A Generic Quiz Hosting Application Using an Efficient Content Management System

A THESIS

submitted by

VISHAL KARTIK KARANAM B120736CS

VANGAPALLY UJWAL B120905CS

VIDHEY PALURU B120419CS

in partial fulfilment for the award of the degree of

Bachelor of Technology
in
Computer Science and Engineering

under the guidance of

T A SUMESH

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
NATIONAL INSTITUTE OF TECHNOLOGY CALICUT
NIT CAMPUS P.O, CALICUT
KERALA, INDIA 673601

April 21, 2016

Abstract

The main motive behind taking up the project of developing a generic and scalable Quiz Hosting Application is to reduce effort, time and energy which is otherwise being wastefully spent in manually conducting and evaluating Quizzes .The project consists developing a standard and responsive platform where in quizzes can be hosted and instantaneously results would be displayed thereby eliminating the cons of manual correction.The added advantage of the project is that it can be put to cross functional uses across departments.

The project report deals with the aim of taking up this project, in-depth design details,salient features that the quiz application provides along with the implementation details.This document specifically consists of clear and concise details about project objectives, scope, application limitation,requirement specifications, team development methodologies, possible risk.The project has been completed with strict adherence to software engineering principles and methodologies and formal coding standards were followed.

Contents

List of Figures

Chapter 1

Introduction

Currently a lot of Tedious manual effort is being put for conducting Quizzes or Lab assignments for academic purposes or Mock Tests for Placement Activities. The current project is aimed at reducing manual effort, time and energy by developing a standard platform where Quizzes can be hosted and instantaneously results would be displayed eliminating the cons of manual correction. We need a more time saving and more accurate examination system as the number of applicants is increasing day by day and also the need for providing a comprehensible and a user friendly environment is more than ever.The Faculty or Quiz setters need not put time and effort by printing out paper sets for the quiz and also need not sit down to correct the examination papers by hand.

The Application is mainly built to serve as a medium for hosting Multiple choice questions (MCQ's).The Setter hosts a quiz with desired settings like the time of availability of quiz,total marks on offer,duration of quiz,number of questions etc.There can be positive and negative marks for each question and every quiz must be a timed one where in student has to complete the test within the allotted duration.The setter can use this application to test the subject knowledge of a student or quiz a user on a particular matter of interest.After the test is completed reports regarding the users performance is generated.

1.1 Problem Statement

The central idea is to develop a generic and scalable quiz hosting application with efficient CMS (Content Management System) for controlled publishing, organizing and maintaining data to ease the burden on faculties and students.

1.2 Literature Survey

The technologies necessary for seamless implementation of the project have been studied. They include Ruby,PHP,Python and frameworks like YII, Laravel, Ruby on Rails , Codeignitor etc. Various brain storming sessions have been held discussing the advantages and disadvantages of each of these technologies and the suitability and applicability of these technologies to the desired final product to be rolled out.Studies have also been done regarding the load handling capacities required.The final project is to be implemented with the help of the latest YII 2.0 framework using PHP. Various software development methods have been studied from Roger S Pressman [1]

and are duly followed in the implementation of the project.After understanding the framework and the features it has to offer , in depth focus has been given to learn code structuring and modular packaging as well as understanding the integration of necessary widgets.The official documentation *www.yiiframework.com*[3] is strictly adhered to.

1.3 Functionalities Offered

The quiz application has been built in a way such that it is light weight and generic but still scalable and offers a wide set of features.
The main salient features are :

1) Students can access previously conducted tests via virtual tests, which gives the students a feel of the actual quiz in case of practise or if they've missed the quiz.

2) A student ranking system : A ranking system is incorporated so that students can track each others performances and to foster a healthy competitive spirit.

3) Students can get their performance review of the test they have attempted so that they can retrospect.

4) Facility for the solutions and comment sections for conducted quizzes for an easy look back from the student point of view.

5) The Roles for the users and admins are allotted with due discretion using structured formulation of RBAC (Role Based Access Specifier).

6) The student can see the list of past,present and future quizzes and can attempt them appropriately.

7) The question set is neatly divided into pages using *pagination* so as to prevent the quiz from looking cluttered.

