CS5704 Software Engineering

Semester Project Report

Spring 2021

**iQuiz**

Calvin Huang

ZhengBo Wang

Lin Zhang

Department of Computer Science

Virginia Tech

Blacksburg, VA 24061

*App's URL:* on Amazon AWS EC2 Virtual Private Server

*Date:* 02/28/2021

*Team Number:* Team 3

*Instructor:* Prof. Osman Balci

**USER ACCOUNTS**

Using the deployed software, each team member is required to create an account and generate *meaningful content* in the database for evaluation and testing. **You will be penalized for not having sufficient content**.

In the case of two-factor authentication, your software is required to provide a Bypass Two-Factor Authentication option for evaluation and testing.

List the usernames and passwords under this option below.

|  |  |
| --- | --- |
| *Username* | *Password* |
| Team Member 1's username | Team Member 1's password |
| Team Member 2's username | Team Member 2's password |
| Team Member 3's username | Team Member 3's password |

This page must be numbered as ii. No page number must appear on the title page.

**EXECUTIVE SUMMARY**

TBD

\*\*\*\*

**ATTENTION**: Delete all instructions in red before submitting the final version of the report.

===> Subtract 5 points out of 100 if any instruction is left in the final version. <===

\*\*\*\*

Provide a structured summary of the content of the document here, by emphasizing your important contributions to catch the attention of a very busy executive.

**List of Implemented Cloud Software Features Learned** (At least 3 required per team member)

1. TBD
2. TBD
3. TBD
4. TBD
5. TBD
6. TBD
7. TBD
8. TBD
9. TBD

**List of Implemented New Cloud Software Features (**At least 1 required per team member)

1. TBD
2. TBD
3. TBD

This page must be numbered as iii.

**TABLE OF CONTENTS**

[**USER ACCOUNTS ii**](#_heading=h.tyjcwt)

[**EXECUTIVE SUMMARY iii**](#_heading=h.3dy6vkm)

[**1.**](#_heading=h.4d34og8) **SOFTWARE LIFE CYCLE 1**

[**2.**](#_heading=h.2s8eyo1) **OUR OBJECTIVE 1**

[**3.**](#_heading=h.17dp8vu) **PROBLEM SPECIFICATION 2**

[3.1](#_heading=h.3rdcrjn) What is the problem? 2

[3.2](#_heading=h.26in1rg) Why is the problem important to solve? 2

[3.3](#_heading=h.lnxbz9) Expected Functionality Description 2

[3.4](#_heading=h.35nkun2) Learned and new cloud software features to be implemented 2

[**4.**](#_heading=h.44sinio) **REQUIREMENTS SPECIFICATION 4**

[4.1](#_heading=h.2jxsxqh) Functional Requirements 4

[4.2](#_heading=h.z337ya) Non-Functional Requirements 4

[**5.**](#_heading=h.3j2qqm3) **ARCHITECTURE SPECIFICATION 5**

[**6.**](#_heading=h.1y810tw) **DESIGN SPECIFICATION 5**

[**7.**](#_heading=h.4i7ojhp) **DELIVERED SOFTWARE FUNCTIONALITY 6**

[**8.**](#_heading=h.2xcytpi) **STEP-BY-STEP INSTRUCTIONS FOR DEVELOPING NEW FEATURES 7**

[8.1](#_heading=h.1ci93xb) New Feature 1 Title 7

[8.2](#_heading=h.3whwml4) New Feature 2 Title 7

[8.3](#_heading=h.2bn6wsx) New Feature 3 Title 7

[8.4](#_heading=h.qsh70q) New Feature N Title 8

[**9.**](#_heading=h.3as4poj) **CONCLUSIONS 8**

[**10.**](#_heading=h.1pxezwc) **SUBMISSION INSTRUCTIONS 8**

[**11.**](#_heading=h.49x2ik5) **PERCENTAGES OF CONTRIBUTION 10**

[**12.**](#_heading=h.147n2zr) **CALCULATION OF GRADES BASED ON PERCENTAGES OF CONTRIBUTION 11**

[**13.**](#_heading=h.3o7alnk) **TEAM PROJECT REQUIREMENTS 11**

[**14.**](#_heading=h.23ckvvd) **GRADING SHEET 12**

[**REFERENCES 13**](#_heading=h.ihv636)

[**APPENDIX A: MEETING MINUTES 14**](#_heading=h.32hioqz)

Each entry in the TOC above must be a hyperlink clickable to jump to that section in the report.

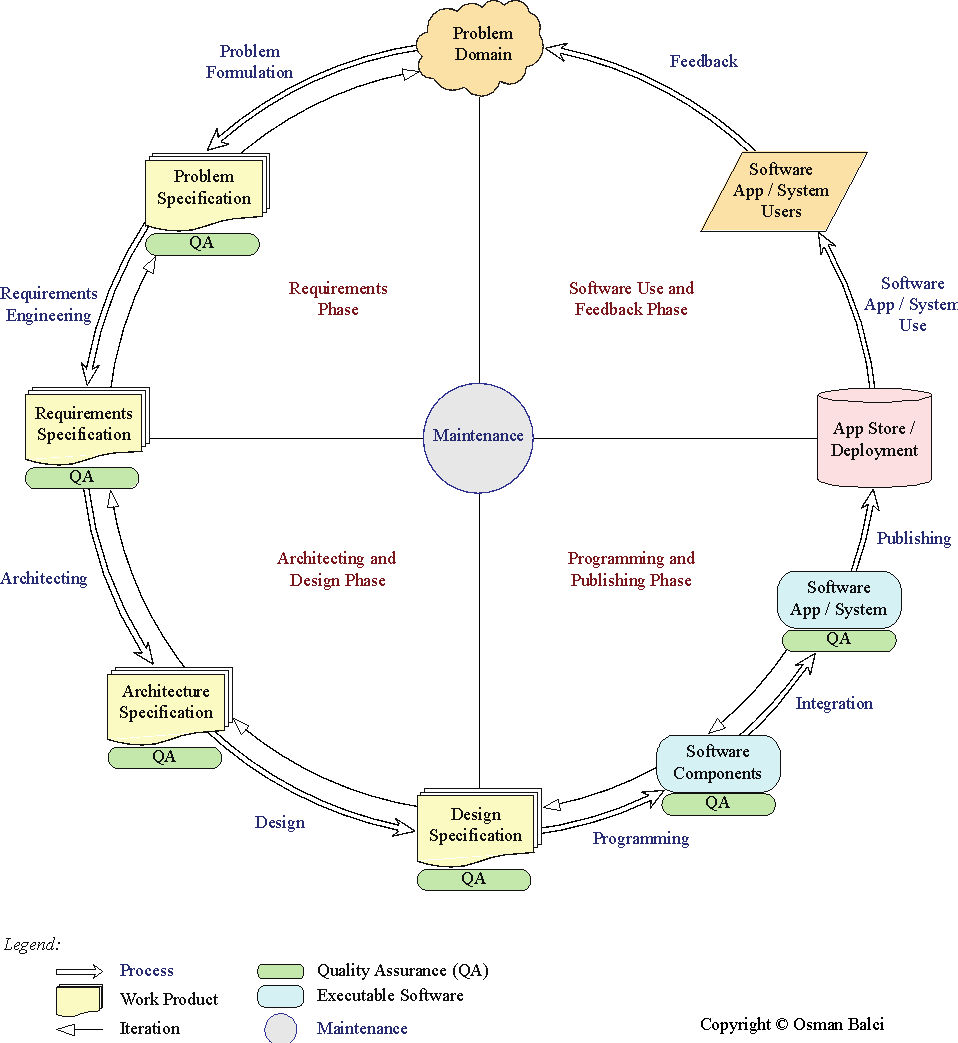
Right click the TOC above and select **Update Field**. Select **Update entire table** in the dialog box.

You must update the TOC to reflect changes before submitting the final version.

This page must be numbered as iv.

# SOFTWARE LIFE CYCLE

A good software engineer develops software by following the software life cycle shown below.



A programmer (hacker or ad-hoc developer) develops software by looking at the problem and directly coding in an IDE. This approach is known as the *Build-and-Fix Approach*, which must never be used!

