ONLINE EXAMINATION SYSTEM

A PROJECT REPORT

Submitted by

DHAVAL PATEL [1519BECE30017]

DHRUVKUMAR D PATEL [1519BECE30025]

In fulfillment for the award of the degree

Of

BACHELOR OF ENGINEERING
In
COMPUTER ENGINEERING

LDRP INSTITUTE OF TECHNOLOGY AND RESEARCH,

GANDHINAGAR

Kadi Sarva VishwaVidyalaya, Gandhinagar

2018 - 2019

ONLINE EXAMINATION SYSTEM

A PROJECT REPORT

Submitted by

DHAVAL PATEL [1519BECE30017]

DHRUVKUMAR D PATEL [1519BECE30025]

In fulfillment for the award of the degree

Of

BACHELOR OF ENGINEERING
In
COMPUTER ENGINEERING

LDRP INSTITUTE OF TECHNOLOGY AND RESEARCH, GANDHINAGAR

Kadi Sarva VishwaVidyalaya, Gandhinagar

2018 - 2019

LDRP Institute of Technology and Research

Computer Engineering Department

CERTIFICATE

This is to certify that the Project Work entitled **"Online Examination System"** has been carried out by **Dhaval N. Patel (1519BECE30017)** under my guidance in fulfillment of the degree of Bachelor of Engineering in Computer Engineering (7th Semester) of Kadi Sarva VishwaVidyalaya University, Gandhinagar during the academic year 2018-19.

Guide:

Prof. Hiren Patel
Internal Guide,
LDRP-ITR.

Prof. Hiren Patel
H.O.D. CE Dept.,
LDRP ITR.

LDRP Institute of Technology and Research

Computer Engineering Department

CERTIFICATE

This is to certify that the Project Work entitled **"Online Examination System"** has been carried out by **Dhruvkumar D Patel [1519BECE30025]** under my guidance in fulfillment of the degree of Bachelor of Engineering in Computer Engineering (7th Semester) of Kadi Sarva VishwaVidyalaya University, Gandhinagar during the academic year 2018-19.

Guide:

Prof. Hiren Patel
Internal Guide,
LDRP-ITR.

Prof. Hiren Patel
H.O.D. CE Dept.,
LDRP ITR.

ACKNOWLEDGEMENT

We express our unfathomable sense of pleasure towards Mrs. Gargi Rajpara Principle of LDRP-ITR, who gave us a chance to work. She frequently came to motivate and provided such an extraordinary infrastructure and resources to work in her premises.

We express our deep sense of gratitude and indebtedness to our internal guide Mr. Ashish Patel for accepting us to work under her training and supervision. He took prolonged interest in our work and directed to us toward the predefined goal. He has shown us a way to pursue excellence. He has been a big factor of motivation in our project.

We are also very thankful to our HOD of CE/IT department Prof. Hiren Patel and all the faculty members who provided to be a constant motivation for the knowledge acquisition and the moral support throughout the whole project.

We sincerely thank our Computer Engineering Department, LDRP-ITR for the academic advancement it has provide to us during the whole diploma course and provided an opportunity for project training.

Last but not the least my heartily thanks is to my friends and batch mates, who have provided us with innumerable discussions on many technicalities and friendly tips.

We would like to express our heartiest gratitude to our family and friends who has always guided us towards the path of success and those entire people who have directly or indirectly helped us in making this project work successful.

Dhaval Patel
(1519BECE30017)

Dhruvkumar D Patel
(1519BECE30025)

ABSTRACT

Online Examination System is web based application for technical evaluation. Online Quiz Examination System not only replace paperwork but also releases the workload of the faculty. It fulfills the requirements of the institutes to conduct the exams online. Students can give exam without the need of going to any physical destination. They can view the result at the same time.Thus the purpose of the site is to provide a system that saves the efforts and time of Faculty.

This Web Application provides facility to conduct online examination worldwide. It saves time as it allows number of students to give the exam at a time and display the result as the test gets over, so no need to wait for the result. It is automatically generated by the server.

Administrator has to create, modify and delete the test paper. While user can register, login and give the test with his specific id and can see the result as well.

LIST OF FIGURES

NO	NAME	PAGE NO
1	Work Breakdown Structure	9
2	Activity Network Requirement	10
3	Gantt Chart	10
4	Usecase Diagram	23
5	Sequence Diagram	25
6	Data flow diagram	28
7	Flow Chart	29

LIST OF TABLES

NO	NAME	PAGE NO
1	tbl_user	24
2	employee	24
3	tbl_log	25
4	tbl_test	25
5	tbl_result	25
6	tbl_question	26
7	tbl_response	26
8	tbl_user_test	26
9	tbl_parent_question	27
10	tbl_category	27
11	tbl_question_test	27
12	tbl_category_sub_category_list	27
13	tbl_subcategory	28

TABLE OF CONTENTS

NO	CHAPTER NAME	PAGE NO
	Acknowledgement	v
	Abstract	vii
	List Of Figures	viii
	List Of Tables	ix
	Table Of Contents	x
1	**Introduction**	1
	1. Introduction	2
	2. Scope	2
	1.2.1Current Scope	2
	1.2.2 Future Scope	2
	3. Project Summary and purpose	
	1.3.1Project Summary	3
	1.3.2Purpose	3
	4. Objectives	3
2	**Technology And Literature Review**	4
	2.1 Tools And Technology	5
	2.2 Definition Acronyms	7

3	**System Requirement Study**	9
	3.1 User Characteristics	10
	3.2 Software and Hardware Requirements	10
	3.2.1 Software Requirements	10
	3.2.2 Hardware Requirements	11
	3.2.3 Functional Requirements	11
	3.2.4 Non Functional Requirements	12
	3.3 Constraints	12
	3.3.1 Parallel Operations	12
	3.3.2 Reliability Requirements	12
	3.3.3 Criticality of the application	13
	3.3.4 Safety and security considerations	13
	3.3.5 Hardware limitations	13
	3.3.6 Regulatory Policies	13
	3.3.7 General Constraint	13
	3.4 Assumptions and Dependencies	14
	3.4.1 Assumptions	14
	3.4.2 Dependencies	14
4	**System Analysis**	15
	4.1 Study of current system	16
	4.2 Modules and Functionality of proposed system	16
	4.3 Feasibility Study	16
	4.3.1 Technical feasibility	16
	4.3.2 Operational feasibility	17
	4.3.3 Economic feasibility	17
	4.4 Requirement Validation	18
	4.5 Class Diagram	18
	4.6 Use Case Diagrams	20
	4.7 Sequence Diagram	21
5	**System Design**	23
	5.1 Database Design	30
	5.2 Data Flow Diagram	

