**Introduction**

Group wants to create a mobile application for Dungeons and Dragons character sheets. Dungeons and Dragons character sheets include a questionnaire, races, abilities, starter spells, classes, capabilities, and attributes, selected from a database both from user input and randomization.

***Purpose***

The purpose of this document is to serve as a guide to designers, developers and testers who are responsible for the engineering of the DnD character sheet and mobile app project. It should give the engineers all of the information necessary to design, develop and test the software.

***Scope***

This document contains a complete description of the functionality of the DnD character sheet and mobile app project. It consists of use cases, functional requirements and nonfunctional requirements, which, taken together form a complete description of the software

***System Overview***

This database and mobile app should be able to print out a questionnaire which will let the player figure out what class and capabilities they will have. There should be a randomization of race, abilities, and starter spells. It should also be able to roll the dice for status. Finally, there will be a 2D character sprite that the player created.

***References***

Dungeons & Dragons The Player's Handbook

Dungeons & Dragons The Xanathar’s Guide to Everything

Dungeons & Dragons Tasha’s Cauldron of Everything

Dungeons & Dragons Ghosts of Saltmarsh

**Definitions**

Skill Monkey: Someone who does an abundance of things

Strength: Character’s physical strength. Abbreviated as STR

Dexterity: Character’s agility and ability to dodge attacks. DEX

Constitution: Physical resilience, especially to disease and poison. Can also refer to the character’s physique. CON

Intelligence: Ability to problem solve and recall information. Used for most spell casting. INT

Wisdom: Common sense, cleverness and observation of the character’s surroundings. WIS

Charisma: Social skills and physical appearance. Examples of usages would be lying and persuasion. CHA

Class: The type of character. Determines how the character can be played in combat as well as their general abilities, such as spellcasting or proficiency with weapons.

Barbarian: A combatant whose abilities focus on physical strength and force. Their abilities come from rage, and can be increased when in the Raging state.

Bard: A musician who can spell cast using a musical instrument. This is their primary form of combat, but they can also use some weaponry.

Cleric: Priests and warriors who use divine magic. The nature of this magic depends on the cleric's deity. To cast spells, they call upon their deity through a holy object.

Druid: Characters whose abilities come from nature and cast spells using flora and fauna. One notable ability is the ability to turn into animals.

Fighter: Warrior that is especially skilled in weaponry. They usually have a primary focus on one particular weapon type.

Paladin: A knight of a chosen deity with a focus on righteousness and justice. Their beliefs on the meaning of justice depend on the ideals of their deity.

Ranger:A hunter class who is proficient in bows and arrows or dual wielding. They have a specific enemy type that is their quarry.

Rogue: Stealthy fighter who can unarm traps and sneak attacks. Typically thieves in origin.

Sorcerer/Sorc: A type of magic user whose magic comes from a magical heritage. They do not need to memorize spells.

Warlock: A type of magic user whose magic comes from a pact with a supernatural entity. Their magic is typically dark and eldrich.

Wizard: Spell caster who must study and memorize spells to use them.There are many different requirements to their spellcasting, such as the day or the components the character has.

Alignments: The nature of the character. Influences their beliefs, actions, and decisions. Each alignment has two traits. The first traits can be Lawful, Neutral, or Chaotic. The second can be Good, Neutral, or Evil. The combination of Neutral and Neutral is called True Neutral.

Race: The different species a character can be. Human, Elf, Half-Elf, Drow, Dragonborn, Dwarf, Gnome, Half-Orc, Goliath, Halfling, Tiefling. A character’s race can influence their class and alignment, as certain races are more likely to be certain classes.

Stats: The attributes of a character. STR CON WIS INT CHA DEX. each stat has a modifier attached to it based on its numeric value

Modifier (Mod) : A number that modifies the rolled number from character traits, or the number added to a player’s skill roll during gameplay.

D4, d6, d8, d10, d12, d20(Dx): Refers to the type of dice based on the number of sides the dice has.

**Use Cases**

| **Name** | **UC-1: Display Information** |
| --- | --- |
| Summary | This will display the information that is given from the class, attribute, background, alignment, stats, and race tables and give the user all the information. |
| Rationale | Allows users to be able to see their information |
| Users | All users |
| Preconditions | Display creation has begun |
| Basic Course of Events | 1. The user indicates that they want the software display all the information (class, attribute, background, alignment, stats, and race) 2. The software responds by requesting all the information from the database 3. Information is displayed |
| Alternative Paths | 1. The user does not want to display information |
| Postconditions | Class information from the questionnaire will be given as well as the randomization of the other tables will be shown. |

