D&D Forge Final Presentation Script
12/1/15

1. Hello Everyone! It is our pleasure to present to you D&D Forge. I am <Sound off names>.
2. Andrew - D&D Forge is a mobile character builder for 5th edition Dungeons and Dragons.
3. Alex- We created D&D Forge Android Phones using Android Studio, we programmed in Java and XML.
4. Andrew - Dungeons and Dragons can sometimes be hard to approach. It's an incredible game of improvisational storytelling and imagination, but the preparation for the game can scare people away before the it even begins. Especially if they have little time or support from experienced players. D&D Forge aims to help soften the learning curve of the game and provide a useful tool for experienced players.
5. Andrew - Originally we set out to create a character builder that would help ease new players into the game, and help old players move through the lengthy process quickly. We wanted an application that would produce a printable PDF that could be used at the table without the need for technology.
   Andy - The app could also act as a digital character sheet. Players would track items and abilities, level up their character, and have useful dice rolling tools on hand.
6. Alex - The user starts by choosing an existing character or creating a new one from the main menu.
7. Andy - Then the user will be given a number of question that will help generate their character. This speeds up the process by creating a character based on the user's preferences.
8. Seth - The characters stats are presented on an multi-page character sheet organized specifically for mobile devices.
9. Kevin - The player can also edit sections of the character sheet to keep track of their inventory and expand upon their characters background.
10. Seth - Unfortunately there were some features we were not able to implement in time including: Characters past level 1, Printable PDF's and Automated Ability Checks, some functionality for magic users and some questionnaire functionality. This is mostly due to time constraints and unanticipated blockers during development.

11. Andy - We used Scrum and additional Project Jams, Slack and Google Hangouts for team collaboration. We used Google Drive and Github for project organization.

12. Andrew - At the beginning of the project we interviewed people who played and managed tabletop roleplaying games. This helped us see what people would actually want out of the app, and helped us set some of our own requirements. Players and GM's emphasised the need for a streamlined character creation process, a "Just the essentials" approach to the character sheet, and integrated tools such as a dice roller and automated ability checks.
13. Seth - Compressing dungeons and dragons into a programmable form was a huge task. Some parts of the game could be handled dynamically, however much of it needed to be hard coded. The magic system gave us the most trouble, as most classes used it slightly differently.
14. Andrew - We databased all the character information at level 1. This included all Race, Class and Background abilities, as well as all cantrips and level 1 spells. The database holds the information for the character sheet that would not need to be changed after creation. We created 4 array lists for each category of data, where each entry is searchable by level, or class/race restrictions. This is one of those areas where hard coding was unavoidable.
15. Alex - We built our program in modules and bringing those modules together was a huge task by itself. This modular approach to development helped us avoid certain pipeline blockers. This also made the team much more disconnected than if we had been otherwise.
16. Alex - <Explained Sharedpreferences>
17. Kevin - At first we expected to use activities, which are the main type of process subdivisions in android, for basically everything. We quickly realized that although it was the most intuitive approach, it made passing data and managing positions of different processes difficult to deal with. Making activities for each process in our app would make it so users would constantly have to click back and forwards on different links as opposed to being able to swipe seamlessly between the information they needed. Furthermore, when passing data through activities, you would need to specify exactly what it is you are sending to each different process, and this would make it much more likely for information to be lost. Due to those drawbacks, we decided to instead use fragments which are lightweight and can be represented in only one activity. They natively come with

the ability to swipe between screens and are therefore much more user friendly. Also since they are contained within one activity, we could instead just send one package of data to the main activity which contained all of the fragments and then allocate it dynamically from there instead of having to individually manage the information being passed. By switching from having an activity dependent app to a fragment dependent one, we improved the experience for both the users and the design team.

18. Andy - Bringing the variety of talents of the team to bear was another challenge. Many on the team were not familiar with Dungeon and Dragons or Android programming at the beginning of the project.
19. Andrew - Project jams where we all worked together on the project at the same time helped ensure team collaboration beyond scrum meetings.

Andy - Modular programming was a pro and a con, it sped up our development time at the expense of team collaboration.

Kevin - Our overall scope ambitious and there were many features that we were not able to implement in time.

20. Andrew - Overall we met our main goals of a usable character creation tool. It is highly modular, and given more time could be expanded upon easily.

Thankyou!

Questions?