****

**DOCUMENTATION**

**Quiz Project**

**(Advanced Java Project)**

November 2018

**Submitted By**

**Sitharaman Deepak Guptha**

Contents

[1. Subject Description: 3](#_Toc530795522)

[2. Subject Analysis:   
 2.1 Major Features: 3](#_Toc530795523)

[2.2 Application Features: 4](#_Toc530795524)

[2.3 Data Description: 4](#_Toc530795528)

[2.4 Expected Results: 5](#_Toc530795572)

[2.5 Scope of the application (limits and evolutions): 5](#_Toc530795576)

[3. Conception: 6](#_Toc530795585)

[3.1 Chosen algorithm: 6](#_Toc530795586)

[3.2 Data structures: 6](#_Toc530795589)

[3.3 Global application flow: 7](#_Toc530795591)

[3.4 Global schema and major features schema: 7](#_Toc530795592)

[4. Web operations description: 8](#_Toc530795593)

[4.1 Creation of questions and the choices. 8](#_Toc530795594)

[4.2 Updation of Question and Choices: 8](#_Toc530795595)

[4.3 View all: 8](#_Toc530795597)

[4.4 Delete a Question: 8](#_Toc530795599)

[5. Configuration Instructions: 8](#_Toc530795601)

[5.1 Development Tools: 8](#_Toc530795602)

[6. Commented Screenshots: 8](#_Toc530795607)

[7. Bibliography 9](#_Toc530795608)

# 1. Subject Description:

This project implements the features of **Quiz Project**. The system is implemented with the features of Questions and Answers.

The Quiz Project has the following operations that allow the user to perform **Create, Search, Update, Delete and View All** – Questions and the Choices

The Quiz Project has been developed using JAVA language, allows the user to perform actions through the web page.

**Technology Used:** JAVA, Maven, Spring JAX-RS, Hibernate and the Derby database.

**Front End Technology Used:** HTML 5, Bootstrap 3.3.7 (CSS Framework) and jQuery (JavaScript library).

# 2. Subject Analysis: **2.1 Major Features:**

The major features of the **Quiz Project** are,

* **Creation of the Questions and Choices:**

This feature allows the user to create or insert a question and their choices (answers) with the valid status (True or False).

* **Search Question:**The user can search a question by entering the question label or the keyword of the Question.
* **View all Question and Choices:**

Allows the user to view all the Questions and Choices with their respected IDs.

* **Update Question and Choices:**

The user can update an existing question and choices by providing the Questions, Choices and IDs.

* **Delete a Question:**

The user can delete a question and choices by providing the question ID.

* **Dashboard page:**

The user can navigate to all the web pages, by clicking the Create, View All, Update, Search and Delete buttons in the Dashboard page.

2.2 Application Features:

* The application allows the user to create, update, delete and search the Questions & Choices through the web pages and the data is stored in the database.
* The tools used in building this application are JAVA (JDK & JRE), Spring JAX-RS Framework, Maven, Eclipse IDE and Apache Derby database.
* It has the features of Create, Search, Update, Delete and View All operations performed by the user (Question and Answers)and the application provides success and error messages to the user.

2.3 Data Description:

The ‘Question’ table and ‘MCQChoice’ table data stored in the Derby database automatically generated by the Hibernate.

Question Table:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Data Type | Field Length | Primary Key | Field Description |
| ID | BIGINT | - | Yes | Auto incremented |
| QUESTIONLABEL | VARCHAR | 255 | No | Storing the Question Label |

MCQChoice Table:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Data Type | Field Length | Primary Key | Field Description |
| ID | BIGINT | - | Yes | Auto incrementing the number. |
| CHOICELABEL | VARCHAR | 255 | No | Storing the Choice Label |
| VALID | BOOLEAN | - | No | Storing the Choice Status |
| QUESTION\_ID | BIGINT | - | No (Foreign Key) | Stores the Question ID |

2.4 Expected Results:

The user can create, update, search, delete and view all the Questions & Choices through the web pages.

Messages:

Success and error messages are displayed for each operation through alerts in the web pages.

2.5 Scope of the application (limits and evolutions):

The web application core functionality is performing the DAO or CRUD operations of Questions and Choices, the datas are stored in the Derby database.

Limitation:

* The web application uses the jQuery (JavaScript) library and the Bootstrap framework.
* There is no creation of users for the admin and the student login.

Evolutions:

* Implementation of the Web application by Angular, React and Vue.js framework.
* User roles can be maintained for the Admin and the Students.
* Functionalities like implementation of timer, uploading of Questions and Choices through csv, dashboard pages for the admin and the students, results of the test.