8) After submitting the quiz the student can also view the score obtained along with his/her number of right and wrong attempts.This gives a chance for scrutiny when he/she logs back to view the previously attempted quizzes.

9) Weightage for each question can also be specified.This has the added advantage when some questions are more important than others.

10) Timer for each quiz is also set so that students can keep track of their pace and also to ensure that each activity is time bound.

11) The questions are also presented in a jumbled order so as to prevent malpractices during the exam.Nonetheless each student gets the same set of questions and options but their timing of occurrence may differ.

12) There is an enrollment key for each quiz so that only the students/ audience aimed at will be attempting the quiz.This feature is added to avoid other non targeted users from attempting the quiz.

13) Graphical performance analysis.

14) A text editor for the setter for neat structuring during question formulation for a specific quiz.

15) Simple and efficient CMS(Content Management System) for simpler managing and content publication.

16) CRUD features for each quiz and each question in the setter side of the application .

17) Highly responsive User Interface .

18) Optimized querying to reduce load on database as far as possible.

19) Facility to use latex type mathematical symbols and operations.

The application will be scalable in such a way that it can be put to several uses across various cross functional platforms. It can be used for academic purposes to host lab/course quizzes or it can be also used by student chapters like CSEA,CSI,ISTE etc for conducting their quizzes.A SWOT analysis is as follows.

Strengths: Project leads to reduction in manual effort. Light and easy to use software,along with scalability provisions.

Weakness: Hardware requirements scale up when number of users increase.

Oppurtunities: Interaction with the aid of tutorial sections. can be used for mock placements and other CSEA/CSI/ISTE/mock GATE exams.

Threats: Security of the application can be a threat . Stability of the application with respect to power failure and crashes is a problem.

Chapter 2

Development Procedure

The Project is being constructed in due accordance and strict adherence to the Software Development lifecycle rules. The following stages are completed in order.

1) Concise Problem Statement : The first step in any development is the understanding of objectives and putting forward thoughts in a clear way. Thus a clear and Concise problem statement has been formulated and the issues being addressed have been thoroughly studied and grasped by the team. Descriptions and metrics in evaluating the preciseness have been addressed properly to prevent ideological lapses later on.Techniques and principles necessary for project design and implementation have been studied and understood.We Divided the project into phases and formulated a SWOT Matrix.

2) Formulating a SRS : The Software Requirement Specification Document has been prepared. The clientele has been approached and they have been quizzed about their expectations from the project and their requirements have been duly noted to be fulfilled accordingly. The document in detail describes about the scope, purpose , features of the project , the functional and non-functional requirements and use cases diagrams for the users of the application. The use case diagrams for all scenarios have been described in detail in the SRS. The complete in-depth details can be found in the SRS Document.

3) Design Phase :In this phase a careful step by step formulation of the application has been done.Object Oriented principles along with software engineering methods have been duly followed along with standard operating procedures.The design document consists of the design specific details as to how the project structuring is done. It includes elucidations of Class Diagrams, E-R diagrams , Sequence and Activity Diagrams taking in due note the SRS of the system. The complete and wholesome procedure is described in detail in the Design Document.

4) Project Implementation : The project implementation has been done with all necessary features included along with security aspects.The several stages during implementation is described in the implementation section.

5) Project Testing : The project testing has been done to verify that necessary features included along with security aspects are working in a proper manner.There have been several tests corresponding to looking out for any untowards bug which may happen during product release,any unwanted control of authority and load testing after hosting it on athena server.

Chapter 3

Requirement Specification

A brain storming session is good source of interactive session among students and between the teacher and students. Requirement gathering is done in order to improve the gatherers comprehension levels and learning motivation.

We have developed a Web-based online test system which can create quizzes competitively and collaboratively for the purpose of reducing the load required for a teacher/setter and promoting interactions among students and between the teacher and students.

