Vinay Jude
Nirupan Maruthappan
CSE 646 Initial Report

UB Noise Mapping

A noise map can be defined as a graphic representation of the sound level distribution existing in a specific region.

The ***Environmental Noise Directive*** of the European Parliament and Council Directive, names a few definitions of noise mapping, as follows:

Environmental Noise: Unwanted or harmful outdoor sound created by human activities, including noise emitted by means of transport, road traffic, rail traffic, air traffic and sites of industrial activity.

Noise Indicator: A physical scale for the description of environmental noise, that has a relationship with a harmful effect.

Strategic Noise Map: A map designed for the global assessment of noise exposure in a given area due to different noise sources or for overall predictions for such an area.

Action plans: Plans designed to manage noise issues and effects, including noise reduction if necessary.

Acoustical Planning: Controlling future noise by planned measures, such as land-use planning, systems engineering for traffic, traffic planning, abatement by sound insulation measures.

Current Problem

Currently, noise maps are created using high cost noise meters that measure high accuracy data at predefined locations. This data is extrapolated using landscape models and simulation tools. Even with the use of accurate landscape models and simulations this leads to inaccurate maps due to the small number of data points and missing noise sources. Increasing the data points using more noise meters and more man power is too expensive and does not scale.

Vinay Jude
Nirupan Maruthappan
CSE 646 Initial Report

Proposed System

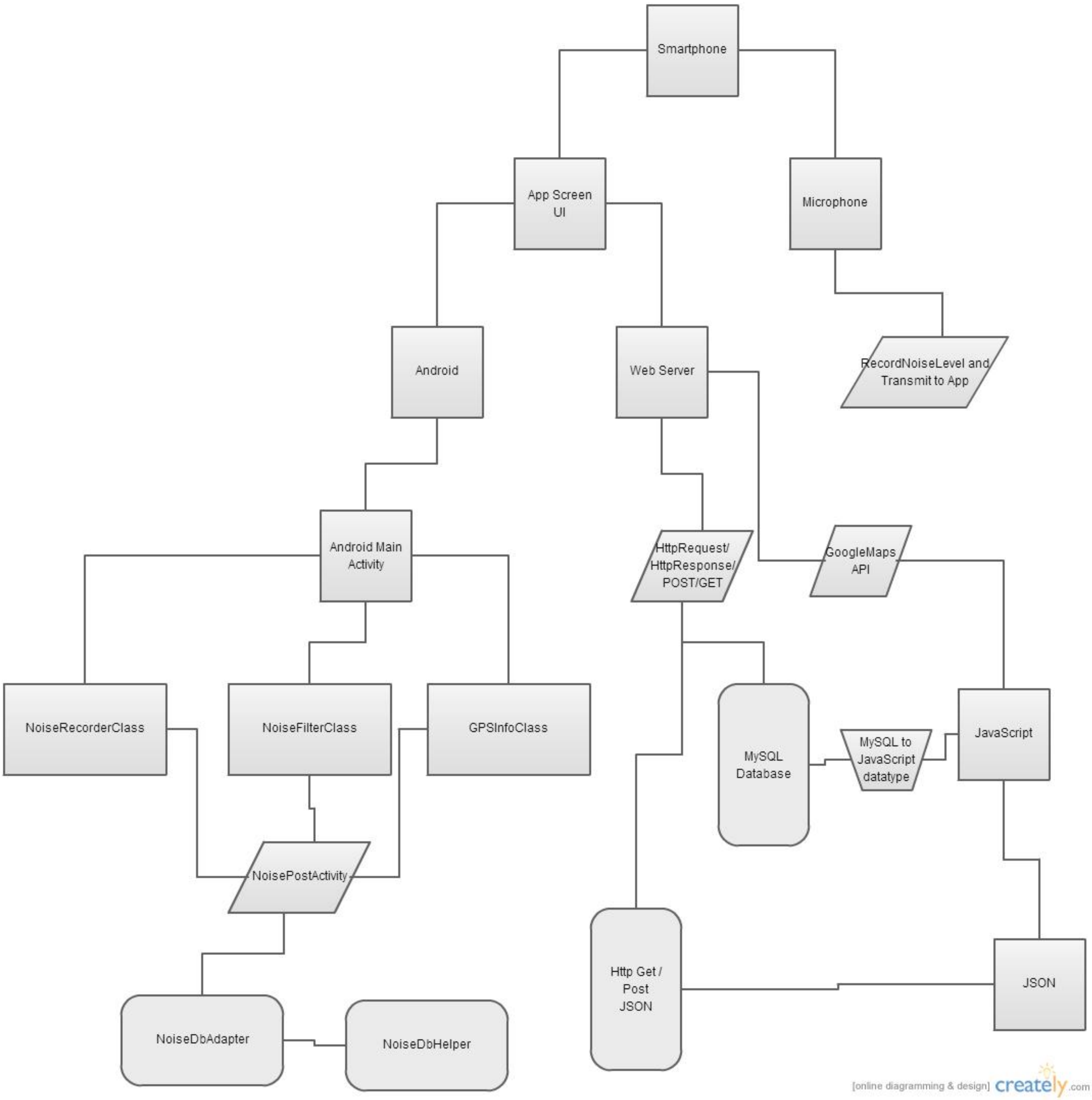
These days, almost everyone uses a smartphone that are equipped with a microphone and GPS transceiver. Thus the smartphone can be used as a noise meter by combining the number of potential sensors to a participatory network that allows for the creation of high density real time noise maps. This system samples the incoming sound to translate the discrete signal to a dB full scale(dbFS) value. On this scale '***0 dBFS'*** stands for the maximum level the microphone can measure and all other ***dBFS*** values are negative. This value has to be translated to ***dBSPL*** where ***SPL*** stands for sound pressure level. ***SPL*** is a reference system relative to a given sound pressure value. This value is usually ***20µPA*** which is considered as the threshold of human hearing. To translate ***dBFS*** to ***dBSPL***, a calibrated value has to be added so $dBSPL = dBFS+x_{cal}$ .. Given a constant pink noise x_{cal} is calculated by the app.

