**Software Requirements**

**Specification**

**for**

***Rocket Math***

Version 1.0

February 18, 2015

**Prepared by:** **Samuel Arseneault, Kelly Markaity, John Luu**

Table of Contents

[**1. Introduction** 1](#_Toc411895715)

[**1.1 Purpose** 1](#_Toc411895716)

[**1.2 Scope** 1](#_Toc411895717)

[**1.3 Definitions, Acronyms, and Abbreviations** 2](#_Toc411895718)

[**1.4 References** 2](#_Toc411895719)

[**1.5 Overview** 2](#_Toc411895720)

[**2. Overall/General Description** 3](#_Toc411895721)

[**2.1 Product Perspective** 3](#_Toc411895722)

[**2.2 Product Functions** 4](#_Toc411895723)

[**2.3 User Characteristics** 5](#_Toc411895724)

[**2.4 General Constraints** 6](#_Toc411895725)

[**2.5 Assumptions and Dependencies** 6](#_Toc411895726)

[**3. Specific Requirements** 6](#_Toc411895727)

[**3.1 External Interface Requirements** 7](#_Toc411895728)

[**3.1.1 User Interfaces** 7](#_Toc411895729)

[**3.1.2 Hardware Interfaces** 14](#_Toc411895730)

[**3.1.3 Software Interfaces** 14](#_Toc411895731)

[**3.1.4 Communications Interfaces** 14](#_Toc411895732)

[**3.2 Functional Requirements** 15](#_Toc411895733)

[**3.2.1 User Class 1- Student** 15](#_Toc411895734)

[**3.2.2 User Class 2- Teacher** 18](#_Toc411895735)

[**3.3 Non-Functional Requirements** 21](#_Toc411895736)

[**3.3.1 Performance** 21](#_Toc411895737)

[**3.3.2 Reliability** 22](#_Toc411895738)

[**3.3.3 Availability** 22](#_Toc411895739)

[**3.3.4 Security** 22](#_Toc411895740)

[**3.3.5 Maintainability** 24](#_Toc411895741)

[**3.4 Design Constraints** 24](#_Toc411895742)

[**4. Analysis Models** 25](#_Toc411895743)

[**4.1 Data Flow Diagrams (DFD)** 25](#_Toc411895744)

[**4.2 Prioritization and Release Plan** 25](#_Toc411895745)

[**4.2.1 Choosing the Priority** 26](#_Toc411895746)

[**Appendix** 26](#_Toc411895747)

[**Appendix I: Priority Scheme** 26](#_Toc411895748)

[**Appendix II: Release Plan** 28](#_Toc411895749)

[**Appendix III: Project Plan** 37](#_Toc411895750)

# **1. Introduction**

## **1.1 Purpose**

The purpose of this Software Requirements Specification document is to provide a detailed description of the functionalities of the Rocket Math system. This document will cover each of the system’s intended features, as well as present the user interface of each component of the system. The document will also cover hardware, software, and various other technical dependencies.

## **1.2 Scope**

The “Rocket Math” is an iOS based application that will help first grade students learn and improve their math skills. Students will interact with the Rocket Math application and will be asked to solve basic math operations. This application should be accessible on the apple store, and free to download. It should also be able to run on the iPad Air and iPad 2.

The first grade teachers will be able to create a class and add students to their class using a web portal. After a teacher has setup a class using the web portal, students should be able to login with the IOS application using their unique identifier. The teacher will also be able to view the students’ records using the login portal. The teacher will be able to view information about their students such as which level students are stuck on, how much time spent on a level, a snapshot of their most current run of a level.

For the iOS application, the students will be asked to perform basic math facts. The difficulty of these basic math facts will be split up into levels, which become increasingly harder as they master the levels. Once a level is mastered, the students will then be able to move on to unlock new and more challenging levels. While the students are interacting with the iOS application, they will also be generating data about their performance throughout the levels. Information such as the number of questions wrong, the amount of time spent on the level, and which level they struggle with the most, will all be sent to a server to be stored in a data base.

Furthermore, this system will need constant internet connection to grab and store student information. The iPads will always need to be connected to a Wi-Fi connection, as well as the server always connected to a network. All the information maintained in a database, which will be located on a web server.

## **1.3 Definitions, Acronyms, and Abbreviations**

|  |  |
| --- | --- |
| **Term** | **Definition** |
| iOS | A mobile operating system developed by Apple Inc. which powers the iPad device. |
| iOS Application | The application that our team will be creating, which will run on iOS devices. |
| Web Portal | An internet application which allows a teacher to log to and access student records. |
| MB | MB stands for megabyte, which is a unit of computer memory or data storage. |
| FR | FR stands for functional requirement. Used for the functional requirement titles. |
| NF | NF stands for non-functional requirement. Used for the non-functional requirement titles. |
| Student | User of the iOS application. |
| Teacher | User of the web portal. |

## **1.4 References**

National Governors Association Center for Best Practices & Council of Chief State School Officers. (2010). Common Core State Standards for Mathematics. Washington, DC: Authors.

1.4 References [1] IEEE Software Engineering Standards Committee, “IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications”, October 20, 1998.

## **1.5 Overview**

Chapters 2, 3, and 4 of this document will provide a description of the systems functionality, and user interaction with the system. Chapter 2 will focus on a general description of how the system will work, along with a high level perspective. It will also contain the user interactions with the different components. Chapter 3 will focus on the specific descriptions of how the system will work. Chapter 3 will contain the functional and nonfunctional requirements that explain in detail how specific individual parts of the system will work. The last chapter, chapter 4, will explain the high level overview of how data will flow through the system. Chapter 4 will also contain the reasoning behind the prioritization of the requirements. The actual prioritization of the requirements can be found the Appendices, along with the project schedule.

# **2. Overall/General Description**

## **2.1 Product Perspective**

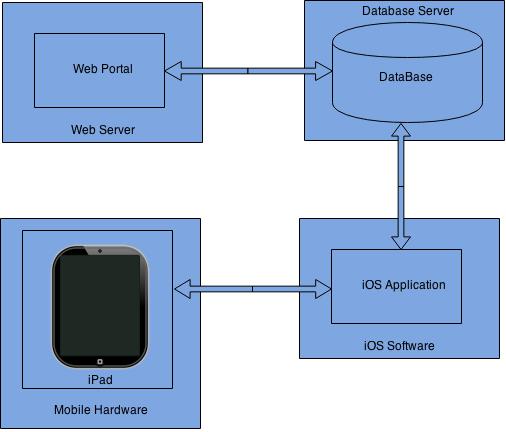
The Rocket Math system is composed of three parts: the login portal for the teacher to interact with, the IOS application for the first grade students to interact with, and a database to store and retrieve data from both the iOS application and the web portal.