This page must be numbered as 1 and subsequent pages must be numbered accordingly

# OUR OBJECTIVE

The objective of our team project is to **demonstrate how capable the team members are** in engineering a Java EE cloud software application to solve a complex problem. The Java EE cloud software application is created for the purpose of showing how *learned* and *new* complex functionalities and cloud features the team members are capable of developing.

# PROBLEM SPECIFICATION

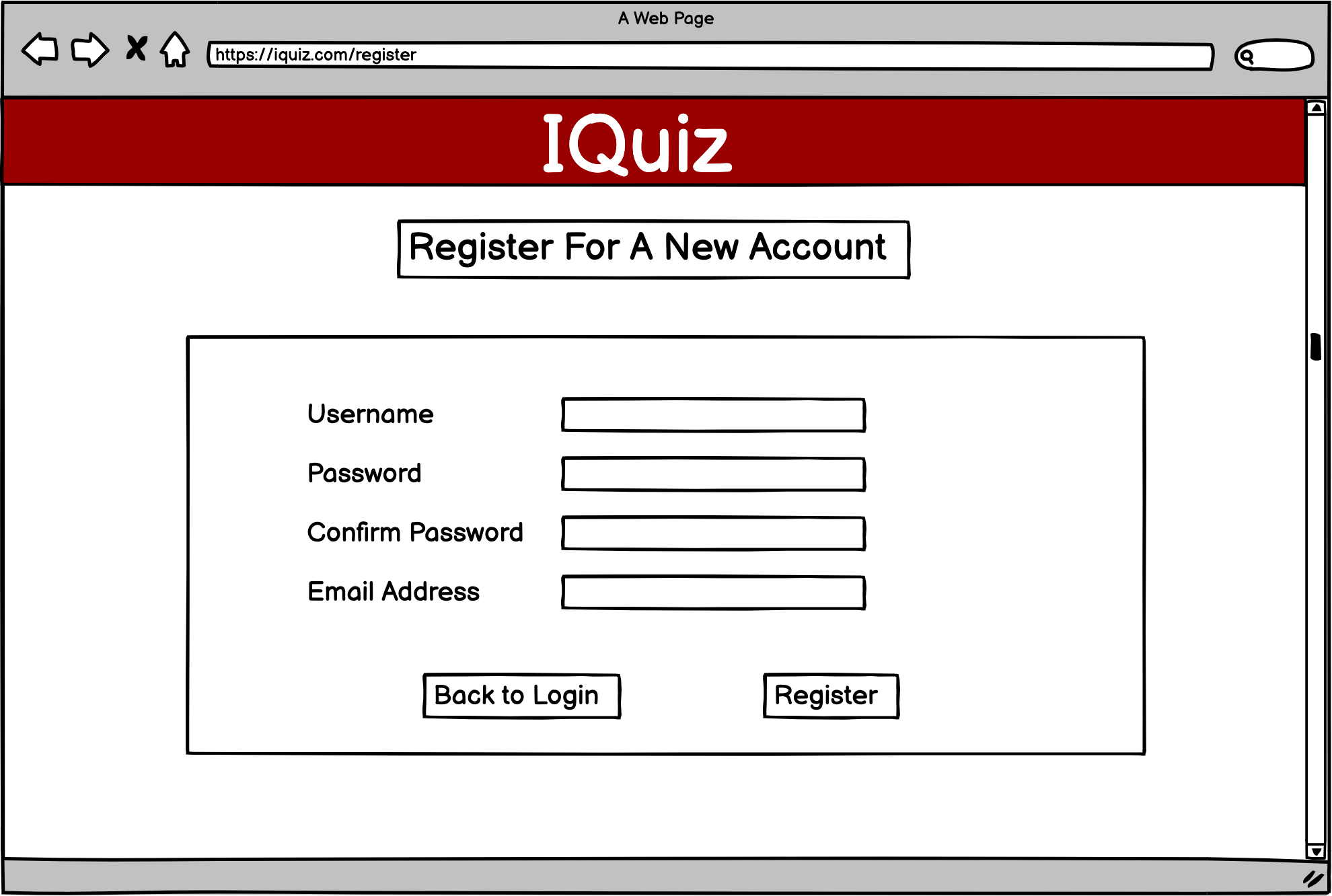
## What is the problem?

Due to the pandemic, more and more universities are deciding to move their courses online. This raises a new problem for all instructors and the departments about how to count student's grades. As more and more online quizzes are used and become a priority way to form the majority part of the students’ grades. Finding an easiest way to create a simple quiz becomes more and more certain. Instead of creating a quiz on some widely known platform like canvas which requires to fill out a large number of useless information for some popup quiz, some instructors are seeking for a simpler way to create quizzes for some lighting used quizzes like popup quizzes. We think this is a very serious problem, especially for the current online education. In order to provide an effective solution, we will create an online iquiz website which will satisfy efficient publishing quizzes and collect statistical results.

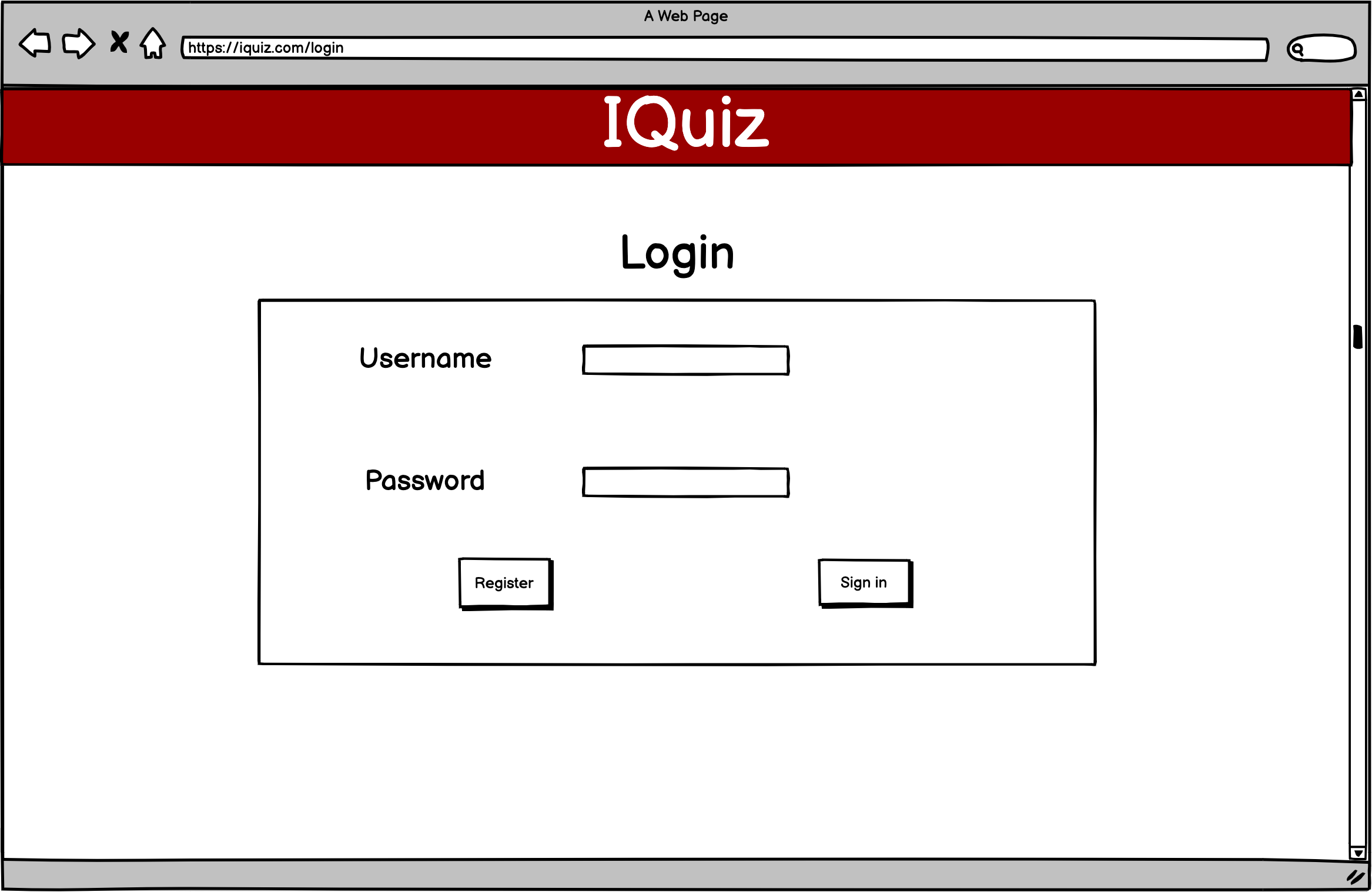
## Why is the problem important to solve?