6	**Implementation Standards**	32
	6.1 Implementation Environment	33
	6.2 Security Features	33
	6.3 Coding Standards	34
7	**Testing**	37
	7.1 Testing Plan	38
	7.1.1 Planning Steps	38
	7.2 Testing Strategies	40

1 INTRODUCTION

- **INTRODUCTION**
- **SCOPE**
- **PROJECT SUMMARY AND PURPOSE**
- **OBJECTIVES**

1. Introduction

This project assesses students by conducting online objective tests. The tests would be highly customisable. This project will enable educational institutes to conduct test and have automated checking of answers based on the response by the candidates.

The project allows faculties to create their own tests. It would enable educational institutes to perform tests, quiz and create feedback forms. It asks faculty to create his/her set of questions. Faculty then creates groups and adds related students into the groups. Further the tests are associated with specific groups so that only associated students can appear for the test. The result of the response would be available to the faculty of the question set. This project would be helpful for creating practice tests, say for educational institutes.

2. Scope

1.2.1 Current Scope

- This project would be very useful for educational institutes where regular evaluation of students' is required.
- The main purpose of this system is to efficiently evaluate candidate thoroughly through a fully automated system that not only same time but also gives fast result and saves paper.
- It is cost-efficient and popular means of mass evaluation system.
- The faculty prepare he test and question for each exam.
- Can be used anywhere any time as it is a web based application(user Location doesn't matter).
- The candidate can login through client computer with unique id and can give exam.
- Questions are objective type like MCQ, fill in the blank, true or false.
- After completion of exam result will be shown.

1.2.2 Future Scope

- Faculties of one subject can create a question bank where every one can put their question and random selection of question for a particular test are chosen from question bank.
- Shuffling of questions if there are sitting next to each other.
- Adding of short notes & descriptive type question

3. Project Purpose

3.1. Purpose

- This Web Application provides facility to conduct online examination world wide.
- It saves time as it allows number of students to give the exam at a time and displays the results as the test gets over, so no need to wait for the result. It is automatically generated by the server.
- Administrator has a privilege to create, modify and delete the test papers and its particular questions.
- User can register, login and give the test with his specific id, and can see the results as well

4. Objectives

- Online Examination will reduce hectic jobs assessing answer given by the candidates manually
- Begin an integrated Online Examination system it will reduce paper work #GoGreen
- To allow faculty to create their own test and answer key
- To allow Automatic grading and manual grading which can be recorded per test
- Result will be shown immediately to the participating students .

2 TECHNOLOGY AND LITERATURE REVIEW

- **TOOLS AND TECHNOLOGY**
- **PROJECT PLANNING**
- **PROJECT SCHEDULING**

2.1 Tools and Technology

2.1.1 Angular js

AngularJS is a JavaScript-based open-source front-end web framework mainly maintained by Google and by a community of individuals and corporations to address many of the challenges encountered in developing single-page applications. It aims to simplify both the development and the testing of such applications by providing a framework for client-side model–view–controller (MVC) and model–view–viewmodel (MVVM) architectures, along with components commonly used in rich Internet applications. (This flexibility has led to the acronym MVW, which stands for "model-view-whatever" and may also encompass model–view–presenter and model–view–adapter.) In 2014, the original AngularJS team began working on the Angular web framework.

The AngularJS framework works by first reading the Hypertext Markup Language (HTML) page, which has additional custom HTML attributes embedded into it. Angular interprets those attributes as directives to bind input or output parts of the page to a model that is represented by standard JavaScript variables. The values of those JavaScript variables can be manually set within the code, or retrieved from static or dynamic JSON resources 30 years.

2.1.2 Spring Boot

Spring Boot makes it easy to create stand-alone, production-grade Spring based Applications that you can "just run".

Spring Boot is a Spring framework module which provides RAD (**Rapid Application Development**) feature to the Spring framework. It is highly dependent on the **starter templates** feature which is very powerful and works flawlessly.

Spring Boot starters are templates that contain a collection of all the relevant transitive dependencies that are needed to start a particular functionality.

2.1.3 IntelliJ IDEA

IntelliJ IDEA is a Java integrated development environment (IDE) for developing computer software. It is developed by JetBrains (formerly known as IntelliJ), and is available as an Apache 2 Licensed community edition, and in a proprietary commercial edition. Both can be used for commercial development.

Every aspect of IntelliJ IDEA is specifically designed to maximize developer productivity.

Together, the powerful static code analysis and ergonomic design make development not only productive but also an enjoyable experience.

2.1.4 MySQL Database:

MySQL is the most popular Open Source Relational SQL Database Management System. MySQL is one of the best RDBMS being used for developing various web-based software applications. MySQL is developed, marketed and supported by MySQL AB, which is a Swedish company. This tutorial will give you a quick start to MySQL and make you comfortable with MySQL programming.

2.2 Definitions, Acronyms

The sub-section provides the definitions of all terms, acronyms, and abbreviations used in this document to understand the SRS properly.

Sr. No.	Terms/Acronyms	Description
1.	Student	User mostly a student who will appear for the examination
2.	Faculty	Another user mostly faculty member, lecturer or examiner who posts set of questions, the available options and correct answers.
3.	Administrator	Super user, adds faculty and manages system.

2.2.1 Functional or Specific Requirements

Required software is for conducting on-line 'objective' type examination and providing immediate results. The system should satisfy the following requirements:

✤ **Administrator Aspect**

1. Taking backup of the database
2. Editing/Deleting/Creating the database.
3. Adding or expelling faculty
4. Changing the super password.

✤ **Faculty Aspect**

1. Logging into the system.
2. Creating a test
3. Posting questions in the above test
4. Posting multiple options to respective question
5. Marking correct answer within the given options
6. Time limit of the test if any

✤ **Student Aspect**

1. Logging into the system
2. Edit user information
3. Selecting the test.
4. Printing the result at the end of the examination
5. Reviewing the given responses.