| **Name** | **UC-2: Create & Display Name Information** |
| --- | --- |
| Summary | This will create the information that is given from the name table and give the user information for that given name. |
| Rationale | Allow user to be able to see name information and change the characters name if they so desire. |
| Users | All users |
| Preconditions | Name creation has begun |
| Basic Course of Events | 1. The user indicates that they do not have a chosen name for the character and asks for a random name 2. The software responds by choosing a randomized name 3. Show the user the chosen name and ask the user if they want a different randomized name 4. Choose and display names until the user clicks the choose name button |
| Alternative Paths | 1. User decides to stop the character creation process |
| Postconditions | Name information is chosen for the created character and will be saved when the character saves |

| **Name** | **UC-3: Save Character Information** |
| --- | --- |
| Summary | The created character’s information is saved for later. |
| Rationale | The user wants to save the character that was created so that they can view it later. |
| Users | All users |
| Preconditions | User completes the character creation process and decides to save the character |
| Basic Course of Events | 1. The user has completed the character creation process 2. The user clicks the save character button 3. The created character is saved to the character list 4. Data is saved under the characters name |
| Alternative Paths | 1. The user decides that they don’t want to save the character. |
| Postconditions | Current character is saved and saved in a list of saved characters. |

| **Name** | **UC-4: Sprite Generation** |
| --- | --- |
| Summary | Making sure that the sprite is visible to the user and they are able to see the final product based on the questionnaire and the rolling of the dice |
| Rationale | Allow user to be able to visualize the sprite and combination of traits given |
| Users | All users |
| Preconditions | The character’s race and class are determined. |
| Basic Course of Events | 1. The software will look for the correct combination of race and class from the collection of sprites. 2. the software will display the correct sprite to the user 3. The user will be able to screenshot the given character |
| Alternative Paths | 1. Given different combinations will give very different character sprites. Every character can look different based on what the user chooses |
| Postconditions | Sprite character will be generated for the user |

| **Name** | **UC-5: Dice Rolling for the Purpose of Ability Assignment** |
| --- | --- |
| Summary | The values that come from rolling 4 d6 will later be used to calculate base ability scores |
| Rationale | The user has requested a roll of the dice. |
| Users | All users |
| Preconditions | The user requests to roll the dice or the dice are being rolled in the character creation process. One or more stat has not been assigned a value. Character class and race have been determined |
| Basic Course of Events | 1. The user request to roll the dice 2. The user chooses the ability score they want to roll for 3. The chosen dice are rolled and a number is chosen (UC 11) 4. The sum of the dice excluding the lowest value is calculated 5. The software looks for and adds any modifiers that come from the character’s class or race 6. The software looks for which range of numbers the resulting number falls into 7. The modifier that corresponds to that range of numbers is set as the character’s modifier for that ability. |
| Alternative Paths | The user chooses not to roll the dice.  The user tries to roll for an ability score that already has a value  The user chooses to start with a basic distribution of ability modifiers |
| Postconditions | The selected ability is assigned a modifier |

| **Name** | **UC-6: Rolling Dice** |
| --- | --- |
| Summary | After the dice finishes ‘rolling’, it must return a value as the result of that dice roll |
| Rationale | The user will be able to see what numbers they rolled so that they understand why their ability modifier comes from |
| Users | All users |
| Preconditions | The user has begun the dice rolling process (UC 10) |
| Basic Course of Events | 1. The user asks to roll the dice 2. A random number between 1-the maximum number for that type of dice is selected 3. This number is set as the return value for a dice object 4. The values from the dice roll are displayed to the user as well as stored in a dice log 5. This can be repeated if more dice values are necessary |
| Alternative Paths | No alternative path |
| Postconditions | The rolling of a die has been simulated and the user has a random number for further usage. |

| **Name** | **UC-7: Dice Storing** |
| --- | --- |
| Summary | A log of all of the numbers that have been chosen by the dice along with any modifications made by the relevant modifiers. |
| Rationale | The user may be interested in the history of all of their dice rolls to look for any patterns. This can also be used by the admin to check that the rolled values are random. |
| Users | All users |
| Preconditions | Dice have been rolled (UC 11) |
| Basic Course of Events | 1. All information that comes from dice rolls is recorded in the dice roll log. |
| Alternative Paths | no alternative paths. |
| Postconditions | The dice roll log only displays the 15 most recent rolls of the dice. |

| **Name** | **UC-8: Questionnaire Use - Class** |
| --- | --- |
| Summary | How the user goes through the questionnaire that determines some character details |
| Rationale | The questionnaire is the first step in the character creation process and determines the character’s class and subclass |
| Users | All users |
| Preconditions | The user has asked to begin the character creation class and has opted to take the questionnaire |
| Basic Course of Events | 1. The user is presented with a question and three answer options 2. The user selects their answer 3. The user is shown the next question 4. Repeat 1-3 until end of questionnaire |
| Alternative Paths | User has chosen to skip the questionnaire |
| Postconditions | The character class has been determined |

