**Undefined** **{?} Parameter**

By Melynda Lindhorst, Blake Bartlett, Joseph Lee, Braeden Bodily

**Senior Capstone Project:**

**Design Document**

**Use Cases and UI Sketches**

Requirements

Section 1: System Architecture

1. User Interface

This is the user’s interaction with the system. In order for QuizZing to provide competitive services, the interface needs to be intuitive and easy to use. To accommodate this, it will be designed to allow features to be used with minimal clicks and screens. For example, the quiz creating will have an “Auto Generate” button that can be clicked without selecting anything, thus allowing quizzes to be created with ease. The interface will be designed using HTML and CSS, supported with JavaScript.

1.2 Network Communication

As a web based application, QuizZing will make use of a server to process requests from web browsers and return HTML responses for the browser to display to the user. To facilitate easy to use, and robust communication, QuizZing will be built on the Dropwizard framework. Dropwizard uses built in frameworks like Jetty, and will provide the functionality for processing HTML requests, and allow us to dynamically parse the data and display it in predefined HTML views.

1.1 Question and Quiz Interaction

This component is responsible for allowing communication between QuizZing members and groups. It also provides functionality for the users to create, edit, and take quizzes. It will make use of categories, and tags to allow users to define specific quizzes to meet their individual needs. Through the use of ratings and flags, the users of QuizZing have the power to maintain the content of their individual groups, allowing scaleability and targeted content. This will provide competitive play and cooperative question and quiz creation between QuizZing users.

1.2 Quizzing Analytic Services

This will be responsible for tracking user interaction with QuizZing to allow the system to provide needed content to the user. An example of this, is if a user is consistently getting specific questions wrong in a quiz, the system will recognize and provide the user with more questions matching those categories. This will assist the learning process that will make QuizZing a tool instead of a game. This will enable smart quiz generation, and allow us to provide statistical analysis, such as bar graphs showing progress, for a user or their group.

1.3 Database

This component will store all of our data in a logical way. There will be multiple tables hosting users, groups, questions, and quizzes. The tables will be constructed to promote the use of our tag and rating systems for quiz generation. The different question types will also be stored in a way that suits them best.

Section 2: Personnel

2.0 Undefined Parameter Team

* **Blake Bartlett** – Blake’s focus will be dedicated to networking and database. He will handle back end communication from the web browser, as well as storing and retrieving transactions to and from the database.
* **Braeden Bodily** – Braeden will be focusing on the User Interface of the system. He will be working to design and implement the HTML front end of the system.
* **Joseph Lee** – Joseph will primarily be working on the back end of the system. This involves intercommunication between the database and views, as well as communication between the quiz system and the view.
* **Melynda Lindhorst** – Melynda’s primary task will be to design the quiz interactions in the system. She will be working on algorithms to implement auto quiz generation and many other features.

Section 3: System Features

3.0 Rank 1 – Our system cannot function without these features.

* **User Interface: Web Services**
  + Dependencies: Dropwizard server, HTML Views
  + Co-dependencies: SQL Database
  + Component: User Interface
* **Database Configuration**
  + Dependencies: SQL Database
  + Co-dependencies: User Interface
  + Component: Database
* **Multiple Choice Question Creation**
  + Dependencies: User Interface, Database
  + Co-dependencies: Quiz Interaction
  + Component: Question Interaction
* **Quiz Interaction**
  + Dependencies: User Interface, Database
  + Co-dependencies: Question Interaction
  + Component: Quiz Interaction

3.1 Rank 2 – Our system will look like a finished product by implementing these features.

* **Additional Question Type Creation**
  + Dependencies: User Interface, Database
  + Co-dependencies: Question Creation
  + Component: Question Interaction
* **Quiz Creation**
  + Dependencies: User Interface, Database
  + Co-dependencies: Q uiz Interaction
  + Component: Quiz Interaction
* **Register Members**
  + Dependencies: User Interface, Database
  + Co-dependencies: Security
  + Component: Services
* **Security**
  + Dependencies: Database, Web Server
  + Co-dependencies: None
  + Component: Services
* **Member Interaction**
  + Dependencies: Register Members
  + Co-dependencies: None
  + Component: Services
* **Tag System**
  + Dependencies: Quiz and Question Creation
  + Co-dependencies: Analytics
  + Component: Quiz and Question Interaction, Services
* **Analytics**
  + Dependencies: Member Interaction, Question and Quiz Creation
  + Co-dependencies: None
  + Component: Services

3.2 - Rank 3: Our system feel polished and professional as well as interactive and fun if these features are implemented.

* **User Interface: Android**
  + Dependencies: Dropwizard Server
  + Co-dependencies: None
  + Component: User Interface
* **Group Interaction**
  + Dependencies: Register Members
  + Co-dependencies: Group Quiz Generation
  + Component: Services
* **Group Quiz Generation**
  + Dependencies: Quiz Generation
  + Co-dependencies: Group Interaction
  + Component: Question and Quiz Interaction
* **Competitive Play**
  + Dependencies: Member Interaction
  + Co-dependencies: Titles/Badges
  + Component: Services
* **Titles/Badges**
  + Dependencies: Quiz and Question Interaction
  + Co-dependencies: Competitive Play
  + Component: Services