Before the students will be able to use the iOS application, a teacher will have to create a class and add students with the web portal. This information will be sent to the database to be stored. Once this is complete the students will be able to logon to the iOS application, and retrieve information from the database.

The iOS application will also receive physical data from the students’ input from the hardware. Inside the iOS application this physical data will be used to generate information about the student’s performance, which will be sent to the database to be stored. The web portal will be also able to retrieve this student information.

The data base that is connected to the iOS application will maintain information about the students accounts for the iPad as well. A student will be able to use any iPad they would like since their account will be linked to this database, opposed to having the application information stored on independent iPads. A student should be able resume their progress on the iOS application even if they were to sign into a different iPad.

Figure 1 - Block Diagram

****

## **2.2 Product Functions**

Below is a brief outline of the basic functionality of the Rocket Math system. The outline is split into two sections: the iOS application features and the web portal features.

**iOS Application**

* Application Login Account
  + Each student will be assigned a username and password.
  + Teacher will have access to all usernames and password incase a student forgets their information.
* Math Practice
  + Teacher will set the number of problems per section.
  + There is no time limit on completing the practice section.
  + Will keep track of how long student worked on practice section.
  + If student does not answer correctly, the correct answer will be displayed and the student must enter the correct answer.
  + There is no limit on how many times a student does a practice section.
* Math Test
  + Student will be asked 15 math questions.
  + In order to pass the test the student must answer the questions correctly within a set time.
  + Student must get all of the questions correct to pass the test.
  + Student can only one test a day.
  + Student will have 1 minute to complete test.
* Math Levels
  + The app will have 18 levels.
  + Each level introduces new math problems
  + Once a student passes the level test, they will move onto the next level.
  + The levels will still display math problems from previous levels

**Web Portal**

* Web Portal Login Account
  + Teacher will be able to create an account or log in to an existing account.
  + Teacher will be able to answer a security question if password is forgotten.
* Create Class and Adding  Students
  + Teacher will be able to create a new class and password.
  + Can add and remove students to existing class.
* Creating Levels
  + Teacher will have the ability to add levels for existing class.
  + Teacher can add new questions to existing levels.
* Adding Questions to Levels
  + Teacher can add as many questions to a level as they want.
  + When creating new questions, the teacher will choose whether it is a practice question or a test question.

## **2.3 User Characteristics**

There are two types of users that will be interacting with our system: the students who will be using the mobile iOS application, and teachers who will be using the online web portal. The students will primarily be generating data, while the teachers will primarily be retrieving the students’ data.

For the iOS application, the students will be able to login and begin learning their math facts. At the home screen, the students will be presented with a map that contains levels to choose from. The difficulty of the levels ranges from easy to difficult, with the first level being easy. They will also not be able to access levels if they did not master the previous level. Once a student selects an available level, they will be presented with 2 choices, practice mode and test mode. After selecting the desired mode, they should be prompted with math questions which they will be asked to enter input for.

The teachers will not be interacting with the iOS application, but instead the web portal. The web portal will allow a teacher to create their own personal account and login. After a teacher is logged into the web portal they will be able to create a class, and add students to the class. After the class and students are set up, the teacher will then be able create levels and add questions to levels. Once the students have interacted with the iOS application, the teacher will be able to access the student’s information from the web-portal. The teacher will be able to view the levels the students have competed, with the ability to enter a completed level to view more information. Inside of a level view, the teacher will be presented with more detailed information such as questions right, number of attempts, or time spent.

## **2.4 General Constraints**

Internet connection is a major constraint for the Rocket Math system. The iOS application must be connected to the internet so it can successfully send information and data to the database. Without internet connection, the application will not work.

The iOS application is also constricted to iPads. The application may not be available on iPods, iPhones, or other apple devices.

Since the iOS application and the web portal will be sharing the same database, at times it may be forced to queue requests from both application. Ultimately this will cause delays when either application is attempting to retrieve information.

## **2.5 Assumptions and Dependencies**

One assumption is that the students will be using iPad 2’s and iPad Air’s to use the iOS application. If a scenario does come up where different iPads are used, the specifications for this document may have to change.

Another assumption is internet access will be available. In order for the iOS application to send data to the database it must have internet access. The same applies for the login portal which requires a connection to the internet to access the database.

# **3. Specific Requirements**

This section contains all the functional and nonfunctional requirements of our system. These requirements provide a detailed description of the available features and the overall system.

## **3.1 External Interface Requirements**

This section provides basic visual prototypes for both the web portal and iOS application. These interface sections s describe the inputs and outputs the system will have. This section will also provide a description of the hardware, software, and the technology that will be used to interface between the applications.

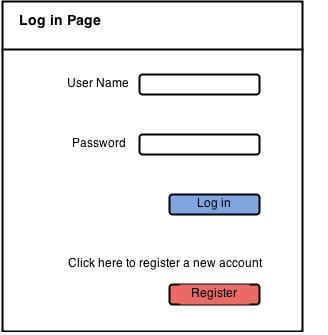
### **3.1.1 User Interfaces**

Since there are only two users, the teacher and the student, this section will just cover their user interfaces with wire diagrams.

#### 3.1.1.1 Teacher View

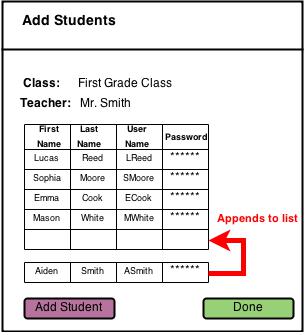
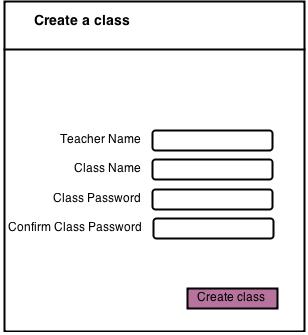
This is the view the teacher will experience when interacting with the web portal.

When connecting to the web portal, a teacher will first be prompted with a login page requesting their username and password, see Figure 2. If the teacher is a first-time user, they will have to select the register button and create an account, see Figure 2 and Figured 3.

