The Dungeon Master – Eric Komperud – For Final Project Consideration

Abstract: I envision The Dungeon Master to be an app which helps with combat in tabletop roleplaying games like Dungeons & Dragons and Pathfinder. The primary screen would be a combat canvas grid which could be drawn on to display character positions and structures as well as keep track of character health, initiative, etc. This could then be shared with other players by connecting a tablet through an HDMI port or through a wireless connection feature. In addition, a monster manager would be included for generating any kind of pre-created monsters on the map.

App Screen Descriptions:

- **Map:** A drawable canvas grid with a settable size. Users would be able to scroll across the map, zoom in, change their brush settings, modify and add character/monster tokens, view initiative order, and simulate spells. Users could also import photos from their phone to underlay on the map. Many features would be handled through small interactable popups contained within it. Character/monster tokens could be interacted in this way, tapping on one to pull up its editable stats and portrait. A bottom bar would allow users to pick from various features to use like the brush, token placer, and spell simulator which would then be clearly displayed on the screen. Tapping a tool which was already selected would bring up options to change the properties of that tool. At the top, a button would take users to a scrollable screen of saved maps and a button to create a new map. If they had modified the current map, they would be prompted to save.
- **Monster Manager:** At the top, a header contains options for sorting (by alphabetical order, monster type, etc.), creating new monsters, and deleting monsters. The main area of the screen will be a scrollable list displaying the monsters currently saved by the users. A list of default monsters may also be included with the app.
    - **Monster Creator:** A basic interface which will allow players to set combat related stats for the monster through input fields, a token to use on the map, and an optional portrait as well. The stats will also have an optional for standard deviations.
        - **HP**
        - **Armor Class**
        - **Attack Modifier**
        - **Saving Throw modifiers**
        - **Token**

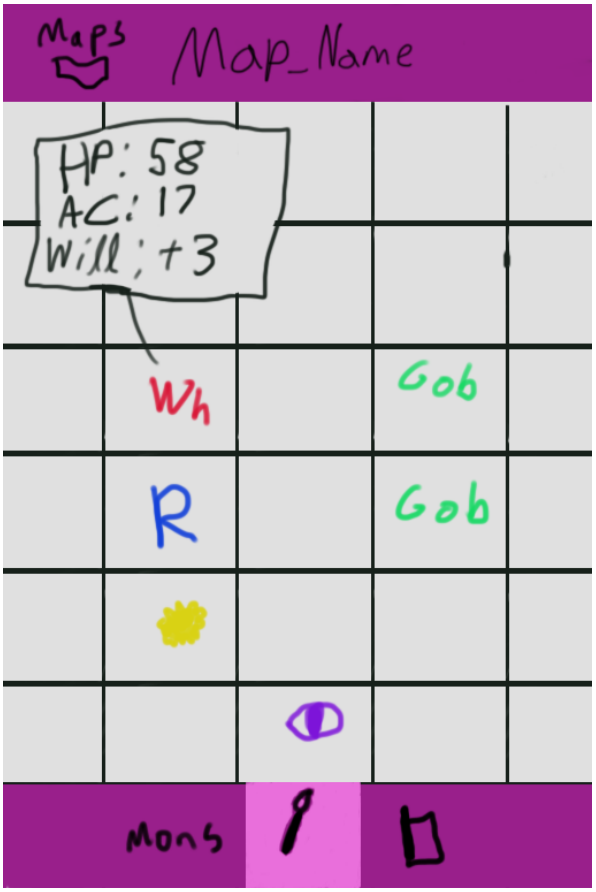
    - **Portrait:** Users would be allowed to import images from their device for the Portrait and/or Token