3.1 Requirement Analysis

Requirements analysis in systems engineering and software engineering, encompasses those tasks that go into determining the needs or conditions to meet for a new or altered product, taking account of the possibly conflicting requirements of the various stakeholders, such as beneficiaries or users. Requirements analysis is critical to the success of a development project. Requirements must be documented, actionable, measurable, testable, related to identified business needs or opportunities, and defined to a level of detail sufficient for system design. Requirements can be architectural, structural, behavioral, functional, and non-functional. The development of project needs some requirement to make the project perform better and achieves the goal of project. In developing an Online Quiz application , the capabilities of computer and hardware plays a big impact on project quality. The project maker should determine the minimum requirements of hardware and also software to be used to develop a good and attractive project. There are two phases of requirement analysis as given below :

1) Primary Research: Identifying the user requirements conducting a survey based on a questionnaire.

2) Secondary Research: Based on these researches the result is defined as the Software Requirement Specification document.

3.2 Purpose

The online quiz application Application provides facility to conduct quizzes online without any manual effort. It saves time as it allows number of students to give the exam at a time and displays the results after the examination finishes,needing no waiting time for the result. Results

are automatically generated by the server. Setter has a privilege to create, modify and delete the test papers and its particular questions as specified CRUD operations in the salient features. User can login and give the test with his specific id, and can see the results as well.

3.3 Functional Requirements

This section in detail describes the use cases for each of the Users separately. The different types of users are.

1) Normal User (Student): He/She is the one who logs on to the system to take the Quiz.

2) Admin: The admin is responsible for entering the questions into the database for the quiz. The admin is also entrusted with other functions such as moderation of comments and posts in the discussion forum, resetting of passwords, overlooking the test etc.

3) Faculty: The faculty is the one who sets the question paper and also receives the detailed report and marks of all the students undertaking the test.

3.4 Non Functional Requirements

Security : Enhanced security for sensitive data that should be accessed only by the authorized users. It should be made sure that users who are given specific rights can perform certain tasks, thus making sure that authorization is intact.

Backup : A back up of the students quiz choices must be maintained such that in case of any power failure or untoward happening the student can resume his quiz from the moment of occurrence of the incident thus saving unwarranted anxiety and concern.

Platform/Browser Independence : The client side system should work on any modern web browsers like Edge, Firefox, Chrome, Opera and all Modern Operating Systems like Windows, Linux and Mac-OS.

Ease of use : The interface should be user friendly, intuitive and easy to navigate and browse about. Care must be taken not to clutter the UI with too much information and the display of questions is neatly presented for the user to answer.

Flexibility : The system should accommodate any future changes in any of the aspects. The application must also be able to handle future integration with other similar applications and also must be generic for it being put to several uses.

Performance : The Quiz application should be able to handle a traffic of around 300 users at any given point of time.

The Use case diagrams, sequence and activity diagrams are explained in detail in the SRS.

Chapter 4

Design

This section deals with the design phase of the application development.

4.1 Class diagram

It is a diagram which is a static structure of a system that shows the systems classes, their attributes and the operations or methods along with their relationships among the classes.The main aim of the setter to create new quizzes and questions and the corresponding answer for them.The user has to attempt the quiz and obtain his/her results. The admin is responsible for moderation and managing users.This diagram makes the view more clear.

4.2 E-R Diagram

The E-R diagram elucidates the data or information aspects of a business domain or its process requirements, in an abstract way that lends itself to ultimately being implemented in a database such as a relational database. The main components of ER models are entities (things) and the relationships that can exist among them.An entityrelationship model is the result of using a systematic process to describe and define a subject area of business data. It does not define business process; only visualize business data. The data is represented as components (entities) that are linked with each other by relationships that express the dependencies and requirements between them. Diagrams created to represent these entities, attributes, and relationships graphically are called entityrelationship diagrams.

An ER model is typically implemented as a database. In the case of a relational database, which stores data in tables, every row of each table represents one instance of an entity. Some data fields in these tables point to indexes in other tables; such pointers are the physical implementation of the relationships.The self explanatory E-R diagram describes the entities and their relationships.

