A Web-based Q&A-type of forum for scientific Math-related content.

Ioanna G. Kapodistria

Human-Computer Interaction Master, University of Patras, iokapodistria@gmail.com

1088746

This project is dedicated towards the designing, implementing, and evaluating of a web-based knowledge management system in a form of a Q&A forum for scientific Math-related content, called MathOverflow. The purpose of the application is educational, therefore there is not a competitive aspect in it. It aims to facilitate not only the collection and organization of answers an questions in order to find a solution to mathematical issues but also the development of a community in which various roles will be developed and the sharing of knowledge will be promoted worldwide. The implementation of the project was based on Express.Js which is a web framework for Node.Js.

Additional Keywords and Phrases: Q&A forum, Web-based app, Sequelize, Express.Js

1 INTRODUCTION

Q&A systems seem to have two main advantages over traditional knowledge-based systems (e.g., knowledge bases). Initially, users are more responsive and willing to answer questions than to enter information into the system themselves. It is also a common phenomenon that one does not understand that has acquired knowledge (and in fact, often remarkable) until is asked about the object.

The general purpose of this system is to collect, record, organize and share users' knowledge. The main functions of the system are the ability of the user to ask a question that is addressed to other users, respectively the ability to answer another user's question and the ability to search for questions and answers.

2 METHODOLOGY

Below is the methodology followed for the implementation of the system.

2.1 Inspiration

The inspiration came from the observation of some similar question-and-answer (Q&A) websites such as Stackoverflow [2] and Quora [3]. Both applications follow a user-centered approach on which this application is also based. Furthermore, the structure of questions and answers is based on the corresponding structure of the above applications.

2.2 Basic Analysis and Design of Users and Functionalities

After the first inspired ideas for the form of the application we proceeded to the conceptual design, we define what will be the purpose of the system, who will it serve, what will be the roles and the corresponding functionalities. Below we present the Users of the system through a use case diagram and the functionalities through an ERD Model.

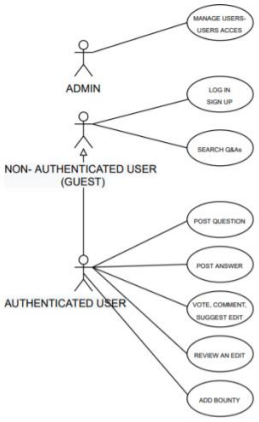

Figure 1: Use case Diagram

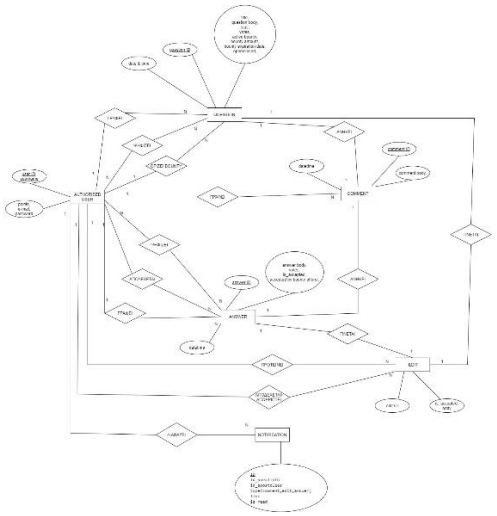

Figure 2: ERD Diagram

2.3 Implementation

The implementation of the system was based on Express.Js which is a web framework for Node.Js. It was completed according to the following steps:

1. Draft design and implementation of the user interface, through HTML, CSS, bootstrap and addition of client side javascript.
2. Connecting the PostgreSQL database to Sequelize through a configuration file.
3. Creation of Sequelize Models (User, Question, Answer)
4. Implementation of server-side functionalities (User's sign up, ask a question, post an answer, show all users, show user's profile, login, logout).
5. Improvement of initials templates.

3 EVALUATION

Although the application is still at an early stage, the main evaluation criterion was to make the system easy to use and functional. That is, the user can find what he is looking for easily and without many steps. For this reason, after the first implementation of the main actions, their functionality was tested by us and improved in a iterative manner.

4 DATA

In order to complete the project and to make the evaluation properly, we needed to collect data, specifically questions and answers on scientific issues in the field of Mathematics. The data was collected from the platform "Mathematics Stack Exchange" [1], which is a more complex question-and-answer forum for Mathematic's topics.

5 MAIN ACTIONS

As for the main actions needed to complete the project, these are mainly related to the back-end part. Not all the functionalities that were originally recorded on the ERD model were implemented.

Specifically, regarding the "User" object, the following functions were implemented:

1. User Sign up
2. Show User by Id. This functionality leads to the profile of each user and retrieves selected user's information such as questions and answers which have been made by him.
3. Show all Users which retrieves all users of the system that have been signed up.
4. User log in/out

Respectively for the object "question" the following functions were implemented:

1. Create/ask a question
2. Show all questions. This functionality gets all the questions that have been made.
5. Show question by Id. This functionality leads to the page of each question and retrieves selected question's answers.

Finally, regarding the object "questions", the functions that were implemented are the following:

1. Create/post an answer

6 TIME SCHEDULE

The process of analysis, design and implementation of the system is carried out throughout the semester but especially in the last weeks. Specifically, for the conceptual design it took 2 weeks, while for the implementation of the system it took 6 weeks. The final report was completed within 2 days.

7 FUTURE WORK

As a future work we intend to improve / add some features which may in the specific version of the project be considered as problems. Firstly, the design needs to be further improved in order to be more responsive. Secondly, the addition of an editor with the appropriate mathematical symbols both for the proper recording of questions and answers is also necessary. Finally, the implementation of functions such as deleting questions / answers, voting question / answer and the integration of a search engine are some important aspects that need to be further considered.

REFERENCES

[1] Mathematics Stack Exchange. "Mathematics Stack Exchange." Accessed June 17, 2021. https://math.stackexchange.com/.

[2] Stack Overflow. "Stack Overflow - Where Developers Learn, Share, & Build Careers." Accessed June 17, 2021. https://stackoverflow.com/.

[3] "Quora." Accessed June 17, 2021. https://www.quora.com/.