As more and more students are taking classes online. Instructors are facing the problem of providing more and more quizzes. Some of the quizzes are counted to students' final grades. When instructors assign these kinds of countable quizzes, they do want to have the option to provide some complicated quiz rules like Anonymous Grading, Submission Attempts, and Assignment Group. Some quiz websites did a great job on setting up these rules like canvas. But for some uncounted quizzes like popup quizzes. The instructors may prefer to use a more efficient way to set up a quiz instead of canvas. The quiz website we provide will only require a simple authorization for the instructor side and all students can join the classroom easily by a link or an enter code. The instructor may only need a few minutes to create a simple popup quiz and publish it to the entire classroom. In the meantime, the quiz result will be shown to students in time and the results will be visualized. This project will be a useful tool for some instructors or departments to save time on creating quizzes.

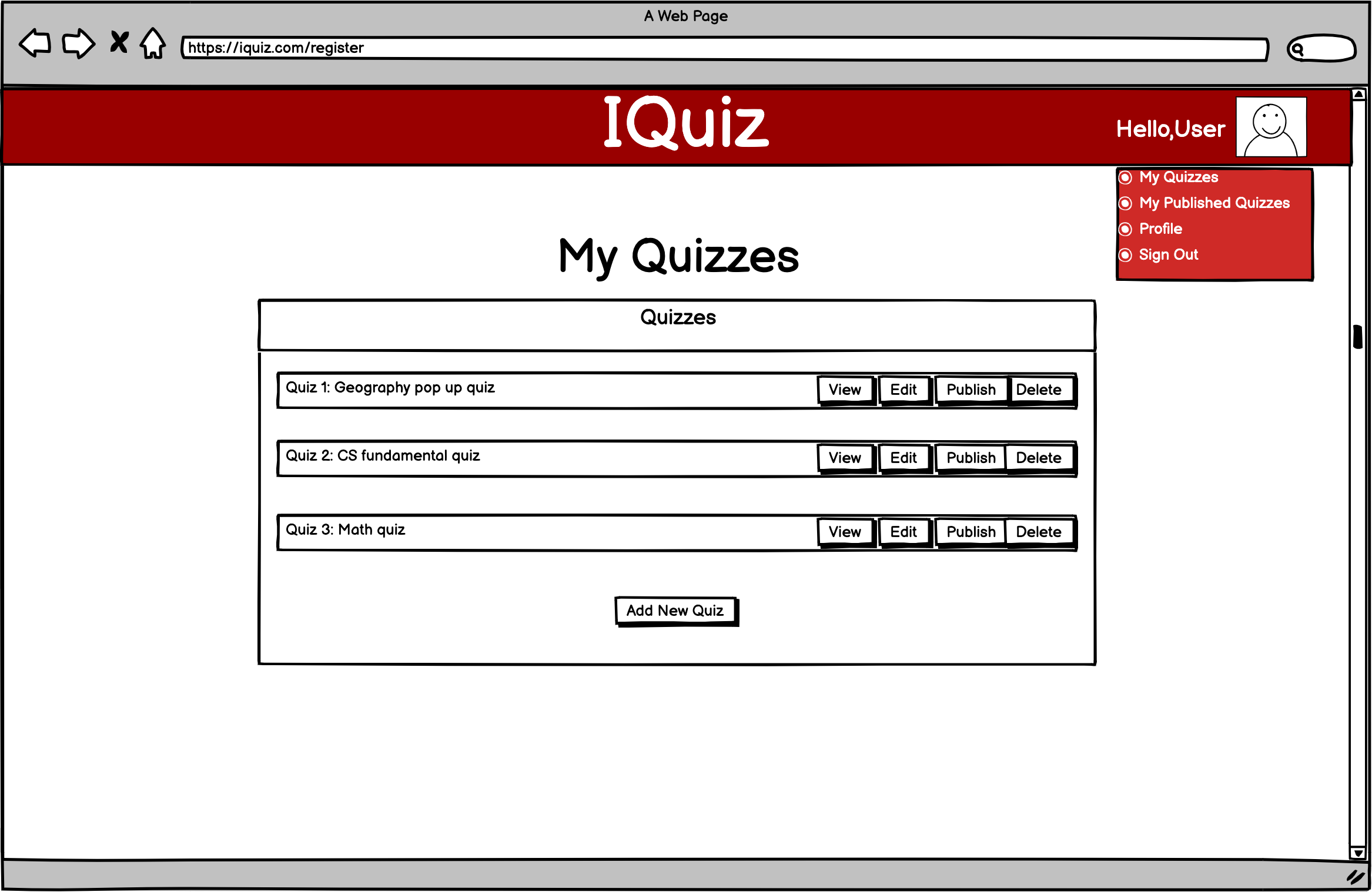
## Expected Functionality Description



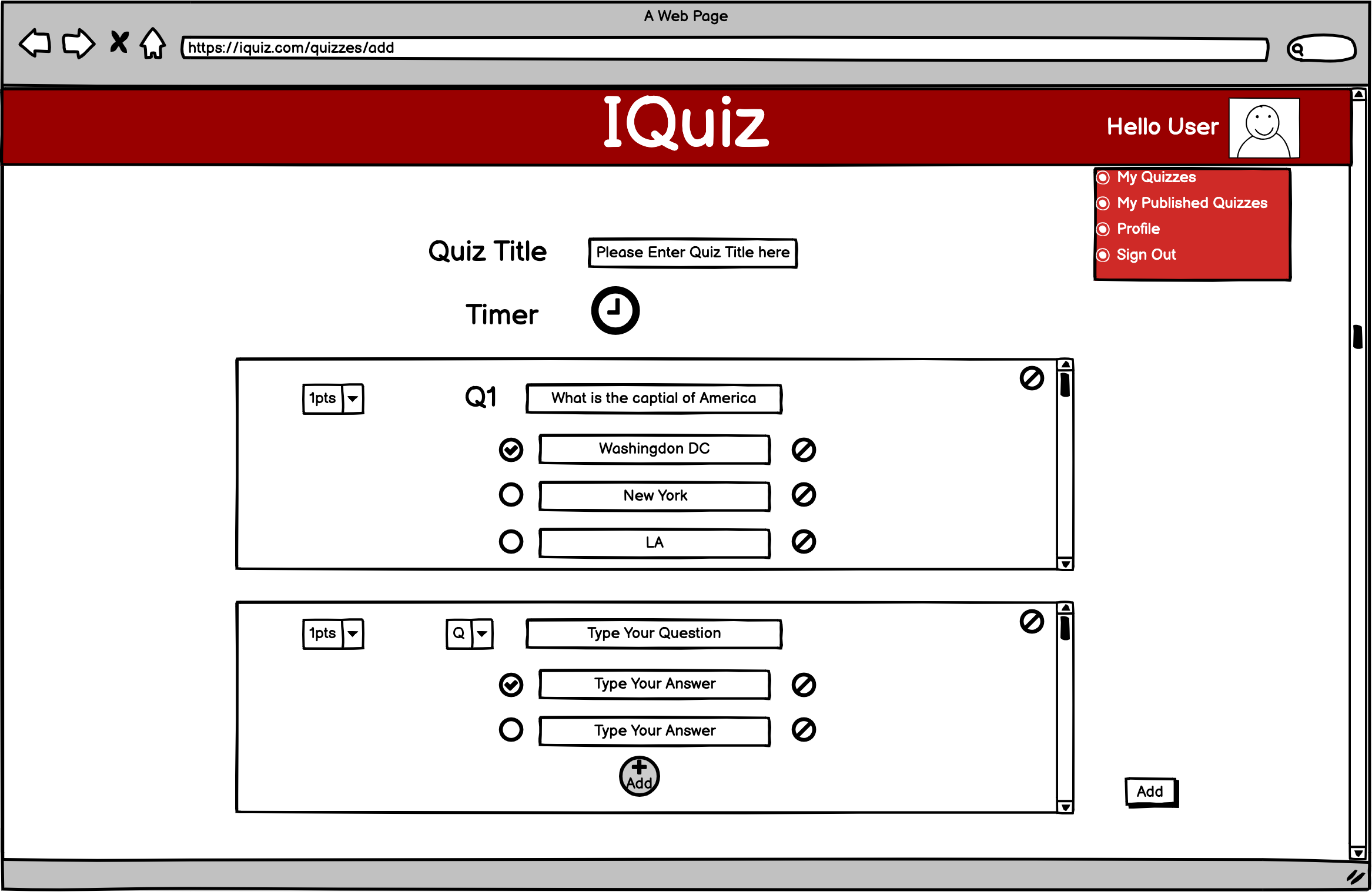
Users can register for an account to access website contents. All user information will be stored in database by using Mysql.