# 3. Conception:

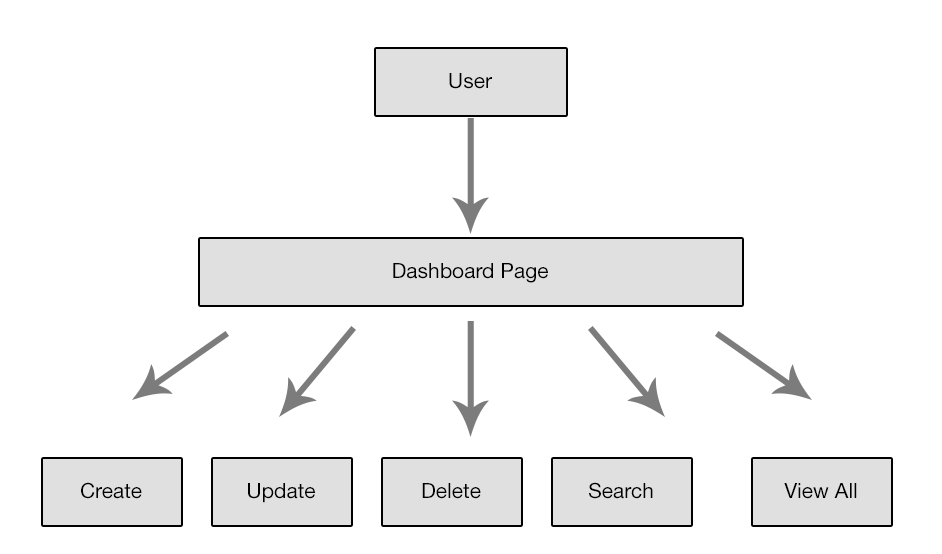
3.1 Chosen algorithm:

* Factory pattern has been used to implement the code.
* Log4j2 package has been implemented in this application for the logs.

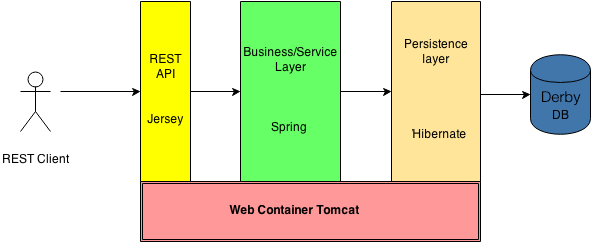
3.2 Data structures:

The data structures used in this application are Lists and Maps. List which is used to store the list of Questions and Choices. Maps used to map the Questions with the choices.

3.3 Global application flow:

****

3.4 Global schema and major features schema:

******

# 4. Web operations description:

The following operations are implemented in this web application are,  
  
4.1 Creation of Question and Choices:

* This operation is used to insert the questions and the choices.

4.2 Updation of Question and Choices:

* This operation is used to update the questions and the choices by the id.

4.3 View all:

* This operation is used to view all the questions and choices with the id.

4.4 Delete a Question:

* This operation is used to delete the question and choices by the question id.

# 5. Configuration Instructions:

5.1 Development Tools:

* Java 8 (64 bit) (Java Development Kit and Java Runtime Environment) installed.
* Eclipse Java EE IDE for Web Developers (Version: Oxygen.3a Release – 4.7.3a).
* Maven, Spring JAX-RS, Hibernate,
* Apache Derby Database.

# 6. Commented Screenshots:

The commented screenshots for the operations are included in the User Guide document.

# 7. Bibliography

* Project Scope, JAVA tutorials concept

**Link:** <http://thomas-broussard.fr/work/java/tutorials/index.xhtml>

* JAVA Oracle Installation Docs

**Link:** [https://docs.oracle.com/javase/9/install/overview-jdk-9-and-jre-9-installation.htm#JSJIG-GUID-8677A77F-231A-40F7-98B9-1FD0B48C346A](https://docs.oracle.com/javase/9/install/overview-jdk-9-and-jre-9-installation.htm%23JSJIG-GUID-8677A77F-231A-40F7-98B9-1FD0B48C346A)

* Apache Derby Database Docs

**Link:** <https://db.apache.org/derby/docs/10.0/manuals/develop/develop02.html>

* Spring JAX-RS

**Link:** <http://cxf.apache.org/docs/jax-rs-client-api.html>

* Maven Dependencies

**Link:** https://mvnrepository.com/

* Log4j 2 Docs

**Link:** <https://logging.apache.org/log4j/2.x/download.html>

<https://examples.javacodegeeks.com/enterprise-java/log4j/log4j-2-best-practices-example/>