which further submit it to the ONMP platform. The ONMP platform can store it and make it available for visualization or processing. In Figure 2 (left) it is possible to see a sample of the battlefield where a game is going to start. As seen, the city is divided in a grid where each cell represents an area the user can conquer through taking noise measures. For each measurement performed, it is possible to see the minimum, maximum and mean of noise measured. Then the player can decide to submit it as his move in the game. In this case there are some items placed for each cell that represent locations where the player can take a noise measurement and receive a reward. Figure 2 (centre) represents a more advanced state of the battle for that city, where some users have started using the application and conquered some areas. The areas conquered can be recognized by the colour of the cell. In this scenario there are three players, each of them with a different colour and avatar associated competing each other. Figure 2 (right) provides a higher level of detail of the image in the centre. In the next section we briefly describe the second prototype.



**Some images depicting application prototype working**

**Conclusion**

The Real-Time Noise Level Monitoring Apps for iOS and Android are intended to provide users with essential tools for monitoring noise levels in their environment. This project report outlines the key steps, features, and considerations for successful app development. By following these guidelines, we aim to deliver user-friendly and reliable apps that enhance users' awareness of noise pollution and its impact on their lives.