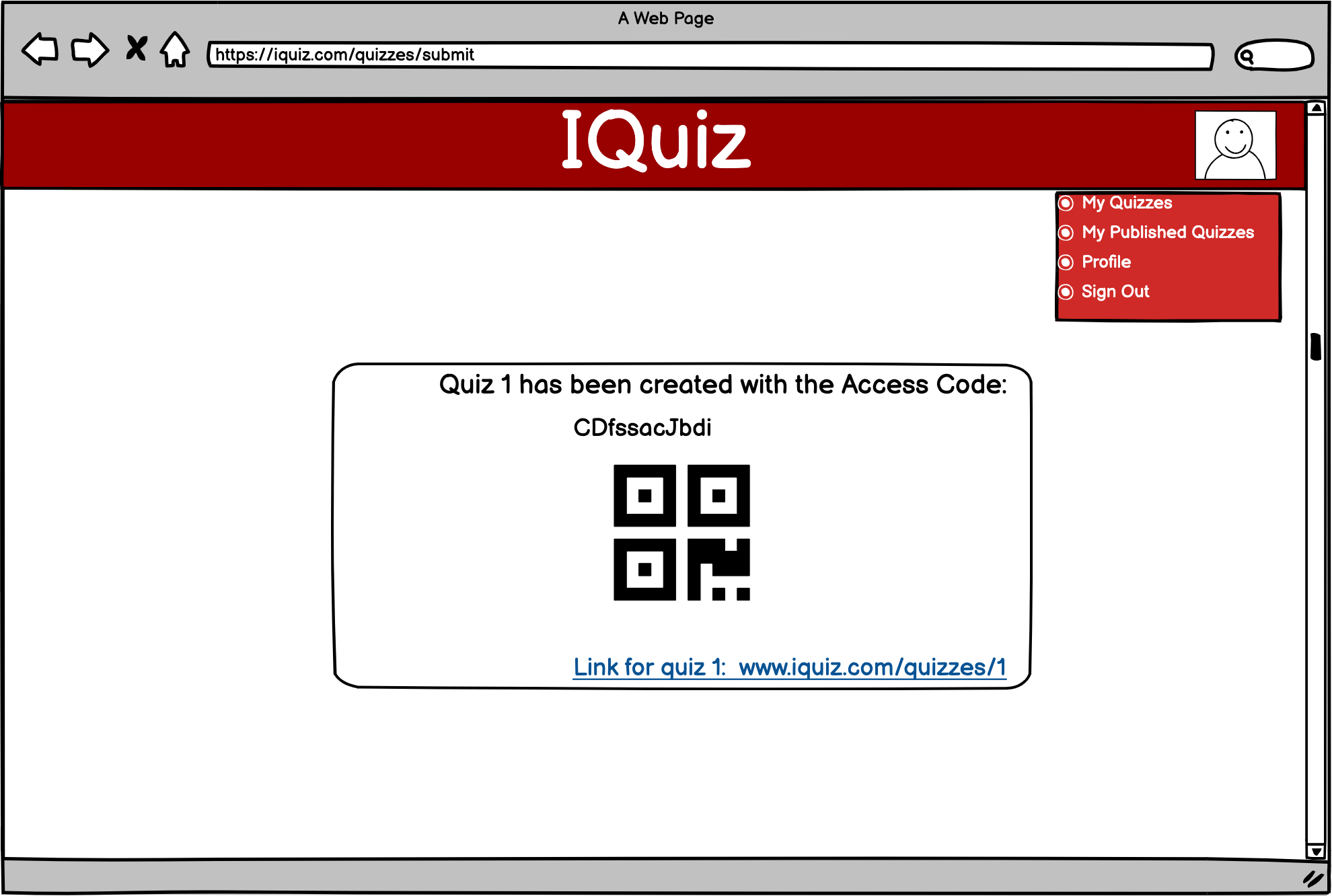
Users need to login to access website contents.



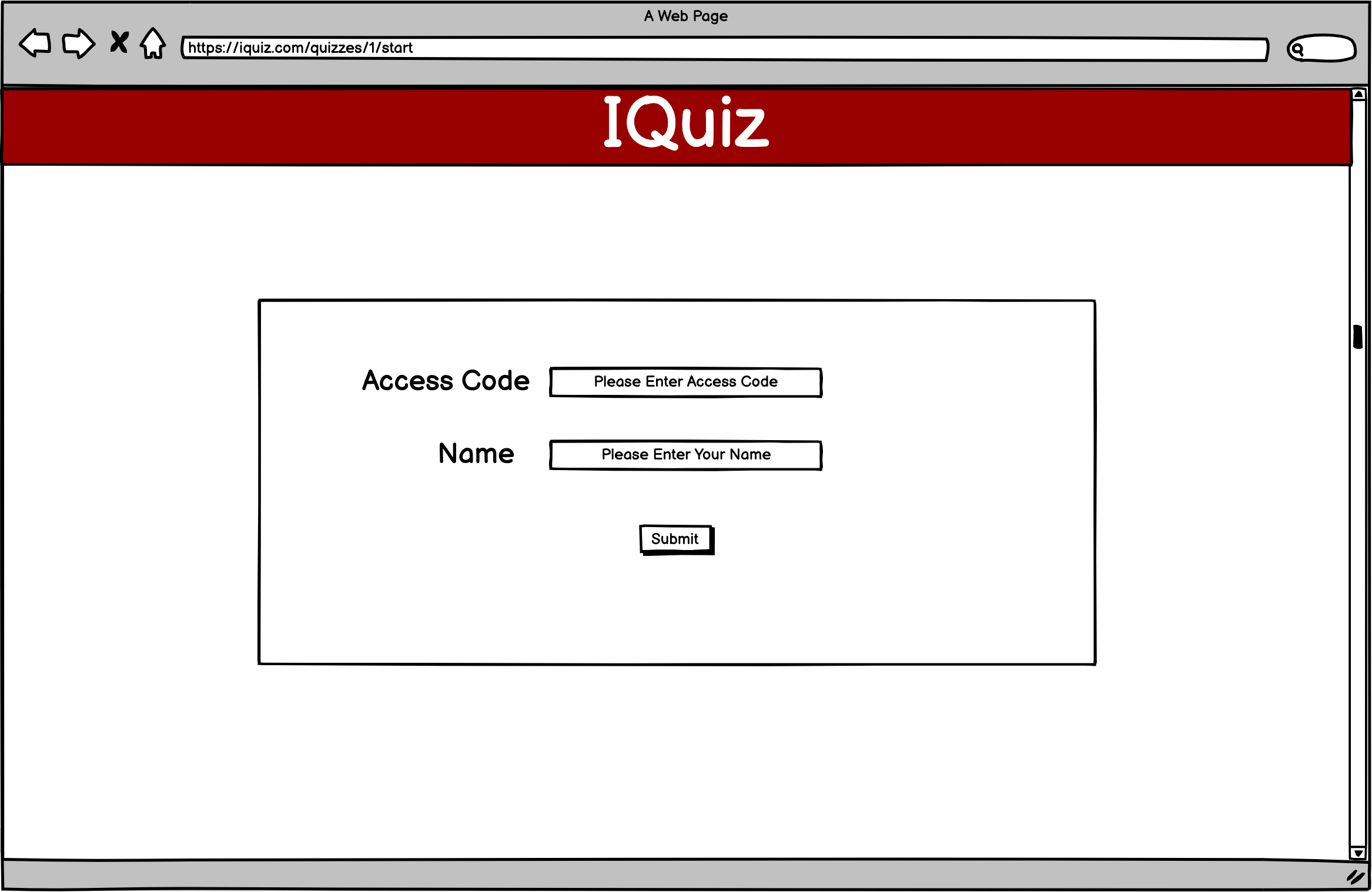
After the user logs in, he/she will be navigated to the home tab. It will contain all the quizzes the user has created. The user can choose to view, edit, publish, or delete the quiz. The user can also create a new quiz.



