Eric Lui - Duy Pham
Android Project Write-up: DOTA2 Recent Game Analyzer

1. **Project intention**
   Dota 2 is a video game of Valve that allows players to role-play in different heroes and Valve provides some public API for developers. The API provides a list of the most current matches, and each match's detail base on match ID.

   We tried to create an app that retrieves data from the API and stores it to the database. This allows user to access match details as quick as possible from their phone while playing the game.

   We also put heroes and items' data to the app, so that users can access to those data separately from the game API.

2. **Accomplishment**
   The final product of this project is rather complete.
   - We finished almost every basic features, we fixed the SQLite database and rework the API call functions so that the app can live from both the database and the API.
   - The current version is highly dependent on the API to be live, but it does store some data in a SQLite database.
   - Other than the recent games section every other part can run without the API.
   - We designed a final color scheme which relates to the game's main color scheme.

3. **Things not accomplish**
   - The app is also heavily dependent on the DOTA2 Wiki pages and the API. One problem we ran into is with the emulator losing internet connection and slowing down significantly. However, with a physical phone, the app ran smoothly and did not drop the connection.
   - We could have added our own written data for each hero and item along with pictures. This means, we need to collect all the needed pictures from the game, which about 112 different heroes and 140 items. Our best choice is to develop our own server data, and the app can simply just update its' core data from the server instead of calling to the API.
   - We can add more features such as match filtering. It will allow user to select their desired matches instead of the default setting.
   - We can also add a Setting feature. User can modify the app's setting from the Setting activity.
   - Rework the menu and put a Menu button in a header bar, so that user can access to the menu easier.

4. Thing would do different next time
- We would have tried to make a different approach to the original idea. We could develop our own server that corresponds with the app, the server will pull data from the API and simply interact with the apps. It's faster, easier than working directly with Valve API.
- We would have finalized the UI/UX before other feature's development. Even though we created the UI first, we made many changes to the UI, and it affected a lot of the users' experiences in a negative way.
- The software development will be better, as we are more fluent with android. Experience will help us reduce the development time at least by half.

5. Next step
- Add feature: Setting activity, Header bar, Filter bar.
- Add heroes and items images.
- Initial heroes and items data from our server.
- Fixing scaling problem so that the app can work better with a tablet or TV.

To simplify the way which the application retrieves data, we better host our own PHP server or other platform. The app will interact with the server instead of Valve's API. It will allow the app to run faster, as we don't have to handle how we deal with data from the app (data will be handled and specialize in server before sending to the app).

6. Things learned

Duy: For me, what I have learned is that Android is not easy to develop as I thought because the limitation of the devices' resources. We must pay attention to the amount of work on each activity, and how it will affect the UI/UX. For example, reducing the loading time of data (API to JSON, JSON to SQLite). The app must be designed for majority type of devices, but also work well with other old android model. I tested the app with a LG G3 and a Huawei Y300 and it works totally different every single run. Resource management is a useful thing to know, and I look forward to develop a real app that can do great in all aspects: performance, scalability, adaptability and stability.

Eric: I learned that working with a device that is very limited in its processing ability is not easy to develop for. Also working with a physical phone is much easier than working with the emulator as the emulator is very slow and sometimes needs to be completely restarted to work. Also I learned the real importance of a database and relying less on existing resources and creating your own.