3rd Year project Blogs
CA326
Andrew Cullen, Daniel Rowe, Eoin McKeever

Sprint 1 14/11/2019

For our first sprint of the project we had to submit a project proposal. The project we chose to do is a music recommender app that will link a user's Spotify account to their own account, and we will recommend events to them based on their preferences and their location. We are calling it Livewire

Sprint 2 20/11/2019

For this sprint we must complete a functional spec on our project and submit it for the 6th of December. The functional spec will consist of an overview and a summary of the overall project. It will also give a general description. In the description we will list and describe all the functions the app will have. We will also include a list of the scenarios that can possibly occur while in use of the app. After that we will include a section on constraints that can potentially affect how our project will turn out. In the spec we will discuss the requirements of the project and what we need to accomplish for the project to come together. The spec will conclude with data flow, user flow and class diagrams. We met with our supervisor to discuss the spec at our scheduled meeting and present a Ghatt chart.

Sprint 3 7/01/2020

Made no notable progress on the project due to exams. Requested to meet with the supervisor after Christmas exams to focus on studies.

Sprint 4 29/01/2020

For this sprint of our project we focus on developing a skeleton app for our scheduled meeting with our supervisor where we would show and discuss progress. We planned to have a full skeleton of the app ready for the meeting but with exams, modules and our inexperience with Android Studio we didn't meet our set goal. What this sprint consisted mostly of was research of android studio and how it operates i.e. looking into manifest files, xml language, activity, fragments and views. Come the scheduled meeting with our supervisor we had a logo designed for us by a media student and we turned this image into our app launcher icon. We had the app launch into a dummy sign in page with edit text

boxes with a login and signup buttons. As of now if pressed they bring the user onto the next fragment which is empty as of now, but we plan to make our homepage.

Sprint 5 12/02/2020

This sprint we focused on bulking out our skeleton app and implementing Firebase into our app to authenticate users at sign in. Our main aim was to have a traversable app with all the pages linked correctly. We planned to fill these pages once we had that completed. We made a toolbar that we planned to keep consistent across all the pages. We also chose a colour scheme to use across the app we chose red and black to match the logo. After that we had a go at linking all the pages together this took some investigation into activities, fragments and views and manifest files and how to use them in terms of android studio and with some time we got it working correctly. The pages are all going to link out from the main scrolling feed page the different options of pages to visit from here will be the user profile page, a scrolling page with related posts and a settings page. The next step we took was integrating Firebase into our Android Studio project, so we had a secure place to store our user's data. Android Studio makes it very convenient to link the two and there are lots of resources available online to walk you through the processes. We implemented a sign-up page to work with the Firebase database. Our plan for the next sprint of the project was to continue to improve the UI we had developed so far and to work at finding out how to implement the API to pull user data. We also need to find a way to create a recommendation based on this user data.

Sprint 6 26/02/2020

This sprint was our most productive to date. We managed to further improve our UI by adding a Settings page. The settings page is found on the navigation bar at the bottom of the screen. When visiting the option to link a user personal Spotify account is now available and to let Livewire know your location so it can recommend events around a user local area. So now with Spotify linked we can now pull a user's Spotify data to feed our recommender code. We made use of Firebases cloud functions this sprint also. We used Firebases cloud functions to pull listings from the newly linked Ticketmaster API. So now we have listings of events from our Ticketmaster API data and we can filter this data by feeding the users Spotify data into our event recommender. Now that we have data to put in our scroller pages we decided to load it into the page, but we were met with a bug. The app became overloaded with data and crashed. We investigated this problem and developed a recycler view for the scroller page. This stopped the app crashing and loading into much data. Also, ethics was submitted and approved. We held our user tests. Unfortunately, we only managed to get 4 participants.