Figure 2 - Login Page                                              Figure 3 - Registration page

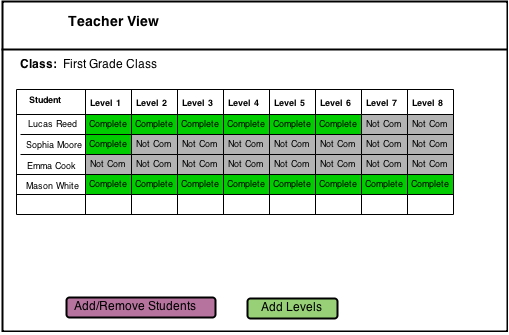
Once the teacher has registered and signed in, they will be presented with an option to create a class, see Figure 4. After a class is created, they should then be allowed to start adding students to the class by selecting the “Add Student” button. Once they are finished adding students they should be able to select the “Done” button, see Figure 5.

 Figure 4 - Creating a class        Figure 5 - Adding students



After a teacher has created a class and added student, they should be presented with a teacher view page. In fact, every time a teacher logins, after their class is setup, they should be presented with this page. Inside of the teacher view page, the teacher will be able to track the progress of their students. This page contains a table of students and levels, which tracks which levels the students have and have not completed, see figure 6.

Figure 6 - Teacher View



Inside the teacher view, the teacher also has the ability to click on the completed levels to view the current progress. After they select the desired level they will be presented with the level progress view. In this view the teacher can assess the most current level performance of her student; observing information such as total time spent on level, number of retries, and the most current snapshot, see figure 7.

Additionally inside of the teacher view, the teacher should be able to add levels. By selecting the “Add Levels” button, a teacher will be presented with the “add new level view”. This view will allow a teacher to add practice and test questions to the level, along with level information, see figure 8.

Figure 7 - Level progress view

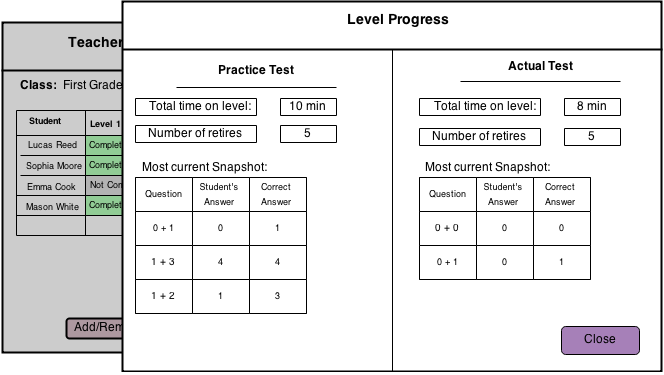
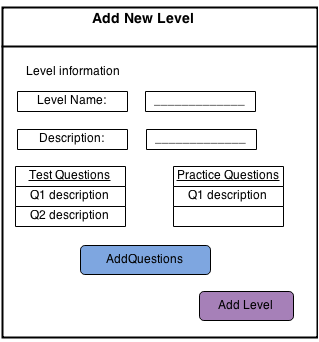
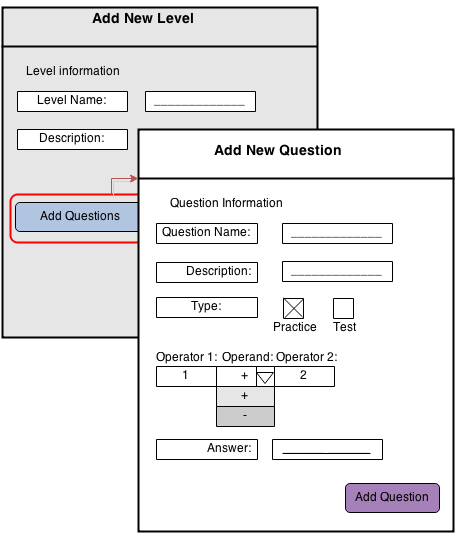


Figure 8 - Adding a new level



When a teacher is inside of the “add new level view”, they will be able to add test and practice questions”, please see figure 8. When they select add test or practice questions they will be prompted with the “add new question view”, see figure 9. Inside of the “add new question view” a teacher will have the ability to enter their question and answer.

Figure 9 - Adding a new question

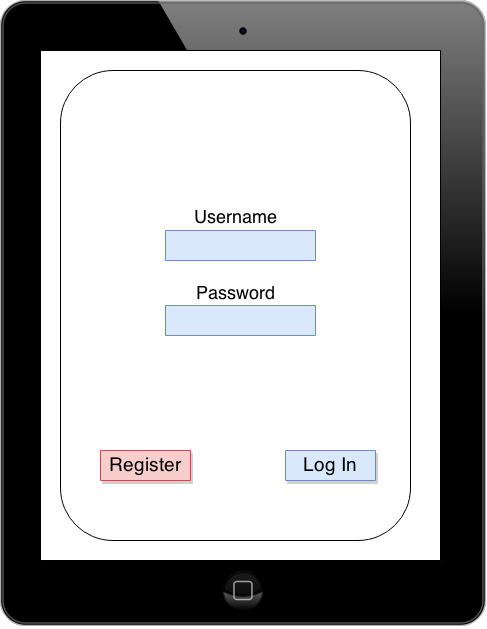
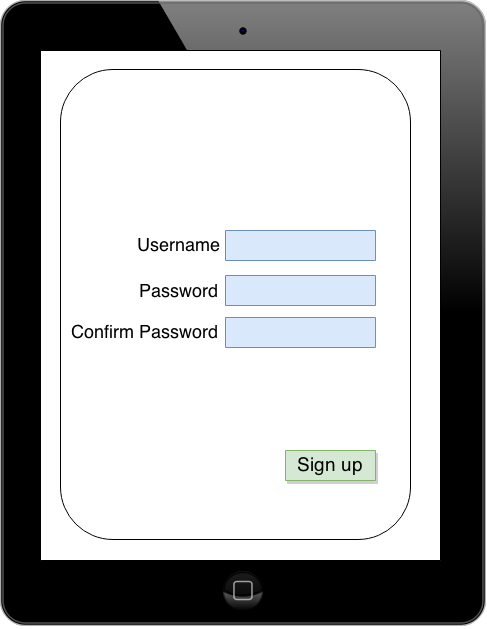
****

#### **3.1.1.2 Student View**

This is the view the students will be using when interacting with the iOS application.

When entering the iOS application, a student will be prompted with a login screen requesting their username and password see Figure 10. If the student is a first-time user, they will have to select the register button and create an account, see figure 11.

Figure 10 - login screen                                     Figure 11 - Registration screen

If a student is logged in they will automatically be presented with the level select screen, see figure 12. This is the screen where they can choose which level they would like to enter. Note: the levels will be accessible depending if the previous level is “mastered”. For a level to be mastered a student must have completed the level with 100% accuracy.