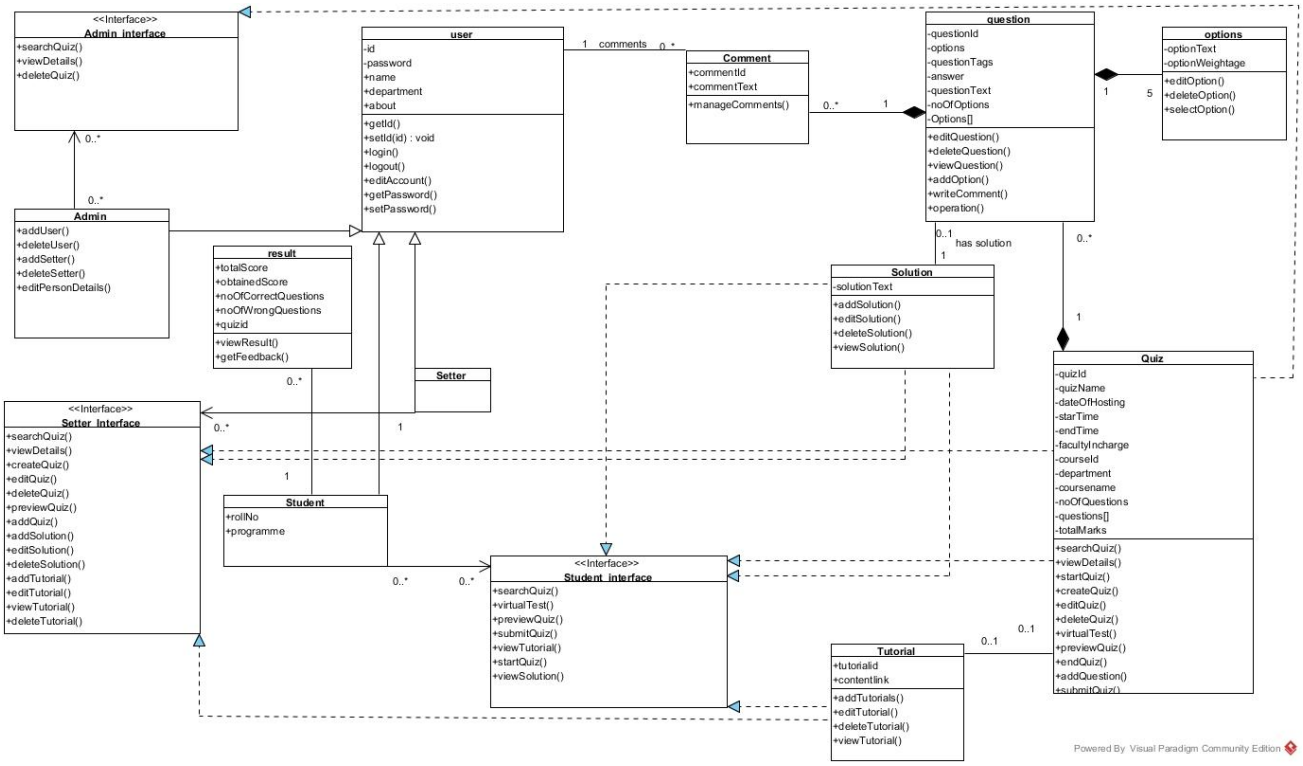


Figure 4.1: Class Diagram

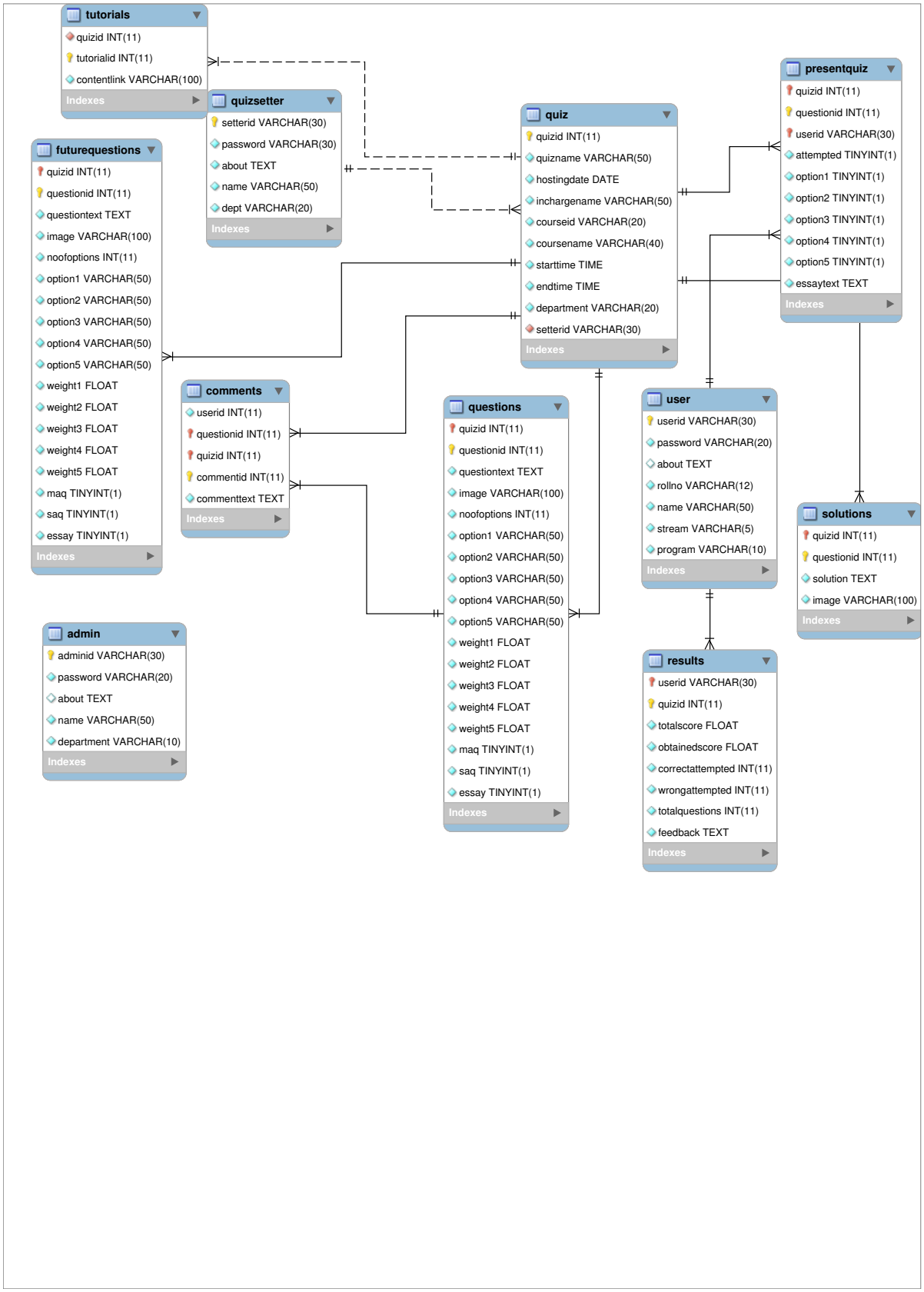


Figure 4.2: E-R Diagram

4.3 Modular Architecture

Quiz Setter:

1.Authentication phase
2.Quiz and subsequent Question generation.
3.Result Viewing and Analysis.

Student/End user :

1.Authentication
2.Enrol for Quiz
3.Quiz attempt/Virtual test
4.Result Generation/ View solutions/ Read Tutorials
5.Analysis/ Report Viewing

Chapter 5

Implementation

The project Implementation started with the setter and admin sections and their respective CRUD generations. The create,read,update and delete operations were implemented using an efficient Content Management system mechanism.This was mainly done in Semester 7.The Step by step work done in this semester is as follows.

The next step was development of an authenticaton system with password hashing and encryption so that security breaching is made impossible.

Next task at hand was developing the presentquiz section for students to have access to questions and later we developed a system for calculating marks for multiple options with different weights and including negative marking scheme for questions.

Developing file upload mechanism for signing up users instead of admin manually adding all of them. Script would be run back-end to ensure this functionality implementation.

The next hurdle overcome was including the timer for a quiz.

Added an option for setter to allow or not to allow user to have multiple attempts for a quiz.

Added AJAX feature for question submission so it won't clog up the network and put extra load on the database.

Retaining the options for the questions attempted so that user can recollect what all he/she attempted later on.

Provided the Changing password feature and edit profile option for all the users.

A priority timer is accommodated along with quiz so that students will know which all challenges are about to come in near future.