3 SYSTEM REQUIREMENTS STUDY

- **USER CHARACTERISTICS**
- **HARDWARE AND SOFTWARE REQUIREMENTS**
- **CONSTRAINTS**
- **ASSUMPTIONS AND DEPENDENCIES**

3.1 User Characteristics

Analysing user characteristics is an important aspect of any project. It allows us to clearly define and focus on who the end users are for the project. Also, it allows checking the progress of the project to ensure that we are still developing the system for the end users. The user must have following characteristics:

- User must have basic knowledge of Computers.
- User should understand the use of all modules.
- User can easily interact with the proposed system.
- User must know the technical terms used in the for examination for performing different tasks specially related to logins, log out, start test etc
- User should be also being aware about the running process of the system.

3.2 Software and Hardware Requirements

Software and Hardware Requirements are used to describe the minimum hardware and software requirements to run the Software. These requirements are described below.

3.2.1 Software Requirements

- **Client:**
  - Operating System: Windows or Linux or OSX device
  - Internet connectivity required
  - Web browser:- chrome, firefox
- **Server:**
  - Operating System: WINDOWS or LINUX SERVER OS
  - Database Server: MySQL
  - Technologies: spring boot, jdk, InteliJ IDE

3.2.2 Hardware Requirements

- **Client:**
  - ➢ 1.6 GHZ CPU
  - ➢ 256 MB of RAM
  - ➢ Internet Connection
  - ➢ Monitor
  - ➢ Keyboard/Mouse
  - ➢ Printer
  - ➢ Windows or Linux or Mac OSX

- **Server:**
  - ➢ 1.6 GHZ CPU
  - ➢ 8 GB of RAM
  - ➢ Internet Connection
  - ➢ Monitor
  - ➢ Keyboard/Mouse

3.2.3 Functional Requirements

The basic features available in the system are as follows:

- Admin can search any details of reports easily.
- Admin can view various reports based on different criteria's.
- Faculty can manages the tests, add new test etc.
- Users can refer to the log details of their own, the analysis they have done or are further going to do.

3.2.4 Non Functional Requirements

Following is a list of non functional requirements:

- **Performance**:

  This system should remain accessible 24x7. At least 1000 users should be able to access the system altogether at any given time.

- **Security:**

  The database of system should not store any password in plain text rather the ceasar cipher text has to be stored.

- **Reliability:**

  It can be accessed by the end users 24*7 as an when needed hence is highly reliable for end users.

- **Availability:**

  Internet connection for the nodes with the database server is ensured and hence the application will be available any time for access.

- **Portability:**

  The developed web application is portable as it can be accessed from any operation system regardless Windows, Mac, Linux provided they have a browser to access Internet.

3.3 Constraints

3.3.1 Parallel Operations

The project is on basis of multi-user. This is used for carrying out updating as well as entry by preventing the redundancy of the data.

3.3.2 Reliability Requirements

Reliability requirements of the system are one of the prime ones in the list. The system is needed to be highly reliable in terms of performance and capable of delivering robust performance. If the reports are generated within 5 seconds then the system is said to be reliable.

3.3.3 Criticality of the Application

The system can stop working on computers with very low internet connection. Other than that there won't be any issues. Apart from these the system should be able to make updates at regular time intervals.

3.3.4 Safety and Security Consideration

Safety and security too are other major concerns of any system. It is necessary to provide safety and security as the system is web application and might be intrude by security threats from the internet. Thus, the code needs to be encrypted and any transaction needs to be done securely.

3.3.5 Hardware Limitations

Hardware Limitations are other constraint of the system. Hardware Limitations should be overcome for better performance of the system. This can be achieved by using minimum and only necessary hardware.

3.3.6 Regulatory Policies

Regulatory policy is about achieving organization's objectives through the use of regulations, laws, and other instruments to deliver better economic and social outcomes and thus enhance the life of business. Thus the system should be developed by using these regulations to provide better outcome to the company.

3.3.7 General Constraints

The service provided by developers is 24*7 to its client companies. Hence the oracle database server as well as the web application needs to be up 24*7. Internet availability is the major requirement for all the end users to successfully access the application. As well as being a live application security enforcement needs to be tight else there are chances of hacking or intrusion.

3.4 Assumptions and Dependencies

3.4.1 Assumptions

- Database transactions are assumed to be secure and reliable.
- User is the person having enough knowledge for the traversing operation.
- We will provide a user friendly interface so that any user can easily navigate through the system, but he/she should be capable of providing valid credentials for successful login.
- The server used for data storing is always secured.

3.4.2 Dependencies

- The system is dependent upon the user's valid credentials.
- This application depends on the server and internet as all the information is collected and then stored in the server through secure internet connection.
- All the users of the system will be assigned a specific role. According to these roles each and every user will be allowed to access predefined set of features.

4 SYSTEM ANALYSIS

- **STUDY OF CURRENT SYSTEM**
- **PROPOSED SYSTEM**
- **FEASIBILITY STUDY**
- **REQUIREMENTS VALIDATION**
- **CLASS DIAGRAM**
- **SYSTEM ACTIVITY (USECASE DIAGRAM)**
- **SEQUENCE DIAGRAM**

4.1 Study of current system

- The Existing system of conducting examination process is manual. It has so many problems. So we introduce a new system, which is fully computerized. Existing system is a large man power process and is difficult to implement. Working of existing system is given below:
- Student Registration is the first process . As the part of the registration, the student has to enter his name, address etc into the registration form. After the registration, make the question papers and it will give to the prospective student. The question papers contain total mark, subject, duration, question paper etc.
- A group of person does evaluation of answer sheet. After the evaluation of the Answer sheet , the result is published. And also make the mark list.

4.2 Proposed system

The main objective of the online examination system is that it helps companies/institutions to conduct exams to any number of candidates at a time, in an automated manner. It reduces the time consumption and workload that exist in the current system of examination. It also helps in storing the record of each examination and the results are also stored in the system. This makes the searching of the records easier than the existing system.

4.3 FEASIBILITY STUDY

4.1. ECONOMIC FEASIBILITY

- Economic analysis is most frequently used for evaluation of the effectiveness of the system. More commonly knows as cost/benefit analysis the procedure is to determine the benefit and saving that are expected from a system and compare them with costs, decisions is made to design and implement the system.
- This part of feasibility study gives the top management the economic justification for the new system. This is an important input to the management the management, because very often

the top management does not like to get confounded by the various technicalities that bound to be associated with a project of this kind. A simple economic analysis that gives the actual comparison of costs and benefits is much more meaningful in such cases.