Once a student has selected their desired level, they will be prompted with a select level type screen, see figure 13. In this screen they will be able to select a test type, the choices are from a practice test, or an actual test. After they have selected the test type they will enter the playable test screen. In this screen the student will have the ability to drag and drop numbers into the selected answer column, see figure 14. Once they have entered an answer they can select the next button to move onto the next question (in the figure the next button is shown as an arrow).

 Figure 12 - Level select screen

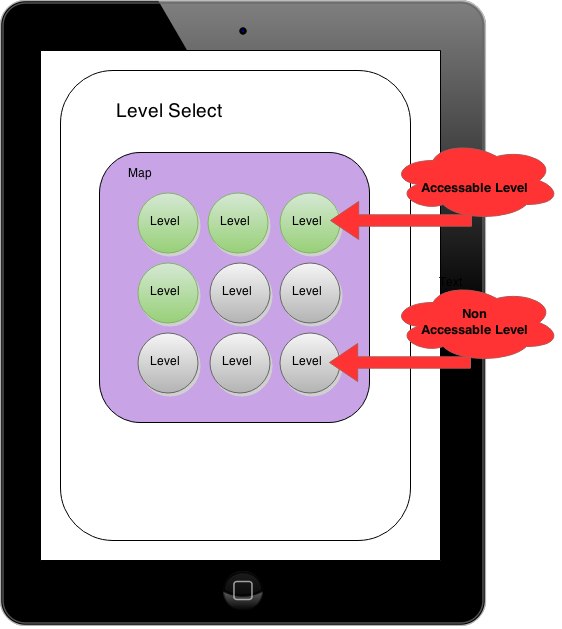
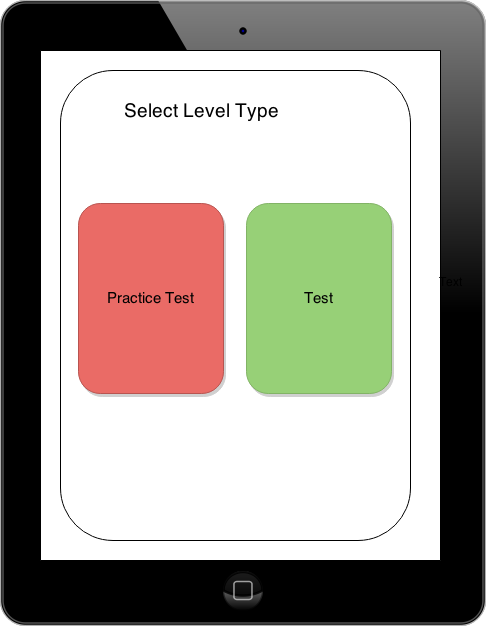
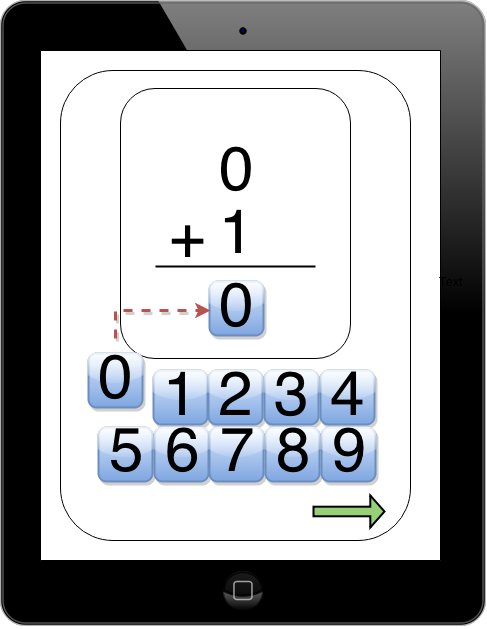


Figure 13 - Select level type screen            Figure 14 - Playable level screen

### **3.1.2 Hardware Interfaces**

The iOS application should be able to interact with any iPad Air and iPad Air 2, that has built-in Wi-Fi and has the required components to connect to the internet.  With this applications native need for internet connectivity, we will be able to communicate to the backend server and database. The communication between the application and the backend server will be facilitated by a third-party web service, which will in turn communicate with our MySQL database.

### **3.1.3 Software Interfaces**

The iOS application will be standalone application running Xcode. It will communicate to our database directly, and will be using database queries. Our database itself, will be built with MySQL, which will send and receive the queries from both the iOS application and the web portal. The web portal will comprise of PHP, HTML, CSS, and JavaScript.

### **3.1.4 Communications Interfaces**

As a student user, they will be able to access our program as an iOS application on the iPad. Upon accessing the application the user will be prompted to enter their login information. If they’re a first time user they will be prompted with a registration screen. The login information will be encrypted, and stored in the MySQL database.

As a teacher, their access will be through the web portal that is directly connected to the database. Access to the web portal will be achieved by using a web browser that is connected to the internet. PHP is the main communication between the teacher and the web portal. When the teacher account has been created, there will be steps to have the user select a class to add and what students to add to the class.  With those settings in place the user will be able to communicate to the device to retrieve data that is captured from the application.

## **3.2 Functional Requirements**

This section describes the specific fundamental features of the Rocket Math project.

### **3.2.1 User Class 1- Student**

#### 3.2.1.1 Functional Requirement 1.1

**ID:** **FR1**

TITLE: Download iOS Application

DESCRIPTION: The student will be able to download the iOS application for their iPad through Apple's app store. The application will be free.   
RATIONAL: In order for the student to use the iOS application.

DEPENDENCY: None

#### 3.2.1.2 Functional Requirement 1.2

**ID:** **FR2**

TITLE: Student Registration

DESCRIPTION: Once the user has the iOS application installed on their iPad, they will be able to create a username and password if they are a first time user for the application.  
RATIONAL: In order for a student to register for an account.

DEPENDENCY: FR1

#### 3.2.1.3 Functional Requirement 1.3

**ID: FR3**

TITLE: Student Log-In

DESCRIPTION: Given that the student has already registered for the application. The student will be able to log into the application and all of the data about the student will be on the iPad. Example, once a student logs in the level they are on will be loaded. Each time a user closes the application, they will be logged out.   
RATIONAL:  In order for a student to login.