The next task was dividing the quizzes set into past and present with different options for each.

Created a leader board with access for both students and setter for that particular quiz.This is

helpful to track one's own performance as well as relative performance.

Options for setter for randomization of questions or options or both or none has been developed.

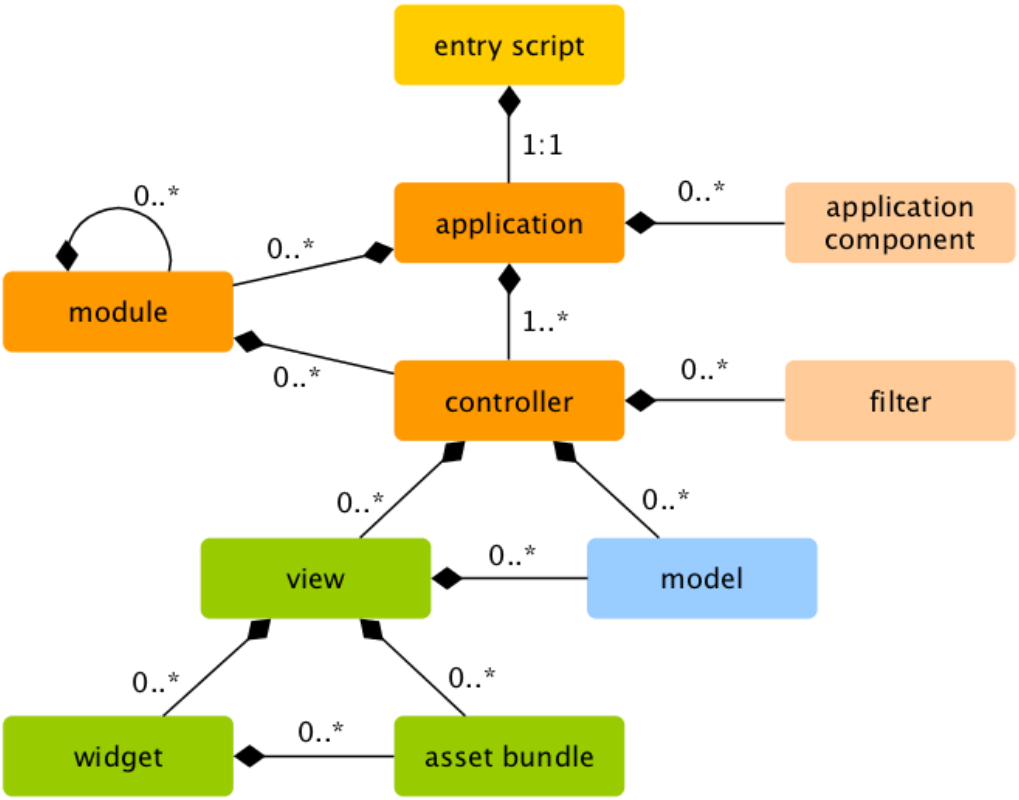
Added an enrolment key option for giving permission to selected students for attempting the quiz.

Virtual test option is developed keeping in mind that it will be better to give students an option of practising past quizzes or quizzes they missed.The main aim of this feature is to be helpful in placement preparation.

Added text editor for question text.

Added solution display option for each question which will be revealed to user after the end of the test.Attempted and correct answers are shown for students.

Developed a graphical representation analysis for student performance in different quizzes and this is of added advantage for setter to analyze difficulty of questions by showing how many students answered a question of a particular difficulty correctly.Dynamically updated and implemented wholes some changes to User Interface using bootstrap.


entry script
1:1
application
0..*
application component
0..*
module
0..*
1..*
0..*
controller
0..*
filter
0..*
0..*
view
0..*
model
0..*
0..*
widget
0..*
asset bundle

Chapter 6

Testing

6.1 Introduction

Testing is a set of activities that are planned in advance and conducted systematically.The process of testing requires that the developer discard preconceived notations of the correctness of the software just developed and overcome a conflict of interest that occurs when errors are encountered. Testing also provides the main objective of our project and understand the risk of implementation. Testing is a process of technical investigation, performed on behalf of stakeholder, that is intended to reveal quantity related information about the product with respect to the context in which it is intended to operate. Testing is the process of executing a program or an application with an intent of finding an error or bugs. Testing can be stated as the process of validating and verifying that a software program/application/product:

6.2 Test Plan

Test plan will describe about the scope and activities of our modules in the project The analysis and planning of the test plans must be done at the starting of our project. It will provide a unique identifier for our document. Testing should begin in small and proceeds in the large. Exhaustive testing is not possible. .The main aim of our test plans is to produce a correct code with all users requirements satisfied.

