CMP3060 Project Proposal
Assessment 1

Mobile Computing: Movie search

Background

How many times have you been having a convocation about films and they suggest you watch a great film such as Space jam, and by the time you get round to it you have forgotten the name!

This is the main reason for developing this idea into an application implementation. The Application that I have developed allows a user to search for movies, view top movies, find nearby cinemas based on location and search for the top movies at current.

From this, the theme that I picked for my application is for arts, culture and creative industries. My app allows users to explore new genres of movies, the application allows them to expand upon what they enjoy and get out of their comfort zone and discover movies that they may enjoy.

Also my application has a social aspect to it, if a friend suggests a film to watch, you can save it in the app so that you can watch it another time, and so that you don't forget what it is. This is good for convocation since you can watch the film ready for next time you meet and discuss your thoughts on the movie.

My main motivation for making this application for finding and saving movies was that, there are other apps out there like IMDB and the cine world app. They have a lot of extra features that can overwhelm users, my development is the basics of what these apps can do illustrated in a simple way. The reason for this idea coming into development was that, I have a friend that suggested the idea to me when I asked him what he would do if he could make an app. My friend studies film and media, he thought that this app will be useful to him for tracking movies that lecturers and demonstrators suggest he watches, but by the time he could, he would have forgotten the name. Another reason for this being my developed app, is that I have a great interest in films myself, I studied film in 6th form and always find it interesting to search up movies and actors to see what other roles they have been in.

As mentioned previously a few other applications that I was inspired by was IMDB (IMDb, 2016), Cine world (Cineworld.co.uk, 2016) and Odeon (Odeon.co.uk, 2016) apps. They all have some good unique features such as Odeon having a cinema finder, and IMDB letting you search all movies, not just those that where currently on in cinemas. One of the main goals for my app is to implement all of these features into one easy to use app.

The positive impact on society my application will have is that it can make society more social when it comes to movies and the different stories they may have. An example of this is that if a friend tells me about a movie that I should watch, I save it with the app, this then allows me to re visit it at another time, the movie can then become a conversation topic to talk about with that friend at a later date once you have watched the film. This then leads to a much more interesting convocation rather than "I forgot what the name was so I couldn't watch it".

Implementation

Core development choices

Chosen API

The API that I chose for the implementation for my application was going to be Rotten Tomatoes (Developer.rottentomatoes.com, 2016) but I had some limitations with this. One of those being that, Rotten Tomatoes had a long response time for the API key, up to 60 days, and I didn't have that long to wait for the key (I don't think I ever did get a reply from them). Another difficulty with this API is that there wasn't much development support for it, the website had a few sections on code and that was it, and not many applications where developed with this API.

Another API that I could have used was the OMDB (Omdbapi.com, 2016). This API had a lot of support for its developers but the key was locked behind a donate button, I would have to have paid for the key.

Finally, I found Movie DB (Themoviedb.org, 2016), an API that was openly free and had a lot of developer support and forums for questions. The responses from the API was in the proper JSON format, and was easy to include into my application.

Difficulties

One of the first difficulties that presented itself was with my searching for a film activity. A problem that I was having with it is that once the search had been done, passed through the JSON code, then the user can click it and view more details. The on click listener that I made originally did not work properly and crashed the app. The reason for the crash was that the code wasn't in the on create, so it didn't have a start point, and on click doesn't work with list views, this also crashes it.

Another problem that I had with the search is passing movie details to the next activity, I tried many methods but most of them didn't work or pass the data, in the end I found a way where I would make a bundle of data, as separate strings, pass the bundle to the other activity, and then unload the bundle into new strings. This data is then passed to the appropriate text view/image view.

Another problem I had with this was displaying an image from a JSON URL, I tried to implement it myself but the code was too complex to do, so I used a library called Picasso (Square.github.io, 2016). Picasso allows for images to be displayed on the app, taking it straight from a URL link.

Another problem was that some of the movie descriptions would be too big for the space that they were given, this was solved by putting the text view inside of a vertical scrollable view, this ensured that all data was then displayed in the right place, without covering up anything else.

An error that I couldn't fix in my program was that if a user was to search for something, and then search again without opening a film, the second search would be added to the bottom of the list, rather than erase the list and start fresh. I have an idea that my method of passing data to another activity has effected the search, as in previous iterations the second search worked.

The final difficulty that I had was saving strings locally, an issue that I had was making the string saved into the file into an array, my way of getting around this was by separating each saved item with a comma and then making an array based upon where those commas are in the string.

The sensor that my application used was location, this was used to let users see cinemas that are nearby based on where they are geographically. The way that this works is the app takes the users location, opens google maps and searches for cinemas. The reason for using google maps rather than open the map in my own activity is that, google maps is the most popular app on android, the app is installed on around 80% of devices.

Another feature of my application was local storage, the method that I used was using internal storage. Movie names would be saved to a file, this is because the main feature of the application is to save films for later viewing, and it is important that they are still there once the application has been closed.

I made use of some of the code from the workshops; location, REST API and Local Storage.

I adapted and developed these topic areas into my finished code, location being that I simplified the code itself down to the basics. A problem that I had with doing this is that there is no error handling, so if the location sensor is turned off or google maps isn't installed the app just crashes, I don't think that this is much of an issue, but if I was to carry on developing this app then I would implement these fixes.

As for the REST API workshop I made use of the JSON class, the reason for this was because this was a simple way of reading in the JSON, and didn't really need much extra development. When it came to assembling the JSON data then changes were made, different values were read in such as title, release date, image URL and an overview. They were then assigned to their corresponding arrays ready for the user to select a film.

Local storage had some developments, I made it so that the data was converted to an array, mentioned previously.

Resources and References

Cineworld.co.uk. (2016). Latest Movies - New Films - 3D Movies | Cineworld Cinemas. [online] Available at: https://www.cineworld.co.uk/ [Accessed 10/12/2016].

Developer.rottentomatoes.com. (2016). Fandango - Welcome to the Partner Developer Network. [online] Available at: http://developer.rottentomatoes.com/ [Accessed 18/11/2016].

IMDb. (2016). IMDb - Movies, TV and Celebrities. [online] Available at: http://www.imdb.com/ [Accessed 10/12/2016].

Odeon.co.uk. (2016). ODEON Cinemas - Book Film Tickets & Check Cinema Listings Now!. [online] Available at: http://www.odeon.co.uk/ [Accessed 10/12/2016].

Omdbapi.com. (2016). OMDb API - The Open Movie Database. [online] Available at: https://www.omdbapi.com/ [Accessed 18/11/2016].

Square.github.io. (2016). Picasso. [online] Available at: http://square.github.io/picasso/ [Accessed 2/12/2016].

Themoviedb.org. (2016). The Movie Database (TMDb). [online] Available at: https://www.themoviedb.org/?language=en [Accessed 18/11/2016].

Workshops

Mr Derek Foster, *Location and maps, extra tasks*, Week A3, [Accessed 10/11/2016]

Mr Derek Foster, *REST tutorial with social media data and extra tasks*, Week A6, [Accessed 18/11/2016]

Mr Derek Foster, *Tutorial for local storage and extra tasks*, Week A8, [Accessed 12/12/2016]