DEPENDENCY: FR2

#### 3.2.1.4 Functional Requirement 1.4

**ID: FR4**

TITLE: Student Username/password

DESCRIPTION: If the student forgets username and password to log into the application, the teacher will have access to the specific student’s login credentials. This information can be retrieved through the web portal which the teacher will have access to.   
RATIONAL: In order for a student to retrieve the username and password

DEPENDENCY: FR3

#### 3.2.1.5 Functional Requirement 1.5

**ID: FR5**

TITLE: Application Main Menu

DESCRIPTION: Once the student is logged in, they will be brought to the applications main menu screen. This screen will display what levels they have completed and the current level they are on. The student has the option to play the current level or select one of the previous levels that have been completed.  
RATIONAL: In order for a student to view the main menu.

DEPENDENCY: FR2

#### 3.2.1.6 Functional Requirement 1.6

**ID: FR6**

TITLE: Level Type Selection

DESCRIPTION: Once a user has selected a level to work on they will have to choose if they want to practice the level or take a test to master the current level.  
RATIONAL: In order for the student to choose between a practice and a test.

DEPENDENCY: FR20

#### 3.2.1.7 Functional Requirement 1.7

**ID: FR7**

TITLE: Level - Practice

DESCRIPTION: If the student selects the practice section for a level, they will be asked to solve a set number of questions determined by the teacher. The number of questions will be set by the teacher in the web portal, along with the amount and type of questions.  
RATIONAL: In order for a student to take a practice exam.

DEPENDENCY: FR20, FR21

#### 3.2.1.8 Functional Requirement 1.8

**ID: FR8**

TITLE: Level - Test

DESCRIPTION: If the student selects the test section for a level, they will be asked to solve a set of questions in a set amount of time. The time and the questions will be determined by the teacher in the login page. After a student completes the test with all questions solved correctly, they will be able to unlock the next level.

RATIONAL: In order for a student to take a test.

DEPENDENCY: FR20, FR 21

#### 3.2.1.9 Functional Requirement 1.9

**ID: FR9**

TITLE: Application User Input - Test

DESCRIPTION: If the student selects the test section for a level, they will be asked to solve a set number of questions. The user must answer all of questions correctly within one minute. If the user completes the test, they will advance to the next level. If the user is unable to answer all questions correctly or does not complete the test in under a minute, they will fail the test. The student will be allowed to take one test every 24 hours.   
RATIONAL: In order for a student to input answers for a test.

DEPENDENCY: FR8

#### 3.2.1.10 Functional Requirement 1.10

**ID: FR10**

TITLE: Application User Input - Practice

DESCRIPTION: If the student selects the practice section for a level, they will be asked to solve a set number of questions. If the student answers the question incorrectly, they will be asked to input the operands and the answer in until they get it correct. If a student answers a question correctly, they will be able to move onto the next question.  
RATIONAL: In order for a student to input answers for a practice test.

DEPENDENCY: FR7

#### 3.2.1.11 Functional Requirement 1.11

**ID: FR11**

TITLE: Application Level End

DESCRIPTION: After a student has completed a practice test or actual test level, they will be prompted with their level performance showing each of the questions they solved and the right answers.  
RATIONAL: In order for a student to view their performance at the end of each level.

DEPENDENCY: FR8, FR7

**ID: FR12**

TITLE: iOS Application Locked Levels

DESCRIPTION: If the student attempts to access a level they did not complete, they will be prompted to complete the previously level with 100% accuracy.  
RATIONAL: In order to disable a student’s ability to access un-mastered levels.

DEPENDENCY: FR8

#### 3.2.1.5 Functional Requirement 1.5

**ID: FR13**

TITLE: Application User Input - Keyboard

DESCRIPTION: The iOS application will have a built in drag and drop keyboard. It will not use the standard keyboard provided by iOS. This keyboard will allow students to drag and drop numbers from the screen into the answer box.  
RATIONAL: In order for student to answer questions.

DEPENDENCY: None

### **3.2.2 User Class 2- Teacher**

#### 3.2.2.1 Functional Requirement 2.1

**ID:** **FR14**

TITLE: Teacher Registration - Web Portal

DESCRIPTION: The teacher will be able to register through the web portal to set up an account with a username and password. The username and password will allow the teacher to log in to the web portal. Section 3.1.1.1, figure 3.  
RATIONAL: In order for the teacher to login to the web portal.

DEPENDENCY: None

#### 3.2.2.2 Functional Requirement 2.2

**ID:** **FR15**

TITLE: Teacher Login - Web Portal

DESCRIPTION: The teacher will be able to register through the web portal to set up an account with a username and password. The username and password will allow the teacher to log in to the web portal. Section 3.1.1.1, figure 3.  
RATIONAL: In order for a teacher to register an account through the web portal.

DEPENDENCY: None

#### 3.2.2.3 Functional Requirement 2.3

**ID:** **FR16**

TITLE:   Retrieve password - Web Portal

DESCRIPTION: If the teacher has registered, they will be able to retrieve password by answering security questions.  
RATIONAL: In order for the teacher to retrieve her password.

DEPENDENCY: FR13

#### 3.2.2.4 Functional Requirement 2.4

**ID:** **FR17**

TITLE: Create Class - Web Portal

DESCRIPTION: If the teacher is a first time user, once they create an account they will be brought to a page that allows them to create a class. This page will ask the teacher to enter his/her name, a name for the class and a class password. Section 3.1.1.1, figure 4.  
RATIONAL: In order for a teacher to create a class.

DEPENDENCY: None

#### 3.2.2.5 Functional Requirement 2.5

**ID:** **FR18**

TITLE: Add Students - Web Portal

DESCRIPTION: Once the teacher sets creates a class they will be able to add students to the class through the web portal. This page will have an add student button and a done button. If the teacher clicks the add student button they will be allowed to add the student to the class using the students first and last name. If the teacher does not need to add anyone to the class they can click the done button which will bring them to the teacher view page. Section 3.1.1.1, figure 5.  
RATIONAL: In order for the teacher to add students to the class.

DEPENDENCY: None

#### 3.2.2.6 Functional Requirement 2.6

**ID:** **FR19**

TITLE: Teacher View - Web Portal

DESCRIPTION: The teacher view page within the portal will display a table with each student’s name and which level he/she has completed. The teacher will have the ability to click on a student’s name which will bring up more information on that particular student (FR19). The teacher view page will also have a button to add/remove students to the class and a button that will allow them to add new levels (FR20).  
RATIONAL: In order for the teacher to track the performance of the entire class.

DEPENDENCY: FR19, FR20

#### 3.2.2.7 Functional Requirement 2.7

**ID:** **FR20**

TITLE: Student Progress- Web Portal

DESCRIPTION: Upon clicking on a student in the teacher view, a new view will appear with the particular student’s performance. This view will display how long the student has been working on a level, number of retries and most current snapshot. 3.1.1.1, figure 7.  
RATIONAL: In order to view the current performance of an individual student.

