****

**Computer & Electronic Engineering**

**Final Year Project "Music Host Interface"**

**Sprint 4: Week 3 Log**

**Thomas Flynn**

**Project Supervisor: Brian O'Shea**

**01/02/16 - 07/02/16**

**Entry 1: 02/02/16**

Today I went for a walk and contemplated how I would tackle the issue of queuing songs on my GUI. My plan is to update my existing code which makes an array list of media objects at compile from all the songs in the temp folder. I want all the songs in the temp folder to appear on the GUI as a selection. And then through an "add to queue" button the song from the selection get's added to the array list of song/media objects. Then later add a "remove from queue" button.

**Entry 2: 04/02/16**

Tomorrow I plan to write some code for my Android Application. I plan to hardcode an SQLite database into the application that matches the one I have for my JavaFX GUI. The reasoning behind this is as follows

A.) This feature doesn't require you to pull the music host's database every time you want to request a song.

B.) You arrive to the music host at a later date and you perform a refresh that will compare your last snapshot of the music host database versus the current one. This feature would truly highlight the power of SQLlite.

C.) I don't really know how to do A or B and by saying that I'm attempting this feature I can get away with taking a short cut towards the Android app actually choosing and queuing a song for my demo at the end of the year.

**Entry 3: 05/02/16**

Using the pomodoro technique of working for 25 minutes and taking a break for 10 minutes, I accomplished a lot today for my Android App.. I created a 2nd activity and populated an SQLite database with hardcoded values. These values are then queried and displayed in a list view. What I select from the list view is sent back to the main activity in preparation for sending to the JavaFX music Host.

**Entry 4: 06/02/16**

I started working on debugging my server. When I added my server code from my FYP-Server repository to my FYP-GUI. I ran into a problem where I could only echo back one message to Android client.

**Entry 5: 07/02/16**

Today I managed to solve 2 major problems with my project.

1- I can successfully send messages over and back to the Android Client.

2- The FX GUI refreshes the screen using an animation clock object. Now the messages from the Android client are displayed in realtime on the GUI. Without the need for a refresh button that I was using earlier.

I also created a 'Song' class that will hold the Media Player object and artist/song name details. It took me some time to figure out I would relate the database info to the media player objects

**Research done:**

List view in android

SQLite Queries

JavaFX Animation clock

JavaFX threading in general

**Tasks completed:**

Create 2nd Activity

Create and read from SQLite database

Songs displaying in 2nd Activity

Each song is clickable and returning to main activity

Android and JavaFX Gui talking perfectly

FX GUI displaying Android messages in realtime.

Created a 'Song' class that will hold the Media Player object and artist/song name details

**GIT:**

**Repo: FYP-Android**

Commit #6 - [**added a button to main activity**](https://github.com/g00291875/FYP-Android/commit/83904fd2b010beb1fed7ff785ce6cefb86ae31c6)

Commit #7 - [**importing Mobile Devices 2nd lab exam code for activity 2**](https://github.com/g00291875/FYP-Android/commit/4ef0416bf53e7f29e1f57f9b5c2c374d24620a0b)

Commit #8 - [**reading from SQLite DB**](https://github.com/g00291875/FYP-Android/commit/0f300f8a700807b233949a415e353daaf5b49877)

Commit #9 - [**reading from db using 'useCursor()' method instead of raw query**](https://github.com/g00291875/FYP-Android/commit/85e8c16c43392c63c49e68c2a661f72a076b0052)

Commit #10 - [**songs from DB are clickable and returning to main activity**](https://github.com/g00291875/FYP-Android/commit/5928c1d084f5402f039b20adfe598913d38ed2f6)