| **Name** | **UC-9: Select Race** |
| --- | --- |
| Summary | At times a user may have an idea for what they want the race of their character to be or may simply change their mind and there should be a way to accommodate for that |
| Rationale | It is useful for the developer and user to be able to select a race |
| Users | all |
| Preconditions | Character class and initial race has been generated |
| Basic Course of Events | 1: user selects the dropdown menu next to race  2: user selects from among the options provided  3: the stats, sprite, and abilities change to reflect the new race’s inherent modifiers |
| Alternative Paths | User cancels the selection process |
| Postconditions | Character sheet is displayed with new race |

| **Name** | **UC-10: Select Class** |
| --- | --- |
| Summary | At times a user may have an idea for what they want the class of their character to be or may simply change their mind and there should be a way to accommodate for that |
| Rationale | It is useful for the developer and user to be able to select a class after it is generated from the questionnaire |
| Users | all |
| Preconditions | Character class and initial race has been generated |
| Basic Course of Events | 1: user selects the dropdown menu next to class  2: user selects from among the options provided  3: the sprite and abilities change to reflect the new class |
| Alternative Paths | User cancels the selection process |
| Postconditions | Character sheet is displayed with new class |

| **Name** | **UC-11: Select Alignment** |
| --- | --- |
| Summary | At times a user may have an idea for what they want the alignment of their character to be or may simply change their mind and there should be a way to accommodate for that |
| Rationale | It is useful for the developer and user to be able to select a alignment |
| Users | all |
| Preconditions | Character class and initial race has been generated |
| Basic Course of Events | 1: user selects the dropdown menu next to alignment  2: user selects from among the options provided  3: the alignment section of the character sheet updates |
| Alternative Paths | User cancels the selection process |
| Postconditions | Character sheet is displayed with the new selected alignment |

**Functional Requirements**

| **Name** | **FR-1: Classes** |
| --- | --- |
| Summary | All the classes that can be selected or generated from the questionnaire must be listed along with their abilities |
| Rationale | The classes are a primary and pivotal thing within the program itself and must be listed correctly |
| Requirements | Classes listed with all abilities from the main class and a subclass up to level 3 |
| References | UC-1  UC-3 |

| **Name** | **FR-2: Dice** |
| --- | --- |
| Summary | The numbers on the dice are associated with the numbers 1-3, 1-4, 1-6, 1-8, 1-20 |
| Rationale | The dice are used to choose random numbers |
| Requirements |  |
| References | UC-11  UC-12 |

| **Name** | **FR-3: Access the Database** |
| --- | --- |
| Summary | Ability to access the database from sql to C# |
| Rationale | There needs to be a connection with a sqlite API |
| Requirements | API needed |
| References |  |

| **Name** | **FR-4:** |
| --- | --- |
| Summary |  |
| Rationale |  |
| Requirements |  |
| References |  |

| **Name** | **FR-5:** |
| --- | --- |
| Summary |  |
| Rationale |  |
| Requirements |  |
| References |  |

| **Name** | **FR-6:** |
| --- | --- |
| Summary |  |
| Rationale |  |
| Requirements |  |
| References |  |

| **Name** | **FR-7:** |
| --- | --- |
| Summary |  |
| Rationale |  |
| Requirements |  |
| References |  |

| **Name** | **FR-8:** |
| --- | --- |
| Summary |  |
| Rationale |  |
| Requirements |  |
| References |  |

**NonFunctional Requirements**

| **Name** | **NF-1: Dice Limits** |
| --- | --- |
| Summary | Each die you can roll has a clearly defined maximum and minimum result |
| Rationale | When I roll a 1d6 I shouldn't get a 7 or a 0 or a -1 |
| Requirements | All dice have a defined minimum and maximum result |
| References | UC-7  UC-10  UC-11  UC-12 |

| **Name** | **NF-2: Dice Appearance** |
| --- | --- |
| Summary | When rolling dice there will be an brief animation |
| Rationale | There should be a clear showing that a dice is being rolled and a result is being given |
| Requirements | Animation where dice rolls is shown and then in the log a result appears  Dice should be clearly visible against the background |
| References | UC-12  UC-10  UC-11  UC-7 |

| **Name** | **NF-3: Dark Mode** |
| --- | --- |
| Summary | Add a function to make the character sheet darker for low light situations |
| Rationale | It is harmful to the user's eyes to view a bright white object in the dark |
| Requirements | Function that changes black to white and white to black and vice versa and back again |
| References |  |

| **Name** | **NF-4: Display of Information** |
| --- | --- |
| Summary | The software should be able to display things based on the stage in the creation process. For example, numbers should be shown when the dice are being rolled |
| Rationale |  |
| Requirements |  |
| References |  |

| **Name** | **NF-5: Correct sprite combination** |
| --- | --- |
| Summary | The sprite that is shown should be the right combination of character class and race. |
| Rationale |  |
| Requirements |  |
| References |  |

| **Name** | **NF-6:** |
| --- | --- |
| Summary |  |
| Rationale |  |
| Requirements |  |
| References |  |

| **Name** | **NF-7:** |
| --- | --- |
| Summary |  |
| Rationale |  |
| Requirements |  |
| References |  |