DEPENDENCY: None

#### 3.2.2.8 Functional Requirement 2.8

**ID:** **FR21**

TITLE:  Add Level - Web Portal

DESCRIPTION: When the teacher wants to add a new level they can click the “add level button” in the teacher view. This will bring them to the “add new level view”, from here they can fill in the new level name and description. They will also be able to add new test and practice questions by clicking the Add Test Question or Add Practice Question button; see 3.1.1.1 Figure 8.  
RATIONAL: In order for a teacher to create a new level.

DEPENDENCY: None

#### 3.2.2.9 Functional Requirement 2.9

**ID:** **FR22**

TITLE: Add Question to Level - Web Portal

DESCRIPTION: If the teacher wants to add a new question to the level they can do so by clicking on the add question button within the “add new level view”. Once clicked, the “add new question view” will appear. From this view the teacher can input the question name and description. They also must specify if it is a test or practice question. Also the teacher must input the question and correct answer; see 3.1.1.1 figure 9.  
RATIONAL: In order for a teacher to add new questions to a level.

DEPENDENCY: None

## **3.3 Non-Functional Requirements**

### **3.3.1 Performance**

**ID: NF1**

TITLE: iOS Startup Time

DESCRIPTION: The time it takes the iOS application to boot, pass the splash screen, and enter the main starting screen.

METER: Measurements obtained from 100 startups during testing.

MUST: No more than 25 seconds 100% of the time.

WISH: No more than 15 seconds 100% of the time.

**ID: NF2**

TITLE: iOS Navigating Time

DESCRIPTION: The time is takes the iOS application to navigate through menus.

METER: Measurements obtained from 1000 navigation swipes during testing.

MUST: No more than 4 second 100% of the time.

WISH: No more than 1 second 100% of the time.

**ID: NF3**

TITLE: iOS Entering PlayScreen Time

DESCRIPTION: The time is takes the iOS application to enter the playable level screen.

METER: Measurements obtained from 1000

MUST: No more than 7 seconds 100% of the time.

WISH: No more than 2 seconds 100% of the time.

### **3.3.2 Reliability**

**ID: NF4**

TITLE: Question Reliability

DESCRIPTION: The reliability that the system returns the students with the correct answer.

MUST: 100% accuracy.

WISH: 100% accuracy.

### **3.3.3 Availability**

**ID: NF5**

TITLE: iOS Internet Connection

DESCRIPTION: The iOS application should be connected to the Internet.  
RATIONAL: In order for the iOS application to communicate to the database.

DEPENDENCY: The user must have an internet connection with an ISP.

**ID: NF6**

TITLE: Web Portal Internet Connection

DESCRIPTION: The web portal should be connected to the Internet.  
RATIONAL: In order for the web portal to communicate to the database.

DEPENDENCY: The user must have an internet connection with an ISP.

### **3.3.4 Security**

**ID: NF7**

TITLE: Web portal Encrypted Messages Security

DESCRIPTION: The security of the communication between the web portal and the server.

SCALE: The login information being sent between the web portal and the server should be encrypted so unauthorized people are unable to access them.

MUST: 100% of the time.

**ID: NF8**

TITLE: iOS Encrypted Messages Security

DESCRIPTION: The security of the communication between the iOS application and the server.

SCALE: The login information being sent between the iOS application and the server should be encrypted so unauthorized people are unable to access them.

MUST: 100% of the time.

**ID: NF9**

TITLE: Teacher Login Account Security

DESCRIPTION: Security of logging into the accounts used within the web portal.  
SCALE: If a teacher attempts to a login to the web portal, without a pre-existing account, then they should not be logged in. Instead they should be prompted with a failure and asked to register an account.

MUST: 100% of the time.

**ID: NF10**

TITLE: Student Login Account Security

DESCRIPTION: Security of logging into the accounts used within the iOS application.  
SCALE: If a student attempts to login to the iOS application, without a pre-existing account, then they should not be logged in. Instead they should be prompted with a failure and asked to register account.

MUST: 100% of the time.

**ID: NF11**

TITLE: Teacher Create Account Security

DESCRIPTION: Security of the creating accounts used within the web portal.  
SCALE: If a teacher attempts to create an account and the desired username is already taken, the teacher should be prompted of this failure.

MUST: 100% of the time.

**ID: NF12**

TITLE: Student Create Account Security

DESCRIPTION: Security of the creating accounts used within the iOS application.

SCALE: If a student attempts to create an account and the desired username is already taken, the student should be prompted of this failure.

MUST: 100% of the time.

**ID: NF13**

TITLE: Teacher Login Limit Security

DESCRIPTION: Security of the limiting login attempts within the web portal.

SCALE: A teacher account and an IP address should not be able to log-in to the web portal for a certain time period after 5 failed login attempts.

MUST: 100% of the time.

**ID: NF14**

TITLE: Student Login Limit Security

DESCRIPTION: Security of the limiting login attempts within the iOS application.

SCALE: A student account and an IP address should not be able to log-in to the iOS application for a certain time period after 5 failed login attempts.

MUST: 100% of the time.

### **3.3.5 Maintainability**

**ID: NF15**

TITLE: System Extendibility

DESCRIPTION: The system as a whole should allow the teacher to maintain the question information, along with level information.  
RATIONAL: In order to increase the lifespan of the software.

DEPENDENCY: None

## **3.4 Design Constraints**

This section contains the design restraints caused by the hardware and software.

**ID: DC 1**

TITLE: iOS Application Hard Drive Space

DESCRIPTION: The size of the iOS application when downloaded from the app store to device.

SCALE: MB.

MUST: No more than 4000 MB. Which is the maximum apple will allow.

WISH: No more than 50 MB.

**ID: DC 2**

TITLE: iOS Version on device

DESCRIPTION:  This is the required iOS version to run the application

SCALE: iOS Version.

MUST: The device must be using at least iOS 7.

WISH: The device is using iOS 8.

**ID: DC 3**

TITLE: Accessing Web Portal

DESCRIPTION: The internet browser used to access the web portal.