- In the system, the organization is most satisfied by economic feasibility. Because, if the organization implements this system, it need not require any additional hardware resources as well as it will be saving lot of time.

4.2. TECHNICAL FEASIBILITY

- Technical feasibility centers on the existing manual system of the test management process and to what extent it can support the system. According to feasibility analysis procedure the technical feasibility of the system is analyzed and the technical requirements such as software facilities, procedure, inputs are identified. It is also one of the important phases of the system development activities.
- The system offers greater levels of user friendliness combined with greater processing speed. Therefore, the cost of maintenance can be reduced. Since processing speed is very high and the work is reduced in the maintenance point of view management convince that the project is operationally feasible.

4.3. BEHAVIOURAL FEASIBILITY

- People are inherently resistant to change and computer has been known to facilitate changes. An estimate should be made of how strong the user is likely to move towards the development of computerized system. These are various levels of users in order to ensure proper authentication and authorization and security of sensitive data of the organization.

4.4 Requirements Validation

The Basic validation from user side is to detect wrong information or blank information:

- When the user has entered correct username and password, then he/she is allowed to enter information or to see the information either into the database or from the database.
- If anyone field from username or password remains blank, then user will not be allowed to enter into the system.
- If a consultant leaves some fields blank in call log form then he/she does not allowed submitting his/her information.
- If user has entered wrong data then accordingly message will generate automatically.

4.5 Class Diagram

A class diagram is a graph of classified elements connected by their various static relationships. It is shown here for Client Support System. This includes the System and the End-users as its main classes. Here are three different fields: Class, Attributes, and Operations.

Class shows the class name, i.e., System, End-user, Data and Sign Clip. They are connected with each other through links and their relation with each other is shown through the numbers represented on the link; here * indicates zero or more multiplicity. Here Data class is connected to System through a Composition link which is the collaboration of all participants are part of one composite class.

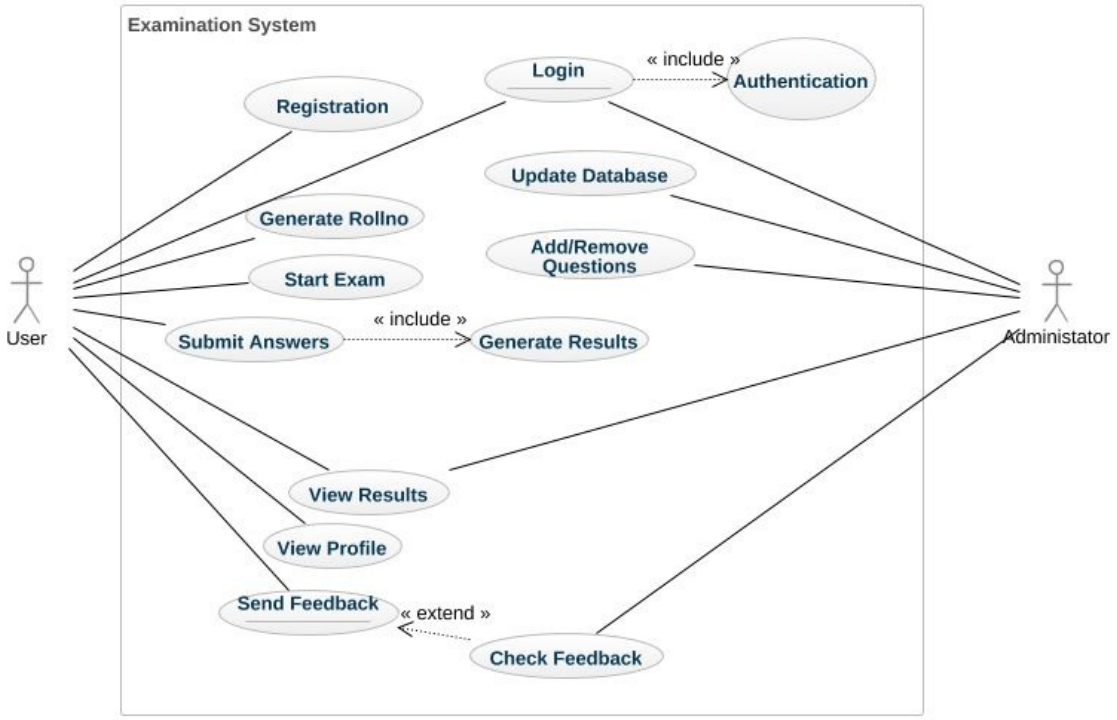
Attributes provide the details of the Class while Operations show all possible operations respective class can do in the system.

tbl_user
user_id INT(11)
dob DATE
email VARCHAR(255)
gender VARCHAR(255)
name VARCHAR(255)
password VARCHAR(255)
user_type VARCHAR(255)
username VARCHAR(255)
country VARCHAR(255)
is_unique BIT(1)
Indexes
employee
id INT(11)
email VARCHAR(255)
name VARCHAR(255)
Indexes
tbl_test
test_id INT(11)
cutoff INT(11)
subcategory_id INT(11)
test_name VARCHAR(255)
total_marks INT(11)
time INT(11)
Indexes
tbl_subcategory
subcategory_id INT(11)
category_id INT(11)
subcategory_name VARCHAR(255)
Indexes
tbl_category
category_id INT(11)
category_name VARCHAR(255)
Indexes
tbl_competency_analysis
competency_id INT(11)
percent_obt DOUBLE
test_id INT(11)
topic_name VARCHAR(255)
user_id INT(11)
Indexes
tbl_category_sub_category_list
category_category_id INT(11)
sub_category_list_subcategory_id INT(11)
Indexes
tbl_log
log_id INT(11)
percent_obt DOUBLE
test_id INT(11)
timestamp DATETIME
user_id INT(11)
Indexes
tbl_test_parent_question_list
test_test_id INT(11)
parent_question_list_parent_question_id INT(11)
Indexes
tbl_question
question_id INT(11)
answer VARCHAR(255)
marks INT(11)
parent_question_id INT(11)
question_type VARCHAR(255)
subtopic_name VARCHAR(255)
question_desc VARCHAR(255)
difficulty INT(11)
option_id INT(11)
Indexes
tbl_response
response_id INT(11)
question_id INT(11)
response VARCHAR(255)
timestamp DATETIME
user_id INT(11)
Indexes
tbl_user_test
user_test_id INT(11)
test_id INT(11)
user_id INT(11)
Indexes
test_model
test_question_id INT(11)
difficulty1 VARCHAR(255)
test_question_desc VARCHAR(255)
subjectivity1 VARCHAR(255)
test_id1 INT(11)
topic_name1 VARCHAR(255)
Indexes
tbl_parent_question
parent_question_id INT(11)
difficulty VARCHAR(255)
parent_question_desc VARCHAR(255)
subjectivity VARCHAR(255)
test_id INT(11)
topic_name VARCHAR(255)
Indexes
tbl_subskill_analysis
subskill_id INT(11)
percent_obt DOUBLE
subtopic_name VARCHAR(255)
test_id INT(11)
user_id INT(11)
Indexes
hibernate_sequence
next_val BIGINT(20)
tbl_result
result_id INT(11)
marks INT(11)
question_id INT(11)
result VARCHAR(255)
user_id INT(11)
Indexes
tbl_question_test
question_test_id INT(11)
question_id INT(11)
test_id INT(11)
parent_question_id INT(11)
Indexes
tbl_options
option_id INT(11)
option_a VARCHAR(255)
option_b VARCHAR(255)
option_c VARCHAR(255)
option_d VARCHAR(255)
Indexes