6.3 Unit Testing

Unit testing is used to test or verify the functionality of specific section of code. This is written by the developers to ensure that specific function is working or not. A unit is the smallest part of an application. Unit tests are created by programmers or occasionally by white box testers during the development process.

6.4 Integration Testing

Integration testing (sometimes called Integration and Testing, abbreviated "IT") is the phase in software testing in which individual software modules are combined and tested as a group. It occurs after unit testing and before validation testing . Integration testing takes as its input

modules that have been unit tested, groups them in larger aggregates, applies tests defined in an integration test plan to those aggregates, and delivers as its output the integrated system ready for system testing. Integration testing is a type of software that seeks to verify the interface between components. The components are integrated in an iterative way. Integration testing will allow the interface to find the issues more quickly and fixed.It works to expose defects in the interfaces and interaction between integrated components (Modules).

TEST CASE FOR END USER:If the user is just visiting our website then he /she can access only the home page.

TEST CASE FOR LOGIN:If the user has joined into our website they will be provided with an user id And using that they can access their account.

TEST CASE FOR CREATE QUIZ:If the user wants to take a test then they can login as user and select the quiz id and other details and can proceed to next step of setting up their quiz.The main catch here is that only admins/setters can create/update or even maintain a quiz. That implies the users of type student have no say in this matter.

TEST CASE FOR CREATE SESSION:If the user wants to take a test then they can login as user and select their category and can proceed to next step.

TEST CASE FOR TIMER:Checking whether the quiz finishes after the timer and also not allowing users to attempt the quiz after the time runs out.

TEST CASE FOR ENROLLMENT:Checking if a quiz can be taken only by those students who have the enrollment key. This is to prevent unnecessary attempts.

TEST CASE FOR RESULTS:Check whether the result gets updated after the quiz time runs down.

TEST CASE FOR QUIZ ATTEMPT:Check whether a quiz can be attempted as many times as requested by the setter and not more.

Many such test case scenarios have been done and checked to see if the final application is fully working.

Chapter 7

Results

The usage and integration of this application in the institutional framework has a lot of benefits like reduced time wastage, faster results, lesser burden on faculties or quiz setters. The application results in delivering faster results and improved user interaction.The result of the joint effort is a fully functional generic quiz hosting application with an efficient CMS.

Chapter 8

Conclusions

This online quiz system is developed in a generic and efficient manner for conducting examination in a hasslefree manner. It saves time as it allows more number of students to give the exam at a time and displays the results as the test time runs out, making the result process instantaneous. Setter has the privilege to create, modify and delete the Quizzes and its questions. User can register, login and give the test with his specific id, and can see the results as well.The Admin manages the users and the Quizzes.

8.1 Future Enhancements

The next stage to which the project could be extended is developing an online judge to test programs during examinations. The system can automatically compile and execute code, and test them with sample test cases already constructed beforehand. Submitted code may be run with restrictions, including time limit, memory limit, security restriction and so on. The output of the code will be captured by the system, and compared with the standard output. The system will then return the result. When mistakes were found in a standard output the specified messages can be displayed so that users correct the most appropriate part of the code causing the error.

Addition of Discussion Forum where in student can hold intellectually stimulating discussion with faculties is an idea to improve interaction.

Possible addition would be using browser caching facility to further reduce querying and loading up the database.

Bibliography

[1] Roger S Pressman *Software Engineering : A Practitioner's Approach*, Mc Graw Hill Publishers.

[2] *www.yiiframework.com*, YII 2.0, 2015.

[3] Some standard wikipedia articles were also referred to regarding software design, development and testing along with blogs comparing various technologies and their advantages and dis advantages.