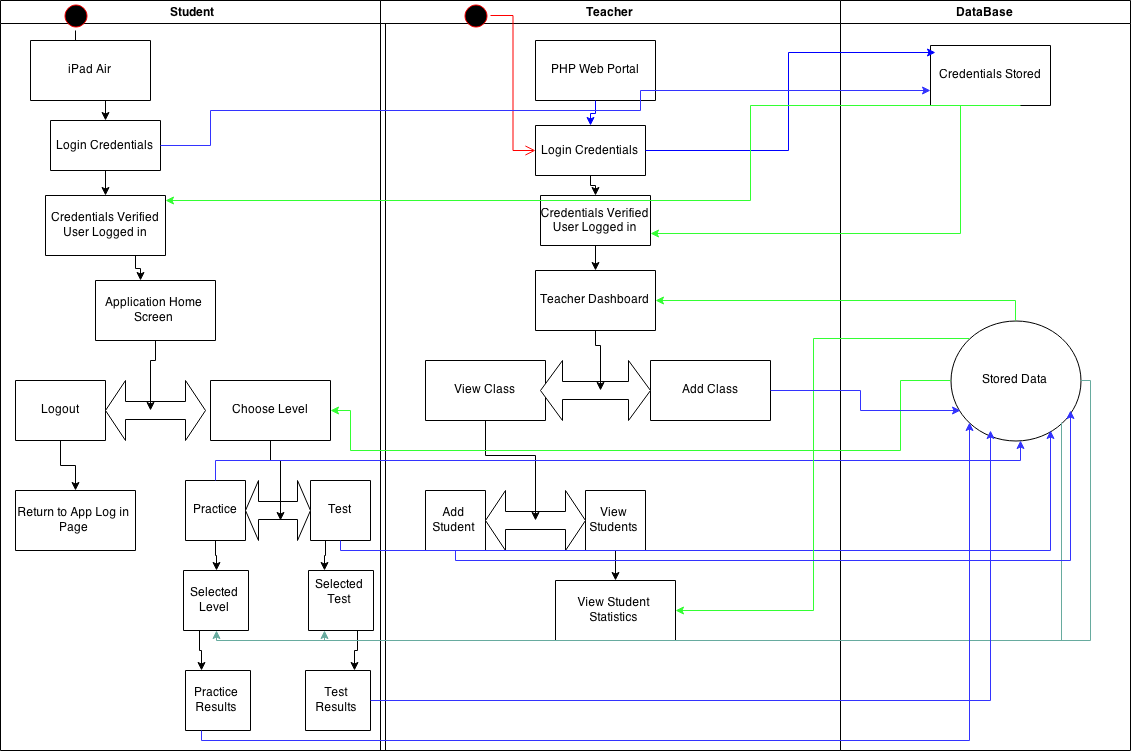
SCALE: Internet browser.

MUST: In order to successfully use the web portal the teacher must be using Google Chrome.

WISH: The portal can be access through Google Chrome, Apple Safari, Microsoft Internet Explorer and Mozilla Firefox.

# **4. Analysis Models**

## **4.1 Data Flow Diagrams (DFD)**

****

## **4.2 Prioritization and Release Plan**

In order to figure out how to understand which requirements will be built into which prototypes, a prioritization of requirements is needed. This section covers the choices the team made for the prioritization of requirements, and shows an estimated plan for when each prototype and requirement will be developed and released. Appendix I, II, and III show the priority of the requirements and the intended release plan.

### **4.2.1 Choosing the Priority**

While prioritizing the requirements the team decided to pick out top 20 requirements that were most important for defining what the system actually did. These 20 requirements were heavily based on the two ender user’s perspectives, the teacher and the student. 13 of these requirements ended up being functional requirements, while the remaining were nonfunctional. The remaining requirements were prioritized by most likely needed for students and teachers to use the system, to least likely. The requirements with lower priority were performance requirements, while the higher priority requirements were additional functionality, such as the ability of the teacher to add questions to levels.

# **Appendix**

## **Appendix I: Priority Scheme**

|  |  |
| --- | --- |
| **Priority:** | **Priority:** |
| 1 | ID: NF6 |
| 2 | ID: NF5 |
| 3 | ID: FR9 |
| 4 | ID: FR10 |
| 5 | ID: FR11 |
| 6 | ID: FR5 |
| 7 | ID: FR18 |
| 8 | ID: FR17 |
| 9 | ID: FR19 |
| 10 | ID: FR20 |
| 11 | ID: FR21 |
| 12 | ID: FR16 |
| 13 | ID: FR22 |
| 14 | ID: NF4 |
| 15 | ID: FR7 |
| 16 | ID: FR8 |
| 17 | ID: FR6 |
| 18 | ID: NF1 |
| 19 | ID: NF2 |
| 20 | ID: NF3 |
| 21 | ID: NF15 |
| 22 | ID: FR1 |
| 23 | ID: FR4 |
| 24 | ID: FR2 |
| 25 | ID: FR3 |
| 26 | ID: NF7 |
| 27 | ID: NF8 |
| 28 | ID: NF13 |
| 29 | ID: NF9 |
| 30 | ID: NF10 |
| 31 | ID: NF11 |
| 32 | ID: NF12 |
| 33 | ID: NF14 |
| 34 | ID: FR12 |
| 35 | ID: FR15 |
| 36 | ID: FR13 |
| 37 | ID: FR14 |
| 38 | ID: DC 3 |
| 39 | ID: DC 1 |
| 40 | ID: DC 2 |