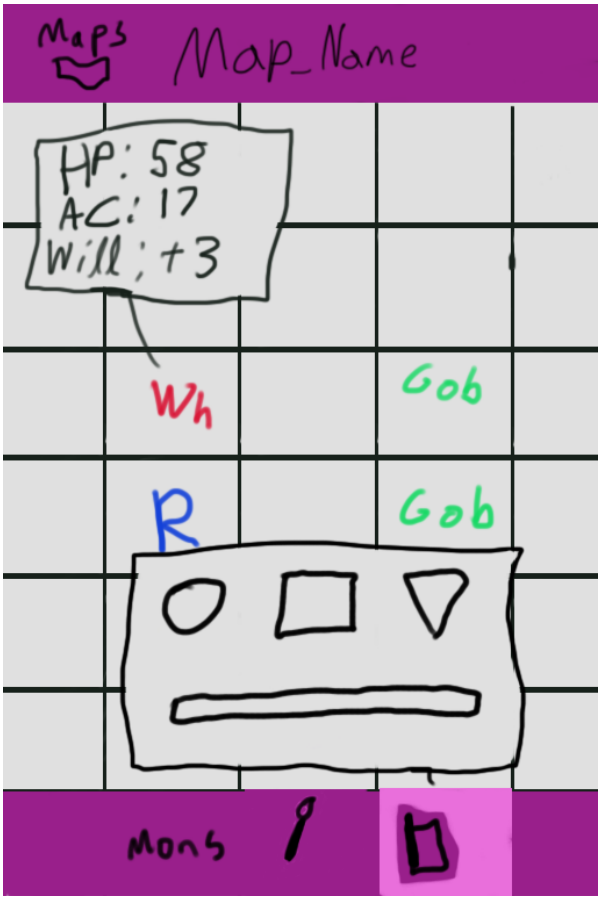
Future Features:

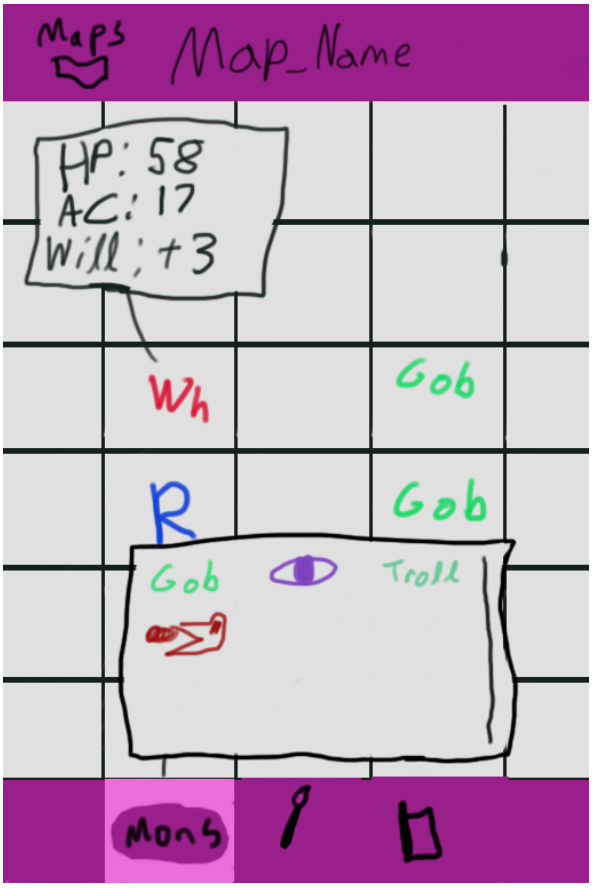
- **Monster Compendium:** Include more complete information on monsters including things like special abilities and features
- **Monster community:** Connect with other users to share monsters

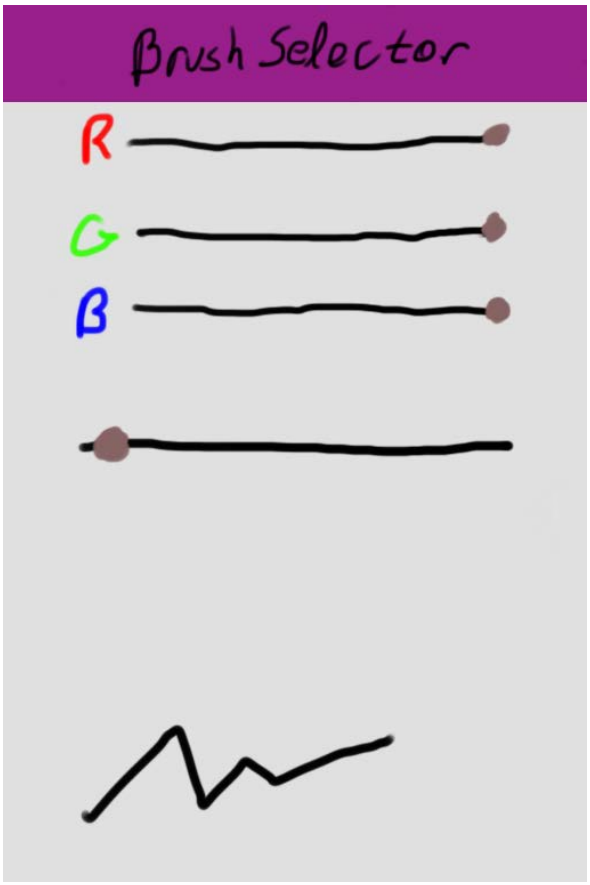
Breakdown and Tracking:

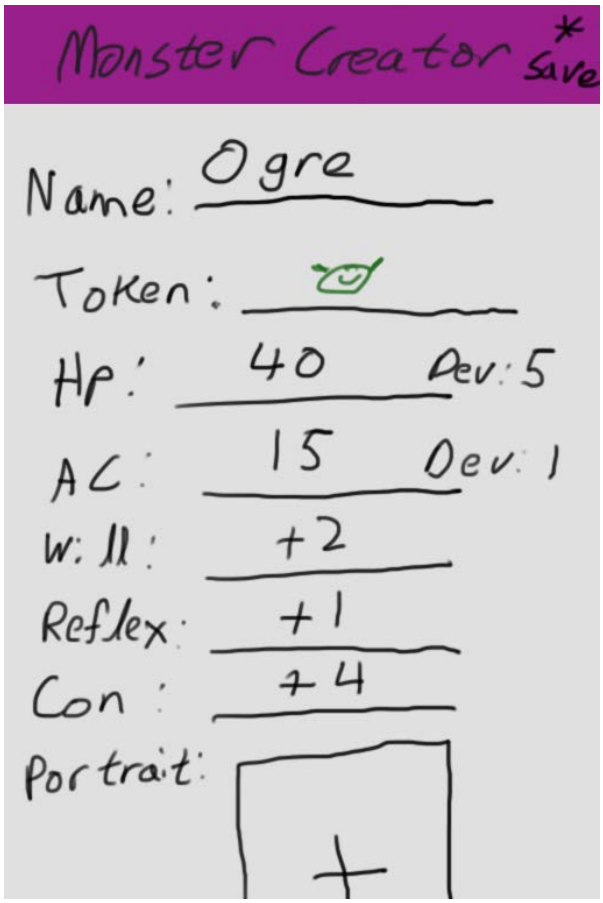
Task	**Estimated Time**	**Actual Time**
Monster Manager data model	1 hour	2 hours
Monster Manager view	2 hours	4 hours
Monster Manager controller	2 hours	4 hours
Monster Creator	1 hour	6 hours
Map data model	2 hours	2 hours
Map view	5 hours	7 hours
Map controller	5 hours	4 hours
Opening screen	< 1 hour	0 hours
App controller	3 hours	0 hours
Totals:	21 ~ 22 hours	29 hours


Maps
Map_Name
HP: 58
AC: 17
Will: +3
Wh
Gob
R
Gob
Mons


Maps
Map_Name
HP: 58
AC: 17
Will: +3
Wh
Gob
R
Gob
Mons


Maps
Map_Name
HP: 58
AC: 17
Will: +3
Wh
Gob
R
Gob
Gob
Troll
Mons


Brush Selector
R
G
B



Post Development Analysis: A general theme of developing this app was me underestimating how much time each feature would take. This was especially due to trying out new technologies such as Interface Builder and Core Data. In addition, I had not familiarized myself with most of the classes I was going to be using such as UIScrollView, and UIImagePickerController, beforehand. That being said, I should definitely take the learning curve into account when approaching future projects with time constraints.

The fully implemented features are as follows:

- Monster Maker
    - Create monsters with various basic stats as well as a portrait
    - Paint portraits with a color-customizable brush. (This was originally going to be an image uploader, but the IPhone simulator's privacy settings were non-existent on the library computers, and thus, I could not access the simulator's photo gallery)
- Monster Manager
    - Save and load monsters made in Monster Maker. Data persists
    - Monsters are sorted alphabetically and a new monster option is clearly visible

Partially Implemented / Unimplemented Features:

- Map
    - Users can choose map size and grid reflects this in units of 100. Screen scrolls to accommodate larger maps
    - Users can pick monsters from Monster Manager and place them onto the map, as well as delete them. They cannot move monster tokens or view/edit their stats from the map screen.
    - Brush remains unimplemented. Was intended to allow users to draw environments.
    - Saving/loading maps remains unimplemented. Users were to be able to save/load maps, and view/pick them in a CollectionView type environment.