4.6 System Activity

A use case diagram shows the relationship among actors and use cases within a system. Hence it provides the characteristics of the actors whose behavior and relationships can be well understood using the diagrams elaborated here.

An end-user can perform various tasks on the application; he may use signs or upload a new sign o his own. On the other hand, the system would respond to the operations done by the user. It



would display the sign clips and store the uploaded sign to the defined category. Also it would provide a message whenever needed for confirmation.

Here the rectangle indicates system boundary, out of which there are actors found who perform various operations on the system which are the end-user and the system here. An elliptical shape shows the use-case while the connecting links between an actor and a use case are said to be communicates.

4.7 Sequence Diagram

A sequence diagram represents an Interaction, which is a set of messages exchanged among objects within collaboration to effect a desired operation or result. Here are the sequence diagrams for various interactions among the user, system and the data storage. It must be noted that the rectangle box on the top of the diagram indicates the object or actor and dashed lines beneath to it shows an object's lifeline. Another rectangles following and followed by the dashed lines in a vertical manner show the activation period of the object or actor when it performs some actions. A solid arrow conveys a message while the dashed arrow gives return message. These message names are written along with their respective arrows as shown further in the diagram.

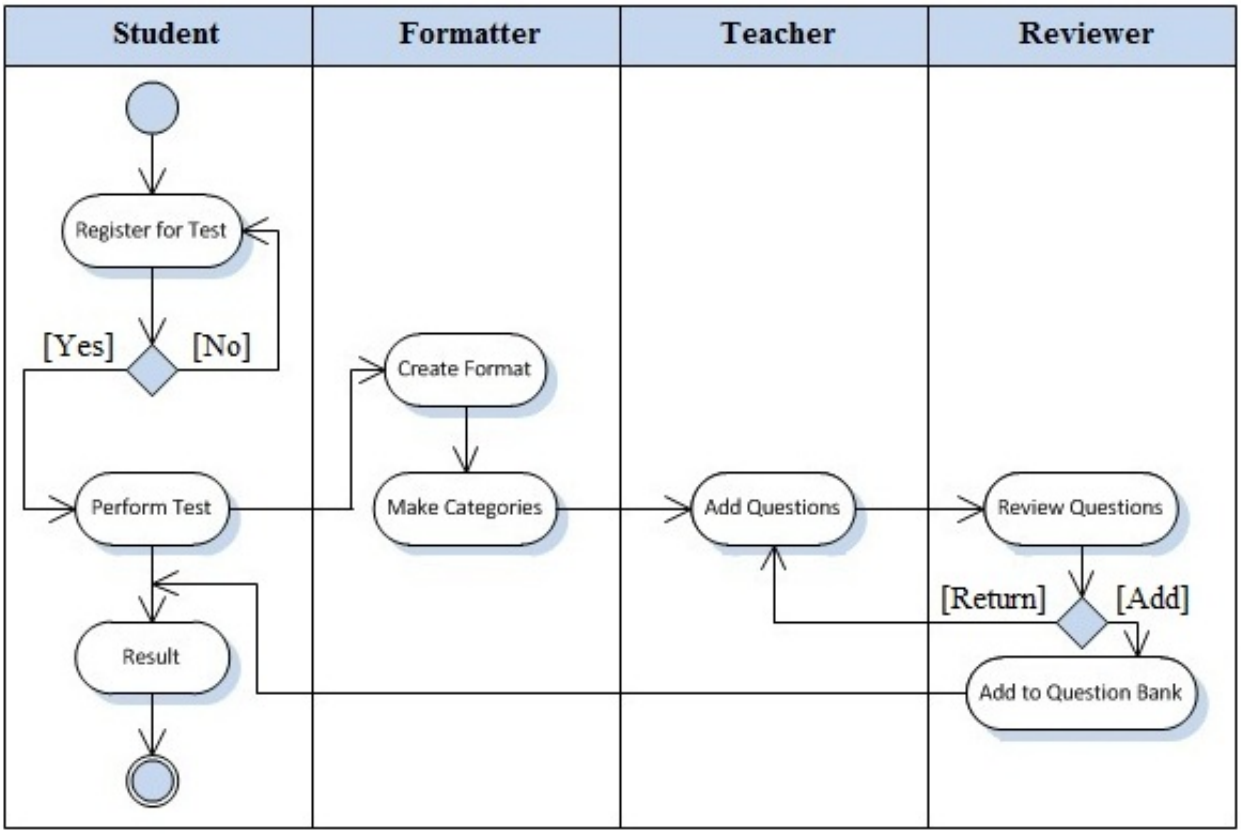


Fig 4.2: "Sequence Diagram"

5 SYSTEM DESIGN

- **DATABASE DESIGN**
- **DATA FLOW DIAGRAM**
- **FLOW CHART**

5.1 Database Design

The current web application uses MySQL Database to store, access and retrieve the data.

Following is the data dictionary that describes the required tables along with their fields.