## **Appendix II: Release Plan**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement:** | **Title:** | **Description:** | **Prototype release:** | **Duration:** |
| FR1 | Download iOS Application | The user will be able to download the iOS application for their iPad through Apple's app store. The application will be free. | 2 | 13 |
| FR2 | Student Registration | Once the student has the iOS application installed on their iPad, they will be able to create a username and password if they are a first time user for the application. | 1 | 4 |
| FR3 | Student Log-In | Given that the student has already registered for the application. The student will be able to log into the application and all of the data about the user will be on the iPad. Example, once a student logs in the level they are on will be loaded. Each time a user closes the application, they will be logged out. | 1 | 4 |
| FR4 | Student Username/password | If the student forgets username and password to log into the application, the teacher will have access to the specific student’s login credentials. This information can be retrieved through the web portal which the teacher will have access to. | 2 | 13 |
| FR5 | Application Main Menu | Once the student is logged in, they will be brought to the applications main menu screen. This screen will display what levels they have completed and the current level they are on. The student has the option to play the current level or select one of the previous levels that have been completed. | 2 | 13 |
| FR6 | Level Type Selection | Once a user has selected a level to work on they will have to choose if they want to practice the level or take a test to master the current level. | 2 | 13 |
| FR7 | Level - Practice | If the user selects the practice section for a level, they will be asked to solve a set number of questions determined by the teacher. There is no time limit on how long it takes the user to answer these questions. 40% of these questions will be from the current level, 15% will come from the previous level. The remaining 45% will come from any level except the current and previous level. | 2 | 13 |
| FR8 | Level - Test | If the user selects the test section for a level, they will be asked 15 questions. The user must answer all 15 questions correctly within one minute. If the user completes the test, they will advance to the next level. If the user is unable to answer all 15 questions correctly or does not complete the test in under a minute, they will fail the test. User will be allowed to take one test every 24 hours. | 2 | 13 |
| FR9 | Application User Input | If the user selects the test section for a level, they will be asked 15 questions. The user must answer all 15 questions correctly within one minute. If the user completes the test, they will advance to the next level. If the user is unable to answer all 15 questions correctly or does not complete the test in under a minute, they will fail the test. User will be allowed to take one test every 24 hours. | 3 | 13 |
| FR10 | Application User Input - Correct | If the user selects the test section for a level, they will be asked 15 questions. The user must answer all 15 questions correctly within one minute. If the user completes the test, they will advance to the next level. If the user is unable to answer all 15 questions correctly or does not complete the test in under a minute, they will fail the test. User will be allowed to take one test every 24 hours. | 3 | 13 |
| FR11 | Application User Input - Incorrect | If the user selects the test section for a level, they will be asked 15 questions. The user must answer all 15 questions correctly within one minute. If the user completes the test, they will advance to the next level. If the user is unable to answer all 15 questions correctly or does not complete the test in under a minute, they will fail the test. User will be allowed to take one test every 24 hours. | 3 | 13 |
| FR12 | iOS Application Locked Levels | If the student attempts to access a level they did not complete, they will be prompted to complete the previously level with 100% accuracy. | 2 | 13 |
| FR13 | Application User Input - Keyboard | The iOS application will have a built in drag and drop keyboard. It will not use the standard keyboard provided by iOS. This keyboard will allow students to drag and drop numbers from the screen into the answer box. | 2 | 13 |
| FR14 | Teacher Registration - Web Portal | The teacher will be able to register through the web portal to set up an account username and password. This username and password will allow the teacher to log in to the web portal. Section 3.1.1.1, figure 3. | 1 | 4 |
| FR15 | Teacher Login - Web Portal | The teacher will be able to register through the web portal to set up an account username and password. This username and password will allow the teacher to log in to the web portal. Section 3.1.1.1, figure 3. | 1 | 4 |
| FR16 | Retrieve password - Web Portal | If the teacher has registered, they will be able to retrieve password by answering security questions. | 3 | 13 |
| FR17 | Create Class - Web Portal | If the teacher is a first time user, once they create an account they will be brought to a page that allows them to create a class. This page will ask the teacher to enter his/her name, a name for the class and a class password. Section 3.1.1.1, figure 4. | 2 | 10 |
| FR18 | Add Students - Web Portal | Once the teacher sets creates a class they will be able to add students to the class through the web portal. This page will have an add student button and a done button. If the teacher clicks the add student button they will be allowed to add the student to the class using the students first and last name. If the teacher does not need to add anyone to the class they can click the done button which will bring them to the teacher view page. Section 3.1.1.1, figure 5. | 2 | 10 |
| FR19 | Teacher View - Web Portal | The teacher view page within the portal will display a table with each student’s name and which level he/she has completed. The teacher will have the ability to click on a student’s name which will bring up more information on that particular student (FR19). The teacher view page will also have a button to add/remove students to the class and a button that will allow them to add new levels (FR20). | 2 | 10 |
| FR20 | Student Progress- Web Portal | Upon clicking on a student in the teacher view, a new view will appear with the particular student’s performance. This view will display how long the student has been working on a level, number of retries and most current snapshot. 3.1.1.1, figure 7. | 2 | 10 |
| FR21 | Student Progress- Web Portal | When the teacher wants to add a new level they can click "the add level button” in the teacher view. This will bring them to the add new level view, from here they can fill in the new level name and description. They will also be able to add new test and practice questions by clicking the Add Test Question or Add Practice Question button. 3.1.1.1 figure 8. | 2 | 10 |
| FR22 | Add Question to Level - Web Portal | If the teacher wants to add a new question to the level they can do so by clicking on the add question button within the add new level view. Once clicked a add new question view will appear. From this view the teacher can input the question name and description. They also must specify if it is a test or practice question. Also the teacher must input the question and correct answer. 3.1.1.1 figure 9. | 2 | 10 |
| NF1 | iOS Startup Time | The time it takes the iOS application to boot, pass the splash screen, and enter the main starting screen. | 3 | 15 |
| NF2 | iOS Navigating Time | The time is takes the iOS application to navigate through menus. | 3 | 15 |
| NF3 | iOS Entering PlayScreen Time | The time is takes the iOS application to enter the playable level screen. | 3 | 15 |
| NF4 | Question Reliability | The reliability that the system returns the students with the correct answer. | 2 | 10 |
| NF5 | iOS Internet Connection | The iOS application should be connected to the Internet. | 1 | 10 |
| NF6 | Web Portal Internet Connection | The web portal should be connected to the Internet. | 1 | 10 |
| NF7 | Web portal Encrypted Messages Security | The security of the communication between the web portal and the server. | 2 | 13 |
| NF8 | iOS Encrypted Messages Security | The security of the communication between the iOS application and the server. | 2 | 13 |
| NF9 | Teacher Login Account Security | Security of logging into the accounts used within the web portal. | 2 | 13 |
| NF10 | Student Login Account Security | Security of logging into the accounts used within the iOS application. | 2 | 13 |
| NF11 | Teacher Create Account Security | Security of the creating accounts used within the web portal. | 2 | 13 |
| NF12 | Student Create Account Security | Security of the creating accounts used within the iOS application. | 2 | 13 |
| NF13 | Teacher Login Limit Security | Security of the limiting login attempts within the web portal. | 2 | 13 |
| NF14 | Student Login Limit Security | Security of the limiting login attempts within the iOS application. | 2 | 13 |
| NF15 | System Extendibility | The system as a whole should allow the teacher to maintain the question information, along with level information. | 1-3 | 64 |
| DC1 | iOS Application Hard Drive Space | The size of the iOS application when downloaded from the app store to device. | 1 | 10 |
| DC2 | iOS Version on device | This is the required iOS version to run the application | 1 | 10 |
| DC3 | Accessing Web Portal | The internet browser used to access the web portal. | 1 | 10 |
|  |  |  |  |  |

## **Appendix III: Project Plan**