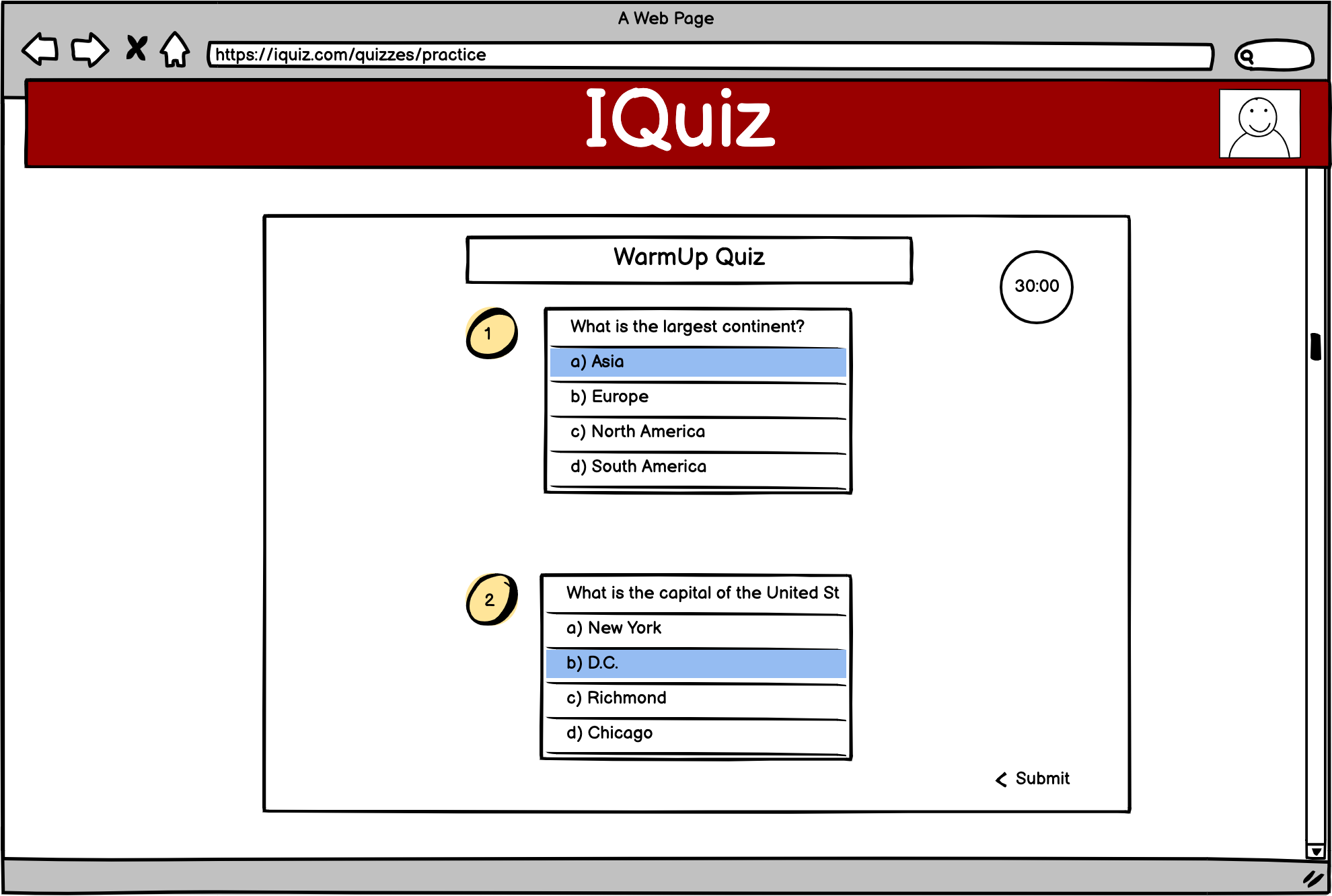
After the user clicks on “Add New Quiz”, he/she will be navigated to the add quiz tab. The user can add questions, remove questions, add question answer choices, remove question answer choices, set points worth, set correct answers, set the allowed time for the quiz. The order of the answers will be exactly the same order as the user entered.



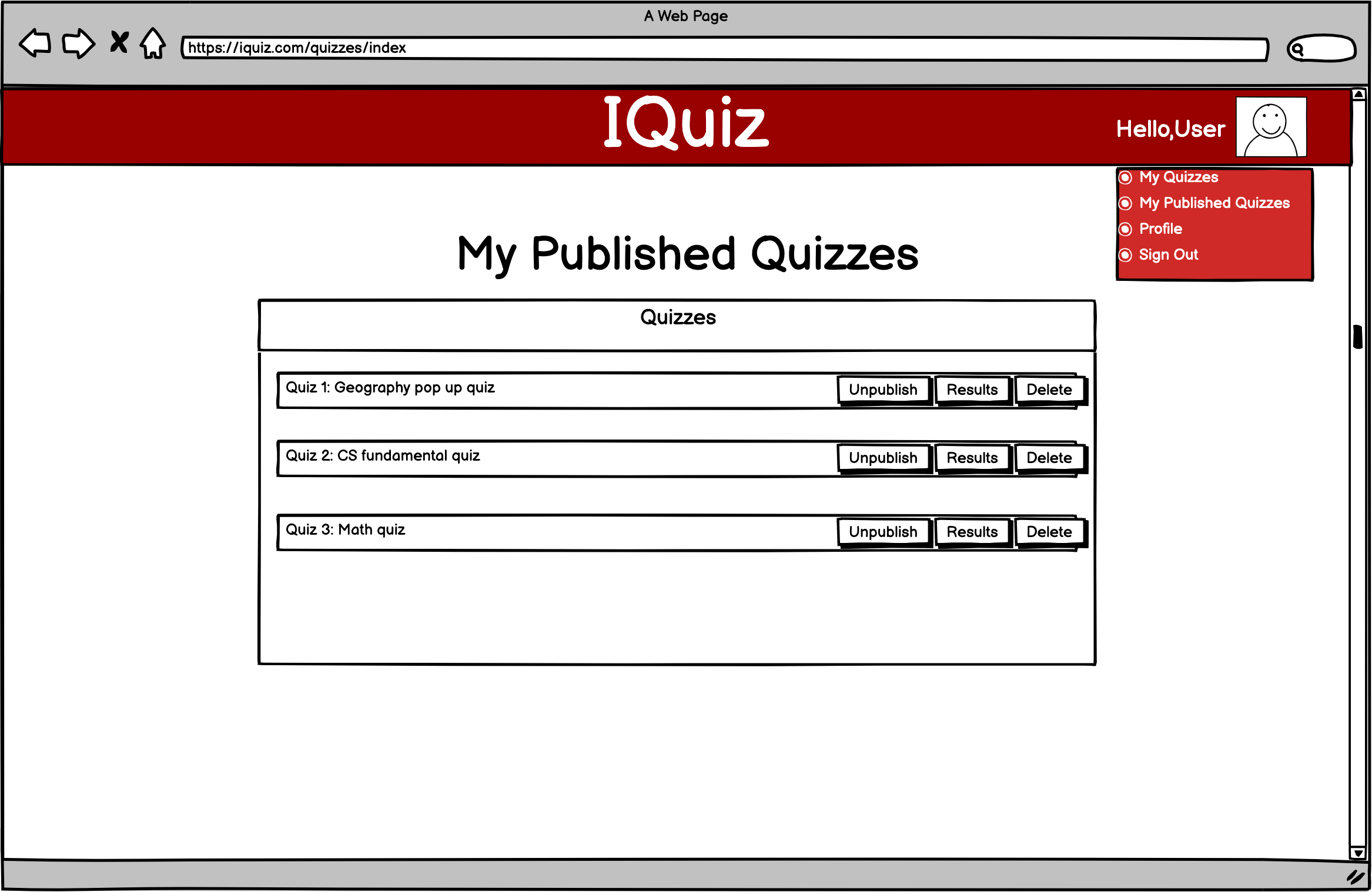
After a new quiz has been created, a pop-up window will appear with access code for the quiz and the link. Another option to access the quiz is using QR code from the phone.