5.1.1 DATA TABLES

1. tbl_user

NAME	DATATYPE	CONSTRAIN
user_id	VARCHAR	Primary Key
dob	DATE	
email	VARCHAR	
gender	VARCHAR	
name	VARCHAR	
password	VARCHAR	
user_type	VARCHAR	
username	VARCHAR	
country, is_unique	VARCHAR	
Is_unique	BIT	

2. employee

NAME	DATA TYPE	CONSTRAIN
id	VARCHAR	Primary Key
Email	VARCHAR	
Name	VARCHAR	

3. tbl_log

NAME	DATA TYPE
log_id	INT
percent_obt	DOUBLE
test_id	INT
timestamp	DATETIME
user_id	INT

4. tbl_test

NAME	DATA TYPE
test_id	INT
cutoff	INT
subcategory_id	INT
test_name	VARCHAR
total_marks	INT
time	INT

5. tbl_result

NAME	DATA TYPE
result_id	INT
marks	INT
question_id	INT
result	VARCHAR
user_id	INT

6. tbl_question

NAME	DATA TYPE
question_id	INT
answer	VARCHAR
marks	INT
parent_question_id	INT
question_type	VARCHAR
subtopic_name	VARCHAR
question_desc	VARCHAR
difficulty	INT
option_id	INT

7. tbl_response

NAME	DATA TYPE
response_id,	INT
question_id	INT
response	VARCHAR
timestamp	DATETIME
user_id	INT

8. tbl_user_test

NAME	DATA TYPE
user_test_id	INT
test_id	INT
user_id	INT

9. tbl_parent_question

NAME	DATA TYPE
parent_question_id	INT
difficulty	VARCHAR
parent_question_desc	VARCHAR
subjectivity	VARCHAR
test_id	INT
topic_name	VARCHAR

10. tbl_catagory

NAME	DATA TYPE
category_id	INT
category_name	VARCHAR

11. tbl_question_test

NAME	DATA TYPE
question_test_id	INT
question_id	INT
test_id	INT
parent_question_id	INT

12. tbl_catagory_sub_catagory_list

NAME	DATA TYPE
category_category_id	INT
sub_category_list_subcategory_id	INT

13. tbl_subcatagory

NAME	DATA TYPE
subcategory_id	INT
category_id	INT
subcategory_name	INT

14. tbl_hibernate_sequence

NAME	DATA TYPE
next_val	BIGINT

15. tbl_competency_analysis

NAME	DATA TYPE
competency_id	INT
percent_obt	DOUBLE
test_id	INT
topic_name	VARCHAR
user_id	INT

16. tbl_test_parent_question_list

NAME	DATA TYPE
test_test_id	INT
parent_question_list_parent_question_id	INT

17. tbl_subskill_analysis

NAME	DATA TYPE
subskill_id	INT
percent_obt	VARCHAR
subtopic_name	DOUBLE

NAME	DATA TYPE
test_id	INT
user_id	INT

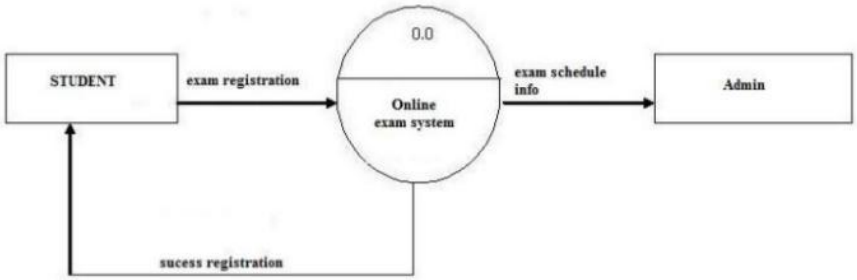
18. test_model

NAME	DATA TYPE
test_question_id	INT
difficulty1	VARCHAR
test_question_desc	VARCHAR
subjectivity1	VARCHAR
test_id1	INT
topic_name1	VARCHAR

19. tbl_options

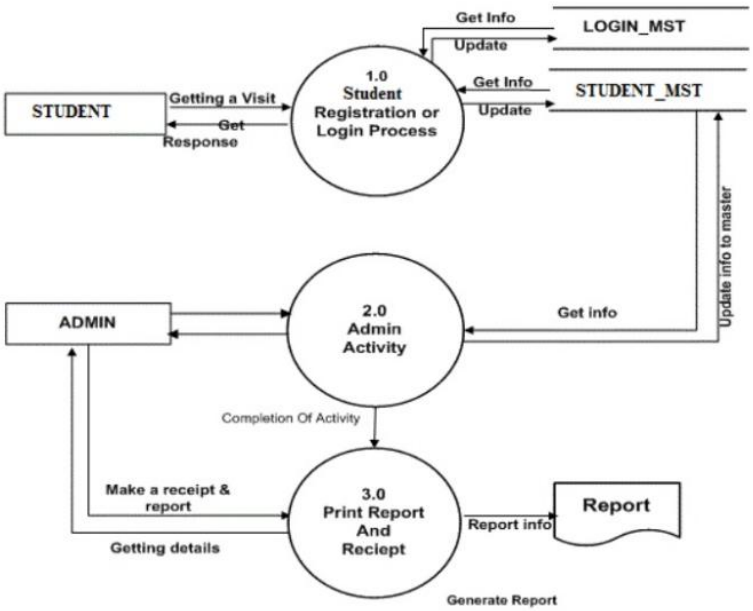
NAME	DATA TYPE
option_id	INT
option_a	VARCHAR
option_b	VARCHAR
option_c	VARCHAR
option_d	VARCHAR

5.2 Data Flow Diagram



Level 0

FIG 5.1: "DATA FLOW DIAGRAM LVL 0"

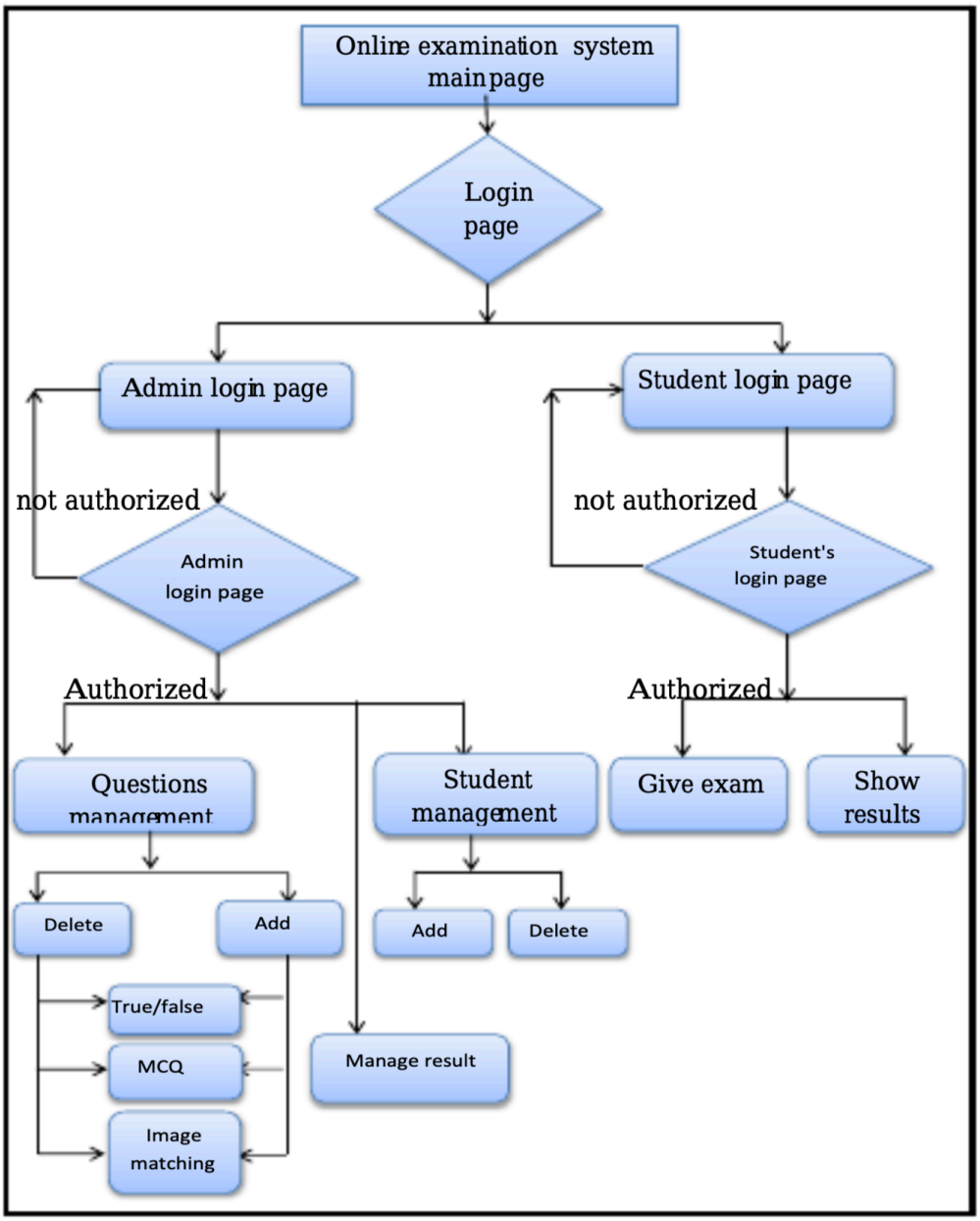


Level 1

FIG 5.1: "DATA FLOW DIAGRAM LVL 1"

5.3 FLOW CHART

"FlowChart"



6 IMPLEMENTATION STANDARDS

- **IMPLEMENTATION ENVIRONMENT**
- **SECURITY FEATURES**
- **CODING STANDARDS**

6.1 Implementation Environment

Challenges identified for successful design and implementation of this project are dominated by: complexity, reliability/availability, transparent data access while respecting security, a high experiment data rate and data volume, scientific exploitation from multiple sites. All the systems have a GUI interface to interact with the user.

6.1.1 Single User

This Software is a single-user Application. Only a single instance can run at a time in a pre installed hardware.

6.1.2 GUI Based

This Software consists of GUI that is user friendly but the background process consists of much center of attraction. That is written in such a way that manipulation of multiple images can be easily done.

6.2 Security Features

Any infrastructure/application development for the web should adhere to the Security Policy detailed by the organization. It is assumed that Organization has a detailed Security policy for protecting its resources and information. What follows is a “common” approach for protecting the resources and information when a web application is published in the web.

Security Layers:

- ✓ Database Security
- ✓ Network - Data Transfer Security

1. Database Security

Server-Enforced, Row Level Access Control - The system will be the using MySQL Database. This database will be deployed on the database server and database access checks will be defined

at the system level. No direct access is granted to Database, all access to this database is via applications and hence governed by Access controls.

6.3 Coding Standards

Single Responsibility Principle

Do define one thing, such as a service or component, per file.

Consider limiting files to 400 lines of code.

Why? One component per file makes it far easier to read, maintain, and avoid collisions with teams in source control.

Why? One component per file avoids hidden bugs that often arise when combining components in a file where they may share variables, create unwanted closures, or unwanted coupling with dependencies.

Why? A single component can be the default export for its file which facilitates lazy loading with the router.

The key is to make the code more reusable, easier to read, and less mistake prone.

Do define small functions

Consider limiting to no more than 75 lines.

Why? Small functions are easier to test, especially when they do one thing and serve one purpose.

Why? Small functions promote reuse.

Why? Small functions are easier to read.

Why? Small functions are easier to maintain.

Why? Small functions help avoid hidden bugs that come with large functions that share variables with external scope, create unwanted closures, or unwanted coupling with dependencies.

Naming

General Naming Guidelines

Do use consistent names for all symbols.

Do follow a pattern that describes the symbol's feature then its type. The recommended pattern is feature.type.ts.

Why? Naming conventions help provide a consistent way to find content at a glance. Consistency within the project is vital. Consistency with a team is important. Consistency across a company provides tremendous efficiency.

Why? The naming conventions should simply help find desired code faster and make it easier to understand.

Why? Names of folders and files should clearly convey their intent. For example, app/heroes/hero-list.component.ts may contain a component that manages a list of heroes.

Separate file names with dots and dashes

Do use dashes to separate words in the descriptive name.

Do use dots to separate the descriptive name from the type.

Do use consistent type names for all components following a pattern that describes the component's feature then its type. A recommended pattern is feature.type.ts.

Do use conventional type names including .service, .component, .pipe, .module, and .directive. Invent additional type names if you must but take care not to create too many.

Why? Type names provide a consistent way to quickly identify what is in the file.

Why? Type names make it easy to find a specific file type using an editor or IDE's fuzzy search techniques.

Why? Unabbreviated type names such as .service are descriptive and unambiguous.

Abbreviations such as .srv, .svc, and .serv can be confusing.

Why? Type names provide pattern matching for any automated tasks.

Bootstrapping

Do put bootstrapping and platform logic for the app in a file named main.ts.

Do include error handling in the bootstrapping logic.

Avoid putting app logic in main.ts. Instead, consider placing it in a component or service.

Why? Follows a consistent convention for the startup logic of an app.

Why? Follows a familiar convention from other technology platforms.

7 TESTING

- **TESTING PLANS**
- **TESTING STRATEGIES**
- **TEST CASES**

7.1 Testing Plan

7.1.1 Planning Steps

1) Functionality Testing

2) Interface testing

3) Performance testing

4) Security testing

1) Functionality Testing:

Test for – all the links in web pages, database connection, forms used in the web pages for submitting or getting information from user, Cookie testing.

➢ **Check all the links:**

- Test the outgoing links from all the pages from specific domain under test.
- Test all internal links.
- Test links jumping on the same pages.
- Test links used to send the email to admin or other users from web pages.
- Test to check if there are any orphan pages.
- Lastly in link checking, check for broken links in all above-mentioned links.

➢ **Database testing:**

Data consistency is very important in web application. Check for data integrity and errors while you edit, delete, modify the forms or do any DB related functionality.

Check if all the database queries are executing correctly, data is retrieved correctly and also updated correctly. More on database testing could be load on DB, we will address this in web load or performance testing below.

2) Interface Testing:

The main interfaces are:

- Web server and application server interface
- Application server and Database server interface.

Check if all the interactions between these servers are executed properly. Errors are handled properly. If database or web server returns any error message for any query by application server then application server should catch and display these error messages appropriately to users. Check what happens if user interrupts any transaction in-between? Check what happens if connection to web server is reset in between?

3) Performance Testing

Stress testing: Generally stress means stretching the system beyond its specification limits. Web stress testing is performed to break the site by giving stress and checked how system reacts to stress and how system recovers from crashes. Stress is generally given on input fields, login and sign up areas.

4) Security Testing:

Following are some test cases for web security testing:

- Test by pasting internal url directly into browser address bar without login. Internal pages should not open.
- If you are logged in using username and password and browsing internal pages then try changing url options directly. I.e. If you are checking some publisher site statistics with publisher site ID= 123. Try directly changing the url site ID parameter to different site ID which is not related to logged in user. Access should denied for this user to view others stats.

- Try some invalid inputs in input fields like login username, password, input text boxes. Check the system reaction on all invalid inputs.
- Web directories or files should not be accessible directly unless given download option.
- Test if SSL is used for security measures. If used proper message should get displayed when user switch from non-secure http:// pages to secure https:// pages and vice versa.
- All transactions, error messages, security breach attempts should get logged in log files somewhere on web server.

7.2 Testing Strategies

- **White Box Testing:**

White box testing (WBT) is also called **Structural or Glass box testing**. White box testing involves looking at the structure of the code. When you know the internal structure of a product, tests can be conducted to ensure that the internal operations performed according to the specification. And all internal components have been adequately exercised.

➢ **Why we do White Box Testing?**

To ensure:

- That all independent paths within a module have been exercised at least once.
- All logical decisions verified on their true and false values.
- All loops executed at their boundaries and within their operational bounds internal data structures validity.

➢ **Need of White Box Testing?**

To discover the following types of bugs:

- Logical error tend to creep into our work when we design and implement functions, conditions or controls that are out of the program

- The design errors due to difference between logical flow of the program and the actual implementation
- Typographical errors and syntax checking

Limitation Of WBT:

Not possible for testing each and every path of the loops in program. This means exhaustive testing is impossible for large systems. This does not mean that WBT is not effective. By selecting important logical paths and data structure for testing is practically possible and effective.

- **Black Box Testing:**

  - Black box testing treats the system as a **"black-box"**, so it doesn't explicitly use Knowledge of the internal structure or code. Or in other words the Test engineer need not know the internal working of the "Black box" or application.
  - **Main focus in black box testing is on functionality of the system as a whole.** The term **'behavioural testing'** is also used for black box testing and white box testing is also sometimes called **'structural testing'**. Behavioural test design is slightly different from black-box test design because the use of internal knowledge isn't strictly forbidden, but it's still discouraged.
  - Each testing method has its own advantages and disadvantages. There are some bugs that cannot be found using only black box or only white box. Majority of the application are tested by black box testing method. We need to cover majority of test cases so that most of the bugs will get discovered by black box testing.
  - Black box testing occurs throughout the software development and Testing life cycle i.e. in Unit, Integration, System, Acceptance and regression testing stages.
  - **Advantages of Black Box Testing**

    - Tester can be non-technical.

- Used to verify contradictions in actual system and the specifications.

- Test cases can be designed as soon as the functional specifications are complete

- **Disadvantages of Black Box Testing**

  - The test inputs needs to be from large sample space.

  - It is difficult to identify all possible inputs in limited testing time. So writing test cases is slow and difficult. Chances of having unidentified paths during this testing.

8 BIBLIOGRAPHY

- **CONCLUSION**
- **BIBLIOGRAPHY**

8.1 CONCLUSION

Online Examination System is web based application for technical evaluation. Online Quiz Examination System not only replace paperwork but also releases the workload of the faculty. It fulfills the requirements of the institutes to conduct the exams online. Students can give exam without the need of going to any physical destination. They can view the result at the same time.Thus the purpose of the site is to provide a system that saves the efforts and time of Faculty.

This Web Application provides facility to conduct online examination worldwide. It saves time as it allows number of students to give the exam at a time and display the result as the test gets over, so no need to wait for the result. It is automatically generated by the server.

Administrator has to create, modify and delete the test paper. While user can register, login and give the test with his specific id and can see the result as well.

8.2 BIBLIOGRAPHY

Web Resources

- https://angular.io/
- https://www.udemy.com/
- https://spring.io/projects/spring-boot