# CSE 535 Project Proposal Phase I

## Project Members

* 1. Dhiraj Gurkhe (1209305002)
  2. Karumpudi Ramakrishna Reddy (1209319770)
  3. Sriram Vellangallor Subramanian (1209270383)

## Project Idea

We are planning to build an *Intelligent Voice Recorder* application on Android platform. The application will enable users to annotate the audio while recording, on the fly. This app will provide two functional modes - the recording and playback. While recording an audio, the user can view the waveform of the recording real time. We will implement a user interface through which he can insert a hook at different time frames to include textual annotations. These annotations can be input by keyboard. In the playback mode, user can view the waveform of the recorded audio as a timeline. Annotations would appear in sequence while the audio is being played. The user can organize his notes based on location fetched from the GPS, tags, time of recording or his own preferences.

## Project relevance to class

The project is a mobile (Android) app which has multiple features that are relevant to this course. The app will enable user to record audio on his mobile device, which will function as audio notes. It uses GPS to sense the location of the handset to enable him to organize his notes by location. In case the user was traveling while the note was being taken, it would suggest him the note as a trip name based on the starting and ending locations detected by the GPS. We will be leveraging on Android Media Recorder APIs to record sound and Android location services to detect his location. We would use storage mechanisms such as SQLite to store user preferences and SD card access to store the files.

## Plan to use external sensors

For the scope of this project, we do not require any external sensors.

## Type of platform being used

The app should work on Android phones with Android API Level 19 + (KitKat and above)

## Development Setup

IDE: Android Studio 1.3

Testing: Emulator (AVD), Personal Cell Phones.

## Specification of context aware behavior

* We plan to use the GPS to get the current location to enable user to organize his notes
* According to the time of the day when the user recorded an audio, the app will change UI/Background
* Based on the annotations provided in the middle of the audio recording, top three URL links will be fetched from google and provided to user for reference

## 15 tasks for the project (with the category – acquisition, delivery and reception, action)

1. Voice recorder: feature to record voice of the speaker
2. GPS tracker: This feature will track position of the device at the time of recording
3. Note hooking: inserting a hook in the middle of the audio
4. Waveform timeline: this is an audio waveform timeline (UI element)
5. UI adaptation: adapting to UI based on the timestamp.
6. Playback mode: including the waveform and playback controls on front end
7. Playback controls- Seeking along playback slider, pause, volume control
8. Folder organization I: organizing data based on the location
9. Folder organization II: organizing data based on the user preference/input
10. Folder organization III: organizing data based on time
11. Download audio: download plain audio file in .mp3 format
12. Visual spots: highlight the areas on the audio timeline where annotations are present
13. Registration: login/signup features will be provided
14. Background service: handling the scenarios when this app is pushed to background
15. Seeking to hooks displayed on waveform.

## Timeline for each task and member responsible

|  |  |  |  |
| --- | --- | --- | --- |
| Serial No | Task | Assignee | Deadline |
| 1 | Voice recorder | Dhiraj | 2/26/2016 |
| 2 | GPS tracker | Dhiraj | 2/26/2016 |
| 3 | Note hooking | Dhiraj | 2/26/2016 |
| 4 | Waveform timeline | Sriram | 3/4/2016 |
| 5 | UI adaptation | RK,Sriram | 3/4/2016 |
| 6 | Playback mode | Sriram | 3/4/2016 |
| 7 | Playback controls | Sriram | 4/15/2016 |
| 8 | Folder organization I | RK | 3/18/2016 |
| 9 | Folder organization II | RK | 3/18/2016 |
| 10 | Folder organization III | RK | 3/18/2016 |
| 11 | Download audio | Dhiraj | 3/25/2016 |
| 12 | Visual spots | Sriram | 3/25/2016 |
| 13 | Registration | RK | 3/25/2016 |
| 14 | Background service | Dhiraj | 4/8/2016 |
| 15 | Seeking to hooks | Sriram | 4/8/2016 |

15, Seeking function, Sophisticated implementation:

If we have sufficient hooks to seek back and the player has played sufficiently less (3.5 seconds), it will seek to the beginning of one complete hook.

If we have only one hook to seek back, it will seek to the beginning of current hook.

If playback of more than 3.5 seconds has completed from the current hook, it will seek to beginning of current hook.

If there are sufficient hooks to seek back and user wishes to seek to beginning of previous hook after 3.5 seconds of playback `into` current hook, he will first press seek back. Within 3.499 seconds he has to press seek back again. To put in simple terms, if user double taps `Seek back` button after completing 3.5 seconds, it will seek to beginning of previous hook and instead, on single tap it will seek to beginning of current hook.

Forward seek is simple. If user taps on seek ahead button, player will always seek to the beginning of the next hook.

Pending task in Playback mode:

Integrate module with RK’s folder view flow. Note: Currently the app will function only if you have a “lw” note *bundle* in “mc” folder within “Mydata” (of course).

## Additional tasks if time permits

Additional Idea: We intend to provide the user relevant links based on the context of his textual annotations, in a particular time frame.

1. Customization: user can customize the app to receive suggestion, auto record at certain time etc.
2. Prediction engine: recommend to start recording based on his location or event or calendar events
3. Auto recording: automatic start recording when user reaches particular location/ customized time
4. URL recommendation: recommending URL based on the annotation provided in the audio file (note taking part)
5. Notifications: pop ups will be provided in certain scenarios (like auto recording gets started)
6. Scribbling Text Input.