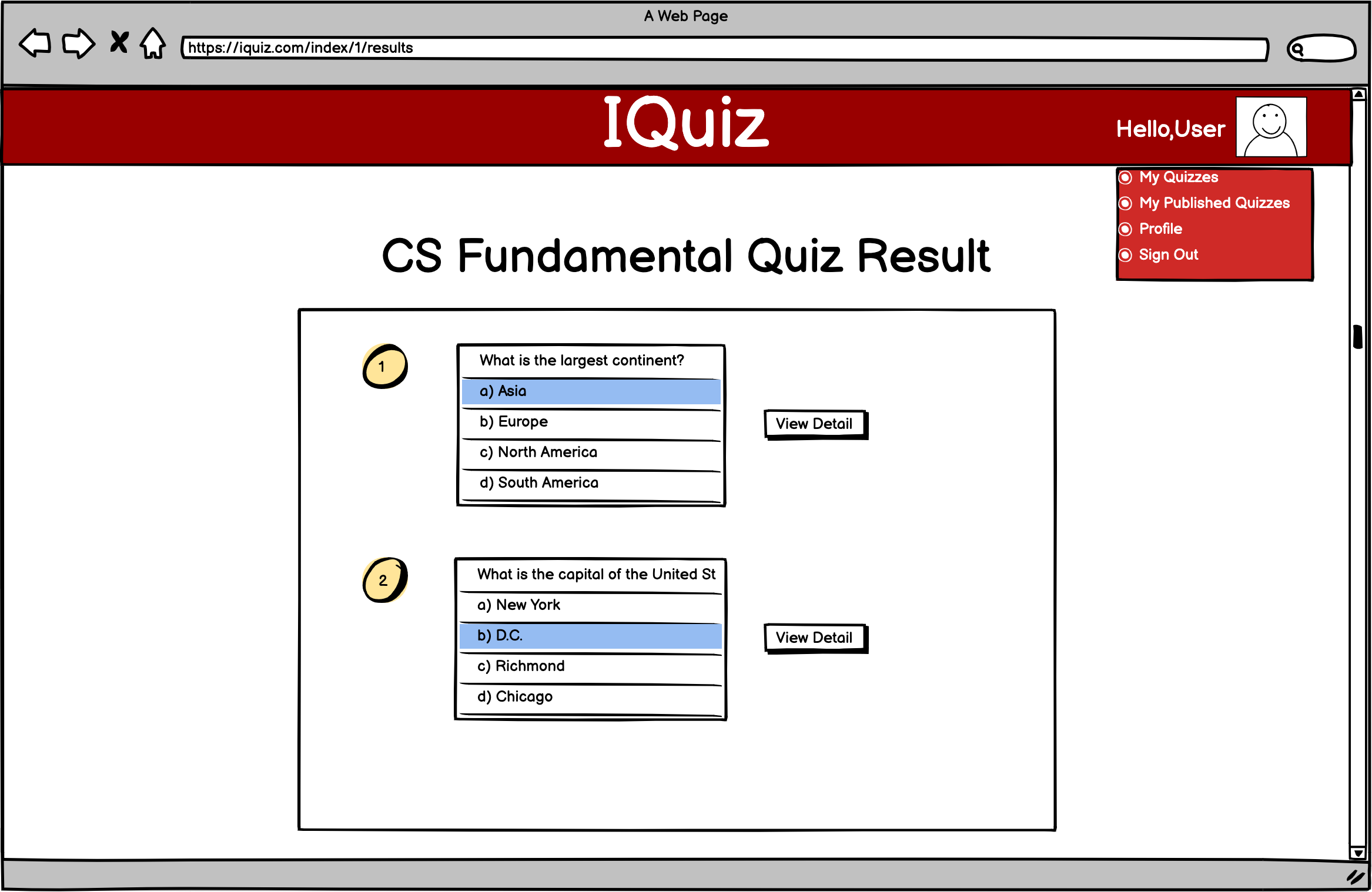
After clicking on the link, it will ask the student to enter the access code and name. If the student scan the QR code, he/she will be redirected to the website without access code requirements.



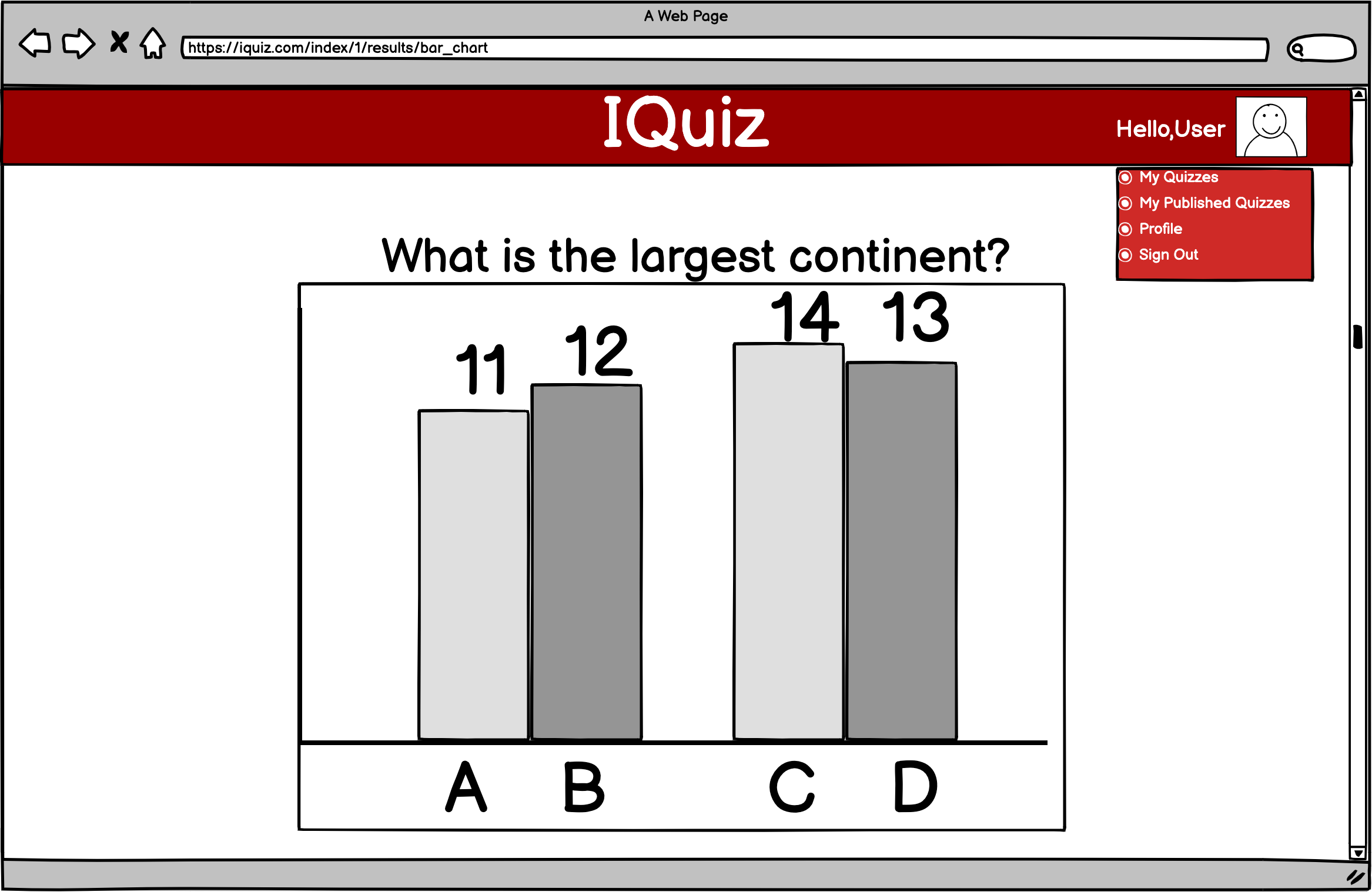
This is the page where the student takes the quiz. Selected answer will be marked in a different color. Click on the submit button to submit. After the student takes the quiz, the student will be navigated to a page saying your results have been submitted. If a timer is setted up and the student does not submit the quiz on time, the quiz will be submitted automatically.