Components Used:

- A simple android smartphone with built-in app and the usual microphone.
- A fully functional android app which is able to handle several devices at the same time and efficiently send the data to the webserver database.
- A webserver that handles all these requests and obtains the information from the database uploaded from the smartphone network.
- An interactive Google maps extension that displays the collected data assimilated by the webserver.

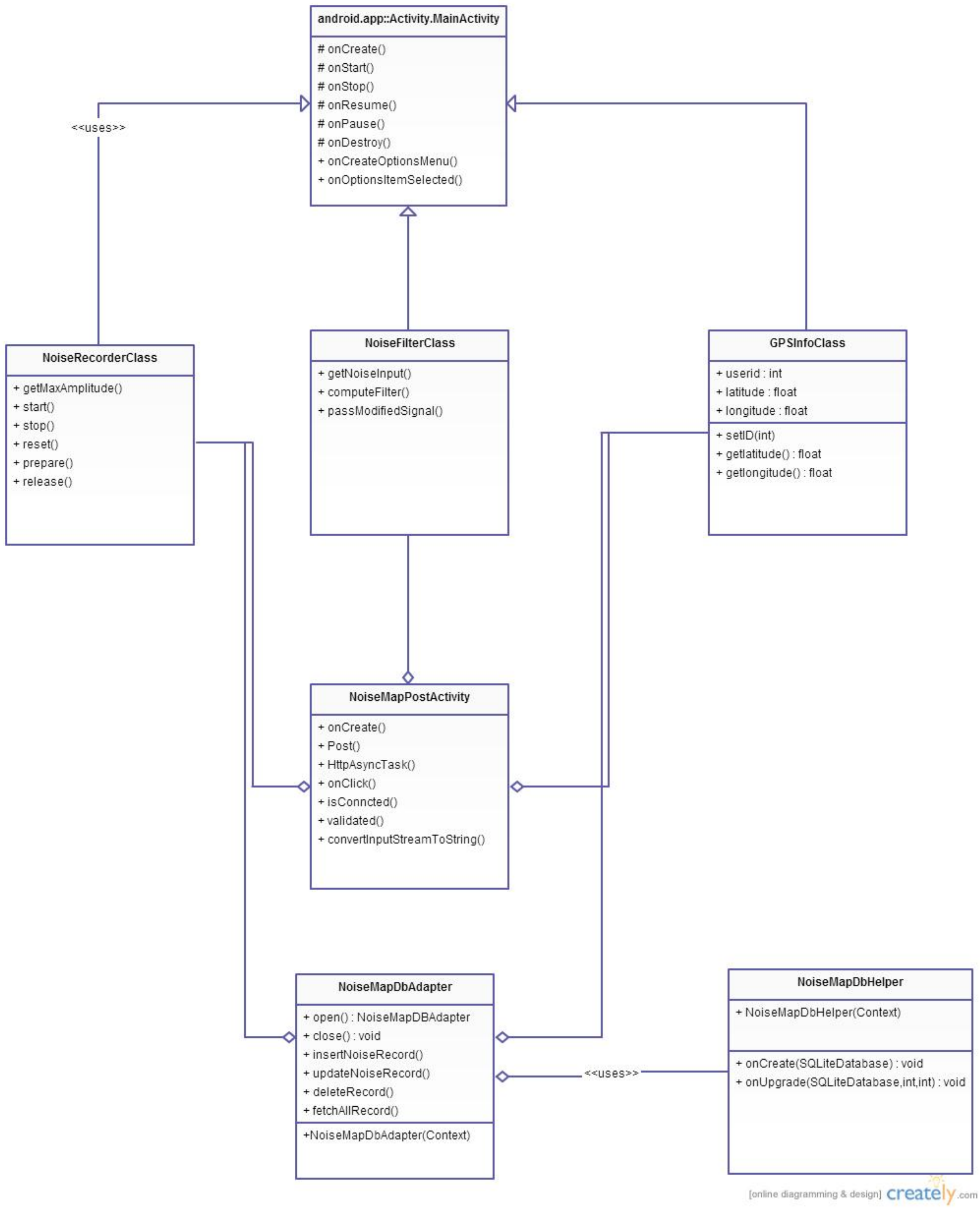
Vinay Jude
Nirupan Maruthappan
CSE 646 Initial Report