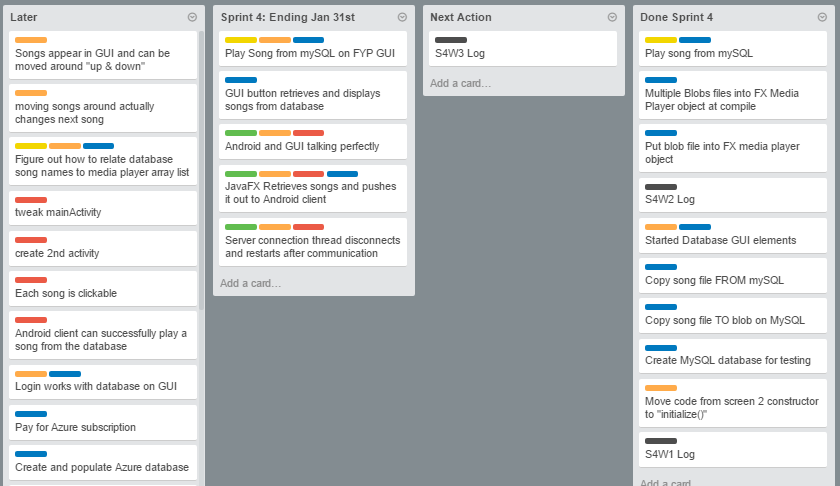
**Repo: FYP-GUI**

Commit #25 - [**GUI updating incoming msgs from Android client (without a refresh but**](https://github.com/g00291875/FYP-GUI/commit/9d6575d5da38975ce012a7b5fe8e50345b7f2d60)

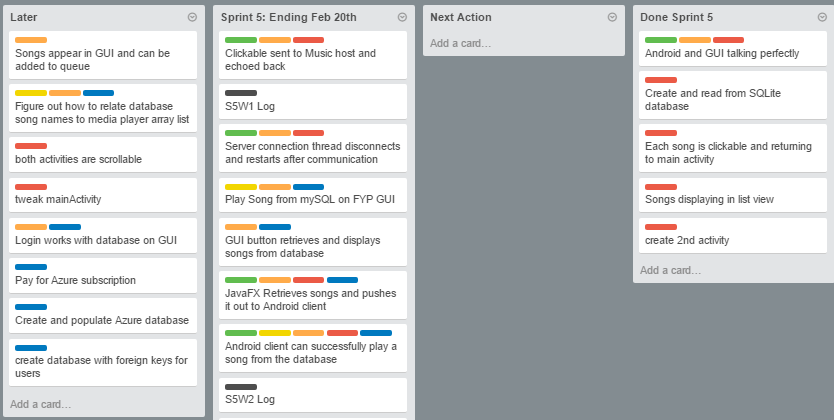
Commit #26 - [**JavaFX GUI and Android now talking perfectly**](https://github.com/g00291875/FYP-GUI/commit/6fecad914d2b47f0ae18dbefb0690d8df645b99e)

Commit #27 - [**created 'Song' class that holds mediaplayer object**](https://github.com/g00291875/FYP-GUI/commit/eacbdada06ed055cfdc3ee12dc8760ead9b0ec35)