After the student submits the quiz, the instructor can go to the published quizzes tab to see the results. He/she can also unpublished the quiz or delete the quiz.



Here is the page after selecting results. It will show each of the questions and the instructor can view the details of the question.



After selecting the view detail button, a bar chart will appear to show the student results of the question.

## Learned and new cloud software features to be implemented

|  |
| --- |
| **1. Learned Cloud Software Features Requirement:** Each team member is required to implement at least 3 learned features. A *learned feature* is defined as the one that is covered in the course tutorials and assignments. Click the hyperlink below to see a list of cloud software features learned in the course.  [Cloud Software Features Learned](https://manta.cs.vt.edu/cs3754/StudentsOnly/Project/LearnedFeatures.html)  **2. New Cloud Software Features Requirement:** Each team member is required to implement at least 1 new feature. A *new feature* is defined as the one that is **not** covered in the course tutorials and assignments. Click the hyperlink below to see a list of potential new cloud software features to implement.  [New Cloud Software Features](https://manta.cs.vt.edu/cs3754/StudentsOnly/Project/NewFeatures.html)  **3. Complexity Requirement:** A *learned* or *new* cloud software feature implemented in your team project app shall be judged with respect to its complexity and programming difficulty in determining your project grade.  **4. Time Requirement:** The following is the **minimum** time required to spend for the project.   * 6 hours per week x 4 weeks = 24 hours per team member * 24 hours per team member x 3 team members = 72 hours   The submitted project shall be judged based on the **minimum** 72 hours required in determining your project grade. |

Our proposed app shall implement the following *learned* and *new* cloud software features.

**Table 1. List of Learned Cloud Software Features to be Implemented**

|  |  |
| --- | --- |
| *No.* | *Description of the Learned Cloud Software Feature to be Implemented* |
| 1 | Store users and quizzes using MySQL database |
| 2 | Use storage in cloud for user icon uploads |
| 3 | Authenticate the user with username and password |
| 4 | Register an account with input validation |
| 5 | Create bar charts to show quiz results |
| 6 | Send quiz result using email |
| 7 | Capture User’s photo using the User’s computer’s camera |
| 8 | Use PrimeFaces Data table to show the quiz list |
| 9 | Quiz clone |

**Table 2. List of New Cloud Software Features to be Implemented**

|  |  |
| --- | --- |
| *No.* | *Description of the New Cloud Software Feature to be Implemented* |
| 1 | Use QRcode to access the quiz faster |
| 2 | Front-end Quiz countdown timer |
| 3 | Auto-Submission |

* At least 3 learned cloud software features required per team member.
* At least 1 new cloud software feature required per team member.

# REQUIREMENTS SPECIFICATION

This section specifies the Functional and Non-Functional Requirements under which our Java EE cloud software application will be developed.

## Functional Requirements

1. TBD
2. TBD
3. TBD
4. ...

Minimum 18 correct Functional Requirements are required. (6 per team member)

## Non-Functional Requirements

Each team shall satisfy the following non-functional requirements:

1. The UI template footer shall contain the following statement:  
     
   [CS5704 Software Engineering](https://manta.cs.vt.edu/cs5704/) course semester project application developed by studentName1, studentName2, and studentName3.  
     
   Course title and each student name above shall be hyperlinked to show the corresponding homepage *in a new window / tab*. Use VT website if you do not have one for yourself.
2. The cloud software application name in NetBeans and when deployed on the server shall be a meaningful name without course number or team number. The app name shall reflect what the app does such as MeetingScheduler. Do **not** name it as “group”, “Team4App”, or “CS5704Team4Project”.

Each team’s non-functional requirements:

1. TBD
2. TBD
3. TBD
4. ...

Minimum 9 correct Non-Functional Requirements are required. (3 per team member)

# ARCHITECTURE SPECIFICATION

TBD

Provide DoDAF diagrams of the Client-Server + Service-Oriented architecture for your Java EE-based software application.

# DESIGN SPECIFICATION

TBD

Provide **UML Class Diagrams** of your Java EE-based object-oriented software design in this section.

Use a software tool listed under Software Tools for Creating UML Diagrams at <https://manta.cs.vt.edu/cs5704/StudentsOnly/Resources/resources.html> or select your own tool in creating the UML diagrams.

If your app uses a file storage directory, name it as **TeamN-FileStorage**, where N is your team number.

**No credit will be given for a UML diagram that is not readable!**

# DELIVERED SOFTWARE FUNCTIONALITY

TBD

Describe all of the functionalities (features) of your deployed Java EE cloud software application by using screenshots of user interfaces. For each UI screenshot, describe how the software functions.

Start with the URL for accessing your software.

Then, screenshot by screenshot, describe how your software is used.

All of the software functionalities (features) must be described in sufficient detail so that your project can be properly graded.

# STEP-BY-STEP INSTRUCTIONS FOR DEVELOPING NEW FEATURES

TBD

<<

Putting together a functionality in an ad hoc manner by using resources on the web does not constitute learning. You are required to demonstrate that you have really learned by creating step-by-step instructions about how to develop the new Cloud Software features listed below, which you have implemented.

Provide a list of the **new** Cloud Software features you have implemented in your project.   
A feature is a **new** feature if it is not covered in the course tutorials and assignments.

>>

## New Feature 1 Title

TBD

* Describe the **new** Cloud Software feature you have implemented in one or more paragraphs.
* Provide **step-by-step instructions with screenshots** for how to develop the **new** Cloud Software feature, similar to the tutorials presented in class.
* Make sure that your Java and XHTML (JSF) code is very well documented.

## New Feature 2 Title

TBD

* Describe the **new** Cloud Software feature you have implemented in one or more paragraphs.
* Provide **step-by-step instructions with screenshots** for how to develop the **new** Cloud Software feature, similar to the tutorials presented in class.
* Make sure that your Java and XHTML (JSF) code is very well documented.

## New Feature 3 Title

TBD

* Describe the **new** Cloud Software feature you have implemented in one or more paragraphs.
* Provide **step-by-step instructions with screenshots** for how to develop the **new** Cloud Software feature, similar to the tutorials presented in class.
* Make sure that your Java and XHTML (JSF) code is very well documented.

:

:

Continue with the other new features, one subsection for each...

:

:

## New Feature N Title

TBD

* Describe the **new** Cloud Software feature you have implemented in one or more paragraphs.
* Provide **step-by-step instructions with screenshots** for how to develop the **new** Cloud Software feature, similar to the tutorials presented in class.
* Make sure that your Java and XHTML (JSF) code is very well documented.

# CONCLUSIONS

TBD

# SUBMISSION INSTRUCTIONS

1. If your app uses a file storage directory, name it as TeamN-FileStorage, where N is your team number.
2. Deploy your cloud software application to your AWS EC2 virtual private server computer.
3. Using the deployed software, each team member is required to create an account and generate *meaningful content* in the database for evaluation and testing. You will be penalized for not having sufficient content. In the case of two-factor authentication, your software is required to provide a Bypass Two-Factor Authentication option for evaluation and testing. List the usernames and passwords under this option in the *User Accounts* section on page ii.
4. Export (dump) your populated MySQL database content into a SQL script file using # mysqldump -u root -p NameOfDB > NameOfDBwithDBcontent.sql
   1. Move all DROP TABLE IF EXISTS 'TableName' statements in the exported SQL script file to the top of the file and *list them in a logical order* based on foreign key and delete cascading dependencies.
   2. *Reorder all of the table creations* with CREATE TABLE 'TableName' in a logical order.
   3. Correct problems with apostrophe as being the same as the single quote. MySQL exports character strings enclosed within single quotes. If there is a string like Dave’s it conflicts. So, if you have 'These are Dave's files.' change it to "These are Dave's files." to prevent single quote conflict.
   4. Test the corrected exported SQL script file on a computer and make sure that it can be imported to populate the database under the NetBeans IDE.
5. Create a ZIP file containing the following:
   1. This file: *Team N Semester Project Report.docx* (Replace N with your team number. Do not change this filename and do not remove the spaces in the filename!)
   2. Your IntelliJ IDEA Ultimate project folder containing your entire application.
   3. Database SQL file **original version**.
   4. Database SQL file **exported from the database** on the server in Step 5 above.
   5. TeamN-FileStorage directory **copied from the server** computer containing the files stored during use by the team members in Step 4.
6. Upload the ZIP file to your account in [Google Drive](https://www.google.com/drive/), right click the uploaded file and select Share from the pop-up menu, enter [balci@vt.edu](mailto:balci@vt.edu), add a note as “CS5704 Team N Semester Project Submission”, and click Send.