Components Block Diagram



Vinay Jude
Nirupan Maruthappan
CSE 646 Initial Report

UML Class Diagram for Android Code



Vinay Jude
Nirupan Maruthappan
CSE 646 Initial Report

UB Noise Mapping Timeline

February 2014 - March 2014

Vinay Jude
Nirupan Maruthappan
CSE 646 Initial Report

Integrating I and II

II) Server Initial Phase

Vinay Jude
Nirupan Maruthappan
CSE 646 Initial Report

March 2014 - May 2014

Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
30	31	April 1	2	3	4	5
			Integrating I and II			
6	7	8	9	10	11	12
	Integrating I and II				Google Map API	
13	14	15	16	17	18	19
		Google Map API			User Interface of I and II	
20	21	22	23	24	25	26
			User Interface of I and II			Testing
27	28	29	30	May 1	2	3
			Testing			
4	5	6	7	8	9	10
	Testing	Demo				

Vinay Jude
Nirupan Maruthappan
CSE 646 Initial Report

Timeline Itinerary

Requirements 2/10/2014 - 2/24/2014

- Information aggregation on noise recording, Google maps API, etc.
- Peruse existing systems.
- Decide features and functionalities needed for the proposed system.

Design 2/25/2014 - 3/4/2014

- Sketching the modules needed.
- Designing the components on Android source code.
- Decide on the web hosting framework.
- Finalizing on the programming and scripting modules.
- Synchronization of visualized UI with the design modules.

Implementation 3/5/2014 - 3/28/2014

I Android app Initial Phase 3/5/2014 - 3/17/2014

- Building the android modules for noise detection.
- Filtering mechanism as to what noise should be sent.
- Distinguishing different types of noise.

II Server Initial Phase 3/18/2014 - 3/28/2014

- Web Host handling scalable database.
- Parsing information received.
- Google Maps connectivity with sample database.

Integrating I and II 3/29/2014 - 4/8/2014

- Integrating the server with the android app.
- Building GET/POST requests management.

Google Map API 4/9/2014 - 4/17/2014

- Receiving coordinates and noise measurements.
- JavaScript codes for manipulating Google Maps.

User Interface of I and II 4/18/2014 - 4/25/2014

- Menu and Screen Configuration for Android App.
- Toggle Modes to start/stop.
- Web site UI integration.
- Interactive Noise Map with additional browsing features.

Testing 4/26/2014 - 5/5/2014

- Testing with sample devices.
- Verifying database information transmission.
- Checking the accuracy of the results on the web interface UI.
- Checking for scalability with multiple devices.

Demo 5/6/2014

Future Work

Frequency-dependent hearing of the human ear can be implemented. A signal of the same pressure is interpreted different for changing frequency levels. In order to normalize the value, A-weighting can be applied to the signal.

Conclusion

Thus we would like to create an effective noise mapping system for the UB campus to monitor the noise levels on the campus and for other monitoring purposes and use this a tool to depict the noise data collected in different representations.

Vinay Jude
Nirupan Maruthappan
CSE 646 Initial Report