QUIZ GAME

BOGDAN KOTZEV, BRENDAN MURRAY, JOSH HOCHULI, & TYLER KOSMACKI

JANUARY 29, 2016

PROJECT PROPOSAL

User Stories	Rank
Questions As a user I want to be able to answer questions So that I can assess my knowledge	1
Tags As a user I want questions to be tagged with categories So that I can track my proficiency in specific areas	2
Submit Page As a user I want to submit questions So I can expand on the question base	3
Review Potential Questions As a user I want to screen potential questions So that added questions can be verified for quality	4
Responsiveness As a user I want a responsive application So that I can use it on a computer or a mobile device	5
Profile As a user I want a profile page So that I can track my progress	6
Administration As a programmer I want to implement access control So that users can be students or administrators	7
Category Statistics As a user I want to see my strengths and weaknesses So I can know what to study	8

Instant Results As a user I want instant feedback So that I know If I got the questions right	9
Leaderboard As a user I want a leader board So that I can compare my progress with others	10
Groupings As a user I want to be able to group users together based on school So that I can see where my school ranks	11
Pharma Points As a user I want a points system So that I can see my progress and achievements	12
Multiple Choice As a user I want multiple choice questions So that I don't have to use the keyboard	13
Loading Page As a user I want to see a loading screen with neat quotes So that I can be inspired	14
Network Map As a user I want to see a network map So I can know who is working around me	15

Determining User Stories

We met with our mentor, Ravi Patel, on January 22nd. The primary goal of the meeting was determining a list of requirements with him that we could elicit user stories from. Rather than hammer out a list of discrete specifications we talked more generally about the functionality of the application. After our meeting we used the notes we had taken to create the list of user stories above: they represent the functional and nonfunctional requirements that we discussed.

We determined user story rankings based on their importance to the application, organized by core features taking highest precedence. The primary goal of this app is to introduce and reinforce topics that pharmacy students are tested on and it is no coincidence that our highest ranked user stories all concern the quiz questions themselves. Most of these user stories come from functional requirements and are therefore testable and verifiable, but others are about traits that we want the application to have.

The most important non functional requirement we have is that this application should be responsive. Users are more likely to use an application that is cross platform and rather than make iPhone or Android app, or even a OS X or Windows program, we have decided that a website that is useable on many different devices is optimal. This is by far the most important quality that we want the application to have. Another performance trait is that users will get instant results for questions that they answer, but this is a much easier feature to implement than responsive web design. This will be a challenge but we believe it to be worth the effort.

The user stories with the lowest rankings are either trivial to implement or so difficult that we may not be able to complete them. The network map would likely be the last thing we work on based on its difficulty to usefulness ratio. While it is a nice idea we are not going to devote time to a non essential feature while there are more important user stories to complete.

System Design Outline

While it will be challenging, we have decided that creating a fully responsive web application is the best direction to go in. We want users to be able to be able to use our quiz game on any platform they desire, whether it be on a personal computer or a mobile device. For this reason we have decided to create a responsive web application using the following languages: javascript, java, SQL, HTML and CSS. The front end portions will be in javascript and HTML/CSS while the backend will contain a SQL database that we access with java.

The quiz questions will have tagged categories so that users can see what areas they are strong in and what areas they need to improve on. They can also be specialized to be part of a specific curriculum. A student could create a question bank tailored to a single class which could then be used by the rest of their class as well. We want to encourage this sort of collaboration with a points system, leaderboard rankings, and a profile that displays progress and accolades. This application would benefit tremendously from power users who actively create and review potential quiz questions so we want to encourage and facilitate that.

We want a multiple choice format that can work on any size screen. This design is easy enough to implement with javascript's built in radio buttons and this would be great for a phone. The questions themselves can be of varying formats but we think a lightweight multiple choice system is the best way to go: there is no fill in the blank information to parse and input is straightforward. Questions, whether they be text or incorporate pictures, will need to be stored somewhere, so we plan to have a back end database to store them in. There are considerable data storage realities we need to plan for so our backend is as important as our front end.
As we get a clearer idea of the type of data we will need to store we will narrow down the architecture of our database.

Testing Plan

The majority of the testing of this system will need to be done on the front end, as there will not be a lot of work being done with the database. We will want to thoroughly test the question submission platform as this will be the main way that the system will grow. This should probably be done through some basic unit testing of the input fields and maybe additional testing should be done to make sure these values are being stored correctly on the back end, seeing as there will be multiple input fields for correctly handling the questions and all the possible multiple choice selections.

We will also go through some small user testing sessions, where, once we have a functional base application, a small selection of students or administrators from the Pharmacy school will have the option of doing some basic exploratory testing on the system to see if it is done in an intuitive manner.

Walking Skeleton

https://github.com/BrendanMurray/cs1980-QG