# PERCENTAGES OF CONTRIBUTION

We hereby certify that the list of contributions and the corresponding percentages of contribution specified below truly reflect the actual contributions of the team members.

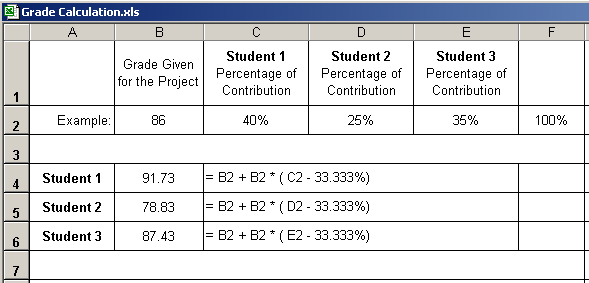
(Write your name as your signature)

|  |  |  |
| --- | --- | --- |
| *Student Name & List of Contributions* | *% Contributed* | *Signature* |
| Student Name 1’s Contributions:   1. tbd 2. tbd 3. tbd 4. ... | a% |  |
| Student Name 2’s Contributions:   1. tbd 2. tbd 3. tbd 4. ... | b% |  |
| Student Name 3’s Contributions:   1. tbd 2. tbd 3. tbd 4. ... | c% |  |
| Sum of Percentages: | 100% |  |

The **Virginia Tech Honor Code** is fully in effect for the above declaration. **The percentage of contribution must be justified by the list of contributions specified in detail**.

If you disagree with your team members, you and the other team members must separately email Dr. Balci and indicate what each member thinks the fair percentages of contribution are for all team members.

# CALCULATION OF GRADES BASED ON PERCENTAGES OF CONTRIBUTION



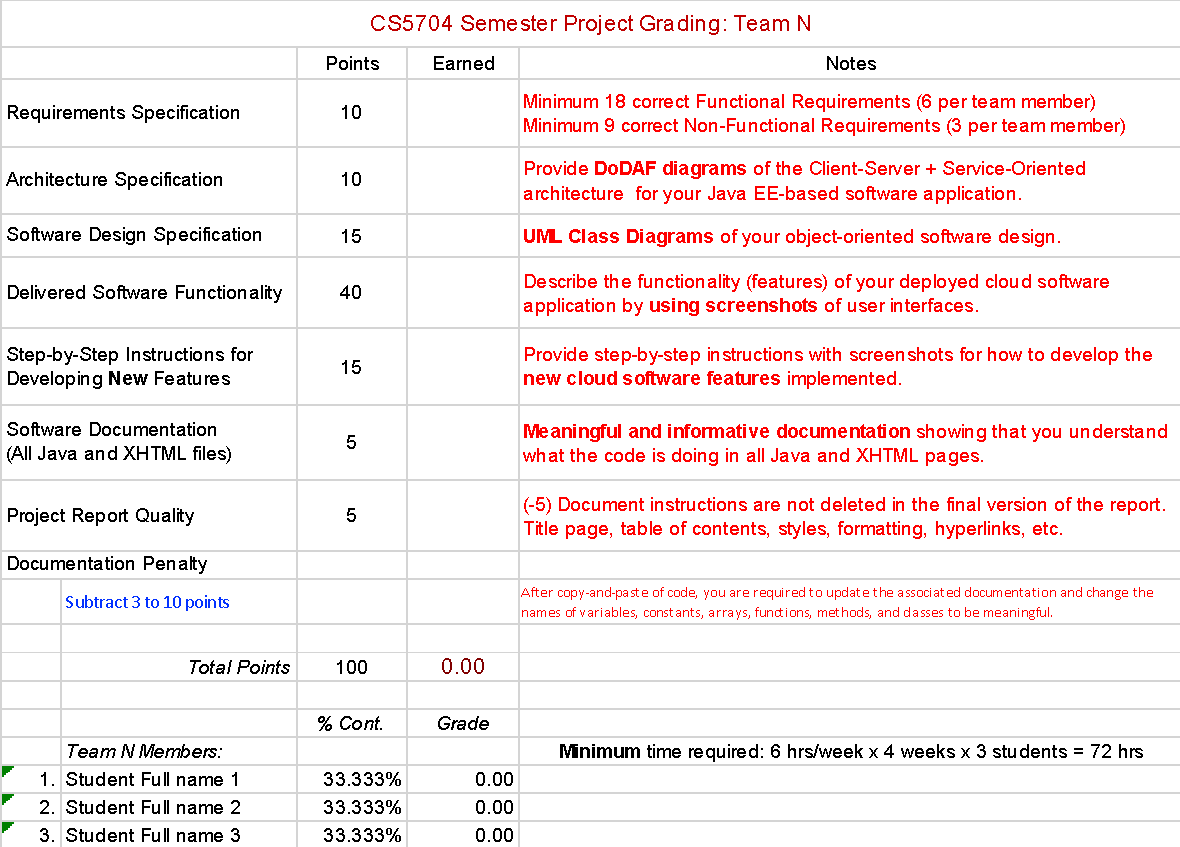
**Additional percentage of contribution cannot exceed 10%.**

**If a situation arises where a student is doing more than 10% extra work, Dr. Balci must be informed immediately.**

# TEAM PROJECT REQUIREMENTS

1. Each team member is expected to contribute equally.
2. You shall submit percentages of contribution together with a list of each student’s individual contributions with signatures of all team members. In case of disagreement, you shall submit it separately with your rationale. (Write your name as representing your signature.)
3. Grades shall be determined based on the percentages of contribution.
4. If you contribute more than your equal share, then your grade shall be increased based on the extra percentage of contribution, which cannot be more than 10%.
5. If you contribute less than your equal share, then your grade shall be decreased accordingly.
6. The extra percentage of contribution shall not be more than 10%. Dr. Balci shall be notified immediately if a situation arises where a student needs to contribute more than 10% extra.
7. A team member who does not cooperate with other team members for conducting the project with equal contribution shall be penalized. Doing more work than agreed upon by the team and claiming extra contribution shall not be acceptable. Cooperation is essential!

# GRADING SHEET



**REFERENCES**

TBD

Balci, O. (2021), “CS5704 Software Engineering Course Website,” <https://manta.cs.vt.edu/cs5704/>

MySQL (2021), “MySQL Open Source Relational Database Management System,” <https://www.mysql.com/>

<< Reference other sources used in your work. Include other references used in alphabetical order styled under the style **References** >>

**APPENDIX A: MEETING MINUTES**

**Meeting Number 1**

|  |  |
| --- | --- |
| *Date & Duration:* |  |
| *Location:* |  |
| *Members Present:* |  |
| *Members Absent:* |  |
| *Discussions:* |  |
| *Decisions Made:* |  |
| *Work Assignments:* |  |
| *Minutes Prepared By:* |  |

**Meeting Number 2**

|  |  |
| --- | --- |
| *Date & Duration:* |  |
| *Location:* |  |
| *Members Present:* |  |
| *Members Absent:* |  |
| *Discussions:* |  |
| *Decisions Made:* |  |
| *Work Assignments:* |  |
| *Minutes Prepared By:* |  |

**Meeting Number 3**

|  |  |
| --- | --- |
| *Date & Duration:* |  |
| *Location:* |  |
| *Members Present:* |  |
| *Members Absent:* |  |
| *Discussions:* |  |
| *Decisions Made:* |  |
| *Work Assignments:* |  |
| *Minutes Prepared By:* |  |

:

:

:

(List for all meetings held)

:

:

:

**Meeting Number N**

|  |  |
| --- | --- |
| *Date & Duration:* |  |
| *Location:* |  |
| *Members Present:* |  |
| *Members Absent:* |  |
| *Discussions:* |  |
| *Decisions Made:* |  |
| *Work Assignments:* |  |
| *Minutes Prepared By:* |  |