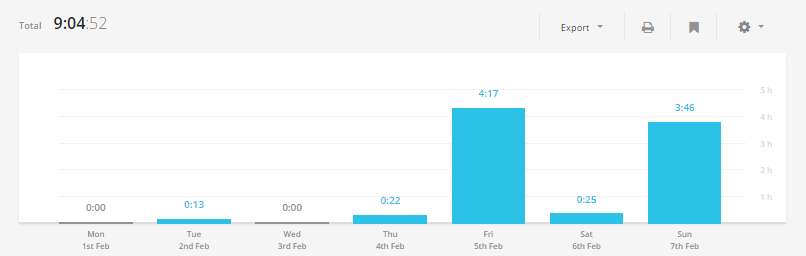
**Board at the start of the week:**

****

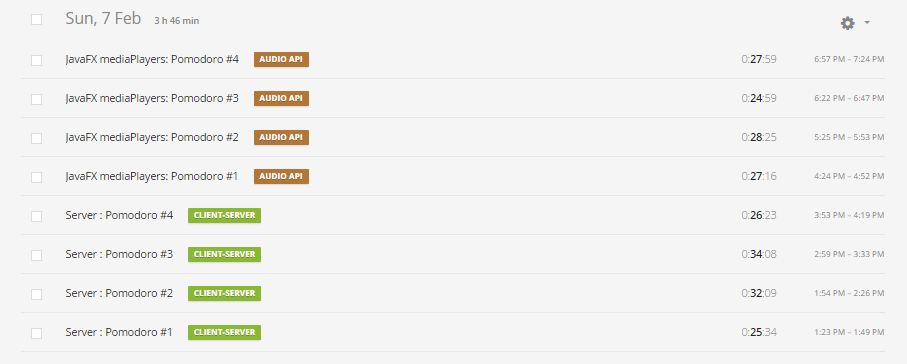
**Board at the end of the week:**

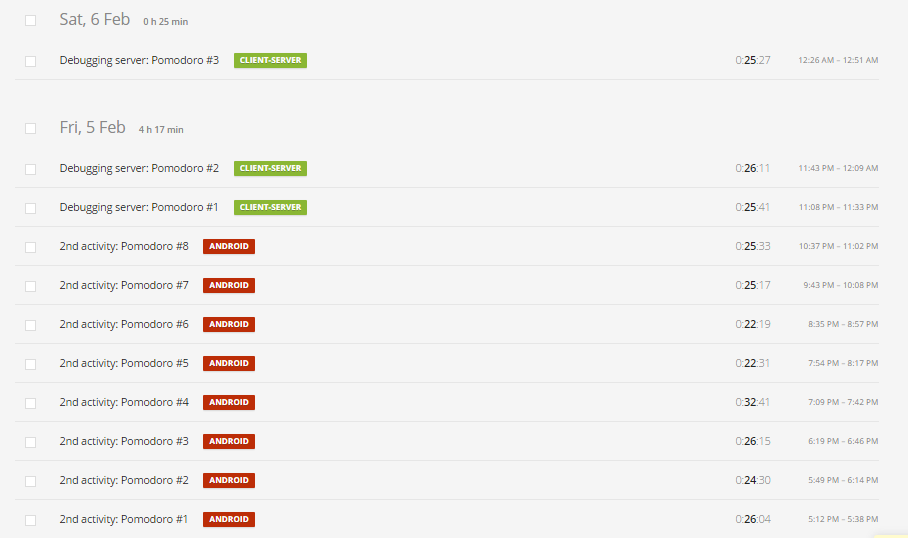
****

**Weekly time Log bar chart:**

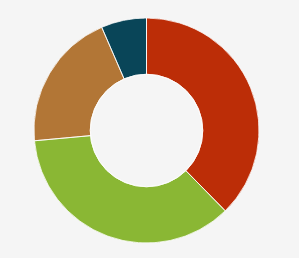
****

**Weekly Time Log:**

****

****

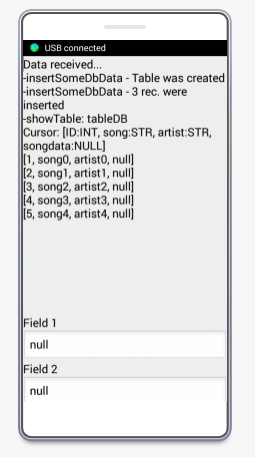
**Weekly log Pie Charts:**

****

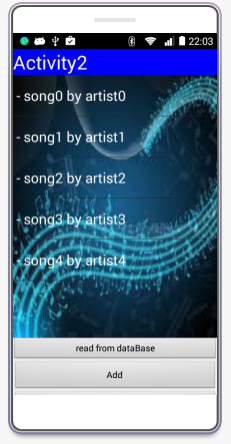
**Screenshots progress:**

**Android client:**

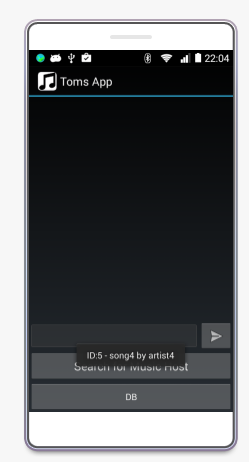
**Step 1:** SQLite database holding arguments and displaying it on 2nd Activity.

****

**Step 2:** Read values from SQLite database and put them in a ListView to show the user.

****

**Step 3:** After the user selects a song from the list, return selection back to main activity for sending to the FX GUI.

****

**FX GUI:**

FX GUI being updated (using animation timer) as soon as message from Android client arrives as well as being able to echo back messages without any problems.

Messages appear under song Rquests:

