
SOFTWARE DESIGN DOCUMENTATION

for

MCQ Test Simulation

Version 1.0

Prepared by

Reena Yadav(IIT2016089)

Vikash Kumar(IIT2016041)

Austin Kispotta(IIT2016051)

Manvendra Dattatrey(IIT2016504)

Sai Sathy Praful Tondomker(IIT2016082)

October 12, 2018

Contents

1	Introduction	3
1.1	Purpose	3
1.2	Scope	3
1.3	Overview	3
2	Design Overview	4
2.1	User Interfaces	4
2.2	Technologies Used	4
2.3	Communications Interfaces	4
3	System Architecture	5
4	System Features	6
4.1	Sign-In/Login Page	6
4.1.1	Description and Priority	6
4.2	MCQ test	6
4.2.1	Description and Priority	6
4.3	Creating contest	6
4.3.1	Description and Priority	6
5	System Object Model	7
5.1	Introduction	7
5.2	Subsystems	7
5.2.1	MCQ Test Simulation Utility Package	7
5.3	Subsystem Interfaces	7
5.3.1	None Defined	7
6	Object Descriptions	8
6.1	User	8
6.2	Contest	8
6.3	Student Response	8
6.4	Result	8
7	Data Design	9
7.1	Entity Relationship Diagram	9
7.2	Database design	10
8	Activity Design	11
8.1	Activity Diagram	11
9	Dynamic Model	12
9.1	Sequence Diagram	12
10	Supplementary Documentation	13
10.1	UML Modeling tools	13
10.2	Entity Relationship Diagramming Tools	13
10.3	For Modeling and Compiling	13

1 Introduction

The Software Design Document is a document to provide documentation which will be used to aid in software development by providing the details for how the software should be built. Within the Software Design Document are narrative and graphical documentation of the software design for the project including use case models, sequence diagrams, collaboration models, object behavior models, and other supporting requirement information

1.1 Purpose

The purpose of the Software Design Document is to provide a description of the design of a system fully enough to allow for software development to proceed with an understanding of what is to be built and how it is expected to built. The Software Design Document provides information necessary to provide description of the details for the software and system to be built.

1.2 Scope

This Software Design Document is for a base level system which will work as a proof of concept for the use of building a system that provides a base level of functionality to show feasibility for providing a suitable platform. This Software Design is focused on the base level system and critical parts of the system. For this particular Software Design Document, the focus is placed on generation of the documents and modification of the documents.

1.3 Overview

The Software Design Document is divided into 10 sections with various subsections. The sections of the Software Design Document are:

- Introduction
- Design Overview
- System Architecture
- System Features
- System Object Model
- Object Descriptions
- Data Design
- Activity Design
- Dynamic Model
- Supplementary Documentation

2 Design Overview

2.1 User Interfaces

- LogIn and Authentication
- Contest Creation

2.2 Technologies Used

- **Frontend Software** :- HTML, CSS, BOOTSTRAP (For designing interactive UI)
- **Backend Software** :- NodeJS (Act as a midway between FrontEnd and database)
- **Database Used** :- MongoDB (To save the user realtime information and records)

2.3 Communications Interfaces

- **Express package** in NodeJS for connecting to server through HTTP protocol.
- **nodemailer package** in NodeJS for e-mail communication.



Figure 2.1: Use Case Diagram

3 System Architecture

- **Login Interface** - The interface which will authenticate users logging into to website.
- **Test Interface** - The main interface used to conduct the test.
- **Data storage** - The interface for storing, importing and exporting the data model and raw collected data.

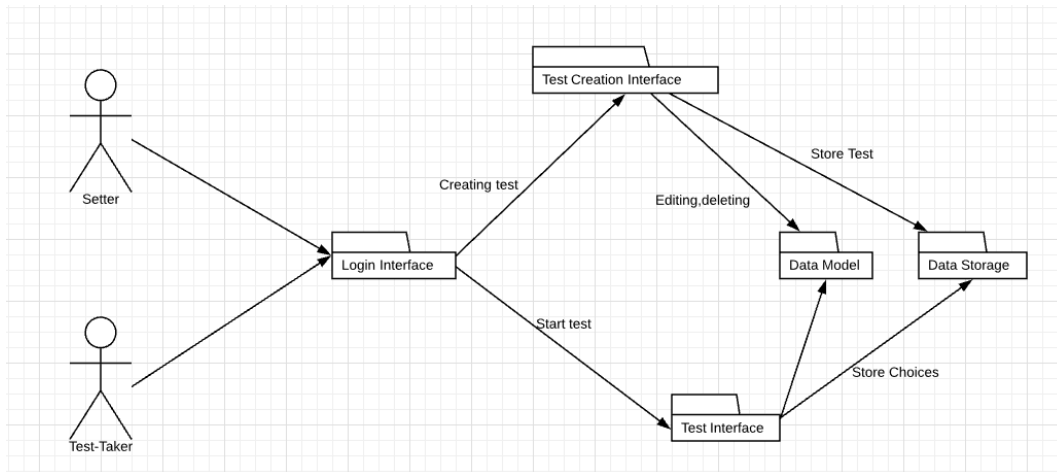


Figure 3.1: Architecture

4 System Features

Following are the major services provided by this software -

4.1 Sign-In/Login Page

4.1.1 Description and Priority

The web software will have a authentication login page through which users login for the test. Every user will also have their own unique login credentials for better security. As this is the initial step before the main test portal, this is the highest priority feature. Refer Figure 4.1.

4.2 MCQ test

4.2.1 Description and Priority

This is the main feature of the software. This page will contain questions and users will be required to select the best possible answer. Refer Figure 4.2.

4.3 Creating contest

4.3.1 Description and Priority

The other important feature of the software is for the test setter to have the ability to create a contest with required number of questions and specify the time limit. Refer Figure 4.3.

5 System Object Model

5.1 Introduction

The System Object Model Section allows for a description of the subsystems in use. This allows for describing the system in a overall manner to show the different groupings of parts into respective systems. For the System Under Design, only one system is used and no subsystems are specified.

5.2 Subsystems

5.2.1 MCQ Test Simulation Utility Package

This system software design supports analyze, configure, optimize and maintain a computer without compromising ordinary users requirements . Following aspects were taken care of while designing:-

- Specify any requirements regarding security or privacy issues surrounding use of the product or protection of the data used or created by the product
- Define any user identity authentication requirements. Refer to any external policies or regulations containing security issues that affect the product.
- Define any security or privacy certifications that must be satisfied.

5.3 Subsystem Interfaces

5.3.1 None Defined

As the system is contained with in a single package, external interfaces are used but internal interfaces are not necessary.

6 Object Descriptions

6.1 User

Objects of this class is responsible for adding, editing, or deleting the details of the users mentioned below in the system.

- Test Setter :- Interacts with Contest class and responsible for managing contest related data.
- Examinees :- Interacts with Student Response class and has following methods-
 - Selects and registers in particular contests.
 - Answers questions in contests
 - Can ask doubts in registered contests.

6.2 Contest

Objects of this class interacts with Test Setter class objects and manages below attributes-

- Unique Contest ID
- Name of the contest
- Time Limit

This class object has a specialization as **Contest Questions** class and it's specialized attributes are as mentioned below:

- Question ID
- Question Content
- Choices
- Correct Answer

6.3 Student Response

Objects of this class interacts and maintains records of Examinee class and also helps in its validation.

6.4 Result

Examinee class instance can access scorecard using an instance of Result class. Result class interacts with Contest class objects to acquire information about contest users.

7 Data Design

7.1 Entity Relationship Diagram

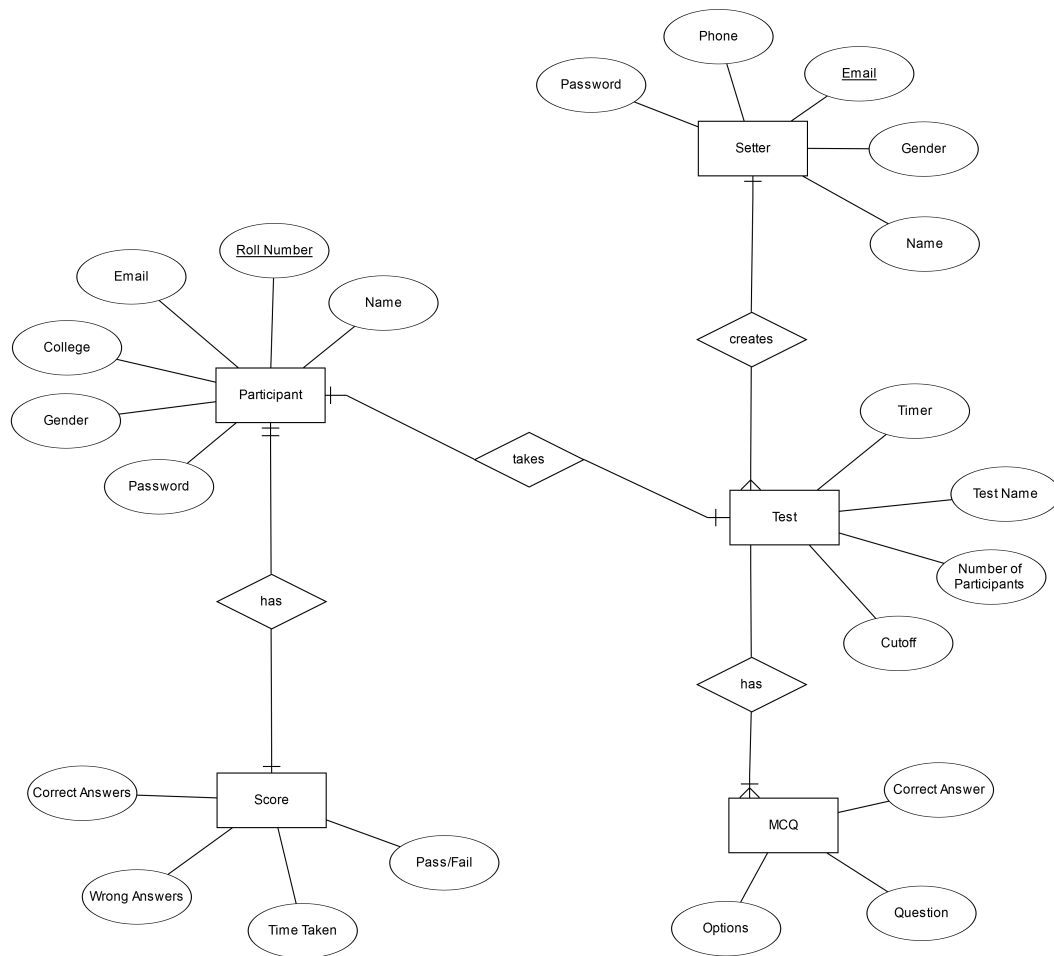


Figure 7.1: Entity Relationship Diagram

7.2 Database design

Participant	
Field	Data Type
Name	VARCHAR(30)
Password	VARCHAR(30)
Roll Number	VARCHAR(30)
Email	VARCHAR(30)
College	VARCHAR(30)
Gender	VARCHAR(30)
Setter	
Field	Data Type
Name	VARCHAR(30)
Password	VARCHAR(30)
Email	VARCHAR(30)
College	VARCHAR(30)
Gender	VARCHAR(30)
Score	
Field	Data Type
Correct Answers	Integer
Wrong Answers	Integer
Time Taken	Timestamp
Pass/Fail	Boolean
Test	
Field	Data Type
Timer	Integer
Test Name	Integer
Participants	Timestamp
Cutoff	Boolean
MCQ	
Field	Data Type
Correct Answers	Integer
Question	VARCHAR(80)
Options	VARCHAR(80)

9 Dynamic Model

9.1 Sequence Diagram

This diagram show the overview level sequence form moving from data and template to a document.

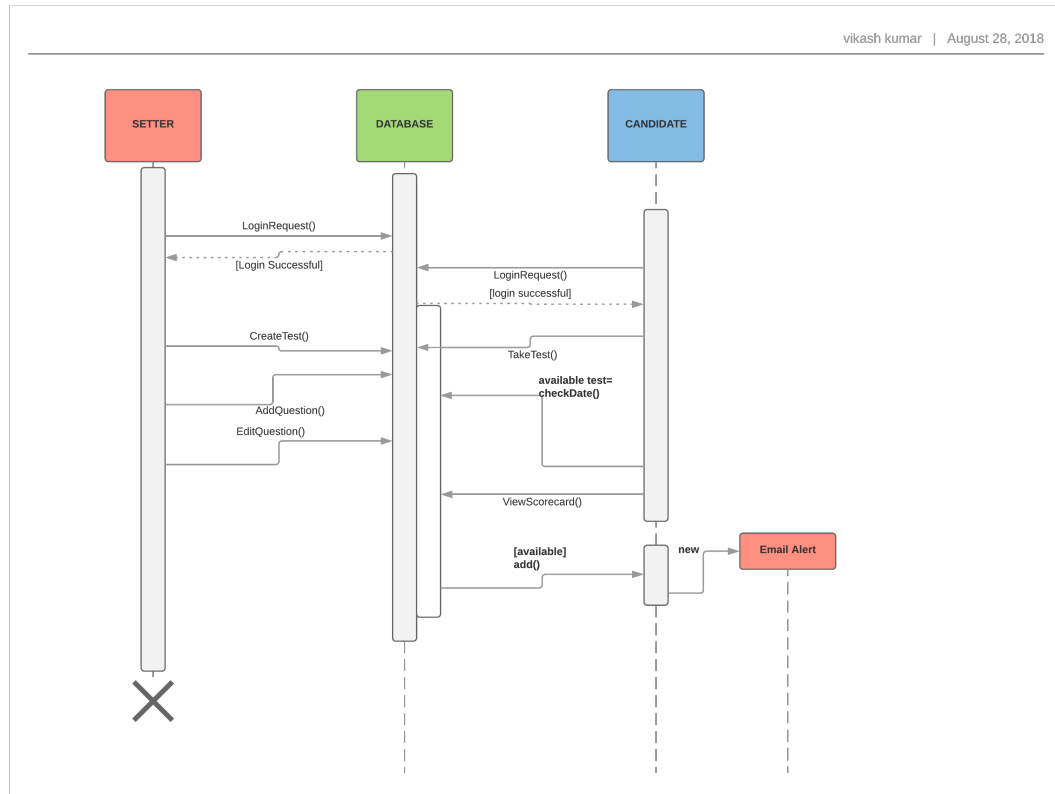


Figure 9.1: Sequence Diagram

10 Supplementary Documentation

10.1 UML Modeling tools

Lucidchart <https://www.lucidchart.com/>

10.2 Entity Relationship Diagramming Tools

ERDPlus <https://erdplus.com/>

10.3 For Modeling and Compiling

TexStudio - Version 2.12.10 <https://www.texstudio.org/>