Software Requirements Specification

for

AI Proctored Examination System

Version 1.0 approved

Prepared by Nerds of a Feather™

Veermata Jijabai Technological Institute

1st November 2020

Table of Contents

Table of Contents

Revision History

1. Introduction
- 1.1 Purpose
- 1.2 Document Conventions
- 1.3 Intended Audience and Reading Suggestions
- 1.4 Product Scope
- 1.5 References

2. Overall Description
- 2.1 Product Perspective
- 2.2 Product Functions
- 2.3 User Classes and Characteristics
- 2.4 Operating Environment
- 2.5 Design and Implementation Constraints
- 2.6 User Documentation
- 2.7 Assumptions and Dependencies

3. External Interface Requirements
- 3.1 User Interfaces
- 3.2 Hardware Interfaces
- 3.3 Software Interfaces
- 3.4 Behaviour Requirements
- 3.5 Database Schema

4. System Features
- 4.1 System Feature 1
- 4.2 System Feature 2 (and so on)

5. Other Nonfunctional Requirements
- 5.1 Performance Requirements
- 5.2 Safety Requirements
- 5.3 Security Requirements
- 5.4 Software System Attributes

1. Introduction

The following subsections of the SRS document provide an overview of the entire SRS.

1.1 Purpose

In COVID times, when exams all over the world are conducted virtually,there is an indispensable need for such a platform. The use of Artificial Intelligence to come up with a more intelligent and secure approach to tackle this problem ensuring integrity of unsupervised examinations is the purpose of this Project. This Web Application provides facility to online examinations Nationwide. It saves time as it allows a number of students/candidates to give the exam at same time and display the results at the end of the test, so there is no need to wait for the result. It is evaluated and generated by the server. Administrator has privileges to create, modify and delete the test papers and its particular questions. Users can register, login and give the test with his specific id, and can see the results as well.

1.2 Document Conventions

For **Main Heading** Font size : 18,Font : Times , Font Type: Bold
For **Subheading** : Font size : 13; Font : Times , Font Type: Bold
For Content : Font size : 11, Font : Times , Font Type: Regular

Margin of Page : Left : 0.9”,Right : 0.9”,Top : 1” ,Bottom : 1”
Alignment : Justify, Line Spacing : 1.15
In Header : Right hand side : Page number , Left hand side : Project Title

1.3 Intended Audience and Reading Suggestions

The Unsupervised test can benefit organizations by leveraging the automation and live proctoring using AI to deliver a secure and confidential environment to conduct online exams. Colleges and universities can conduct online tests while maintaining integrity. Assessment providers, E-learning programs can provide Training and certification using our platform.

1.4 Product Scope

The website to conduct online examinations is “OnlineExamination”. This website provides facilities to institutes to conduct online exams by providing a unique id to each institute. The institute provides questions along with positive and negative marks. Institute also enters the list of eligible students. All the information entered can be later edited by the institute.

In turn students can login with their id, name and institute to give the exams and can view their result then and there. Institutes can also view the result of their students.

- This website can be used in educational institutions as well as in the corporate world.

- Can be used anywhere any time as it is a web based application (user location doesn't matter).
- No restriction that the examiner has to be present when the candidate takes the test.
- No manual work of preparing and storing the result information.
- Less time consumption, as the result is calculated immediately after the test and displayed to the student/candidate.

1.5 References

- IEEE Recommended Practice for Software Requirements Specification- IEEE STD 830-1993.
- Chen Ying-ying, "The Design and Implementation of Online Examination System with Characteristics of Cloud Service [D]", Beijing University of Posts and Telecommunications, 2013.
- Muna R. Hameed , Firas. A. Abdullatif, "Online Examination System ", International Advanced Research Journal in Science, Engineering and Technology ISO 3297:2007 Certified

2. Overall Description

2.1 Product Perspective

This is an online examination system. It is accessible via the Internet, 24 hours a day, 7 days a week. The objective of this app is to reach and connect candidates and examiner in remote communities and conduct exams in a virtual environment online. This web-app will only allow the registered users to enter the test module. The various stages in the app are as follows:-

- **Login**
- **System Overview**
- **Test**
- **Result**

Login:
This window offers the user two choices for logging into the system according to the preset privileges Candidate login and Administrator login. The candidate login will take the user to the user profile. The Administrator Login will take the user to the administrator profile.

System Overview:
This window can only be accessed by the administrator. It allows the administrator to add and edit exams.

Test:
This window contains all the exams a candidate can give. All these exams are organized on one window. The test window will be different for students giving different exams.

Result:
This window displays the result of exams the candidate has just appeared. This data will be saved and displayed in the user profile.

2.2 Product Functions

The functions are divided according to the user types such as

Administrator:-
The function of the administrator is to add/edit exams in the test module.

Student:-
The function of the student is to update his/her profile and give various exams.

2.3 User Classes and Characteristics

The various users/Classes in our system are Administrator, Student, Test.

Administrator characteristics:

- userName
- E-mail
- Password

Student Characteristics:

- Name
- E-mail
- Password

Test Characteristics:

- TImer
- Penalty Threshold

2.4 Operating Environment

- This is a web application so the only required software is Web-Browser.
- Hardware requirements are Webcam and working Microphone.

2.5 Design and Implementation Constraints

The candidate is allowed to give the exams any number of times, until specified otherwise by the administrator while creating the test. While giving the exam the candidate is given only a specific amount of time and the remaining time should be displayed, after which the exam should close and display the result.

2.6 User Documentation

- This website will come with a predefined set of rules which will be shown to a user before taking the exam.

- Some hardware and software requirements will be shown before taking the exam.

2.7 Assumptions and Dependencies

Proper working of this web application is dependent on the internet connectivity of the users' computer Assumptions and dependencies:

- It is assumed that the user has basic knowledge of the system.
- It is assumed that the data entered by the user while registering is true.
- It is assumed that basic hardware requirements such as webcam are specified.

3. External Interface Requirements

This section provides software requirements to a level of detail sufficient to enable designers to design the system and testers to test the system.

3.1 User Interfaces / GUI

For Admin

- Admin Registration Screen: Various fields available on this screen will be:

  *Email Id

  *Password

- Admin Login Screen: Fields available on this screen are:

  *Email Id

  *Password

- *Create test: Various Fields are:

  *Questions

  *Options (4)

  *Correct Answer

- *Student List Screen: Various Fields are:

  *Student ID

*Student Name

*Student Login Screen: Various Fields are:

*Student ID

*Student Name

*Institute ID

*Student Taking Exam Screen: Various Fields are:

*Display Of Question With Options

*Control Buttons To switch questions

- *Screen displaying results of all test takers : Various Fields are:

  *No. Of Correct Questions

  *No. Of Incorrect Questions

  *No. Of Unattempted Questions.

  *Total Marks.

  *Result (Pass/Fail)

For Test taker

- Test Taker Registration Screen: Various fields available on this screen will be:

  *Email Id

  *Username

  *Password

  *Photo upload

- Test Taker Login Screen: Fields available on this screen are:

  *Email Id

  *Password

- *Exam Details Screen: Various Fields are:

  *Exam Name

  *No. Of Questions

  *Time Limit

  *+ve, -ve Marks

  *Passing Marks

- *Screen displaying results of all tests attempted by a user : Various Fields are:

  *No. Of Correct Questions

  *No. Of Incorrect Questions

  *No. Of Unattempted Questions.

  *Total Marks.

  *Result (Pass/Fail)

3.2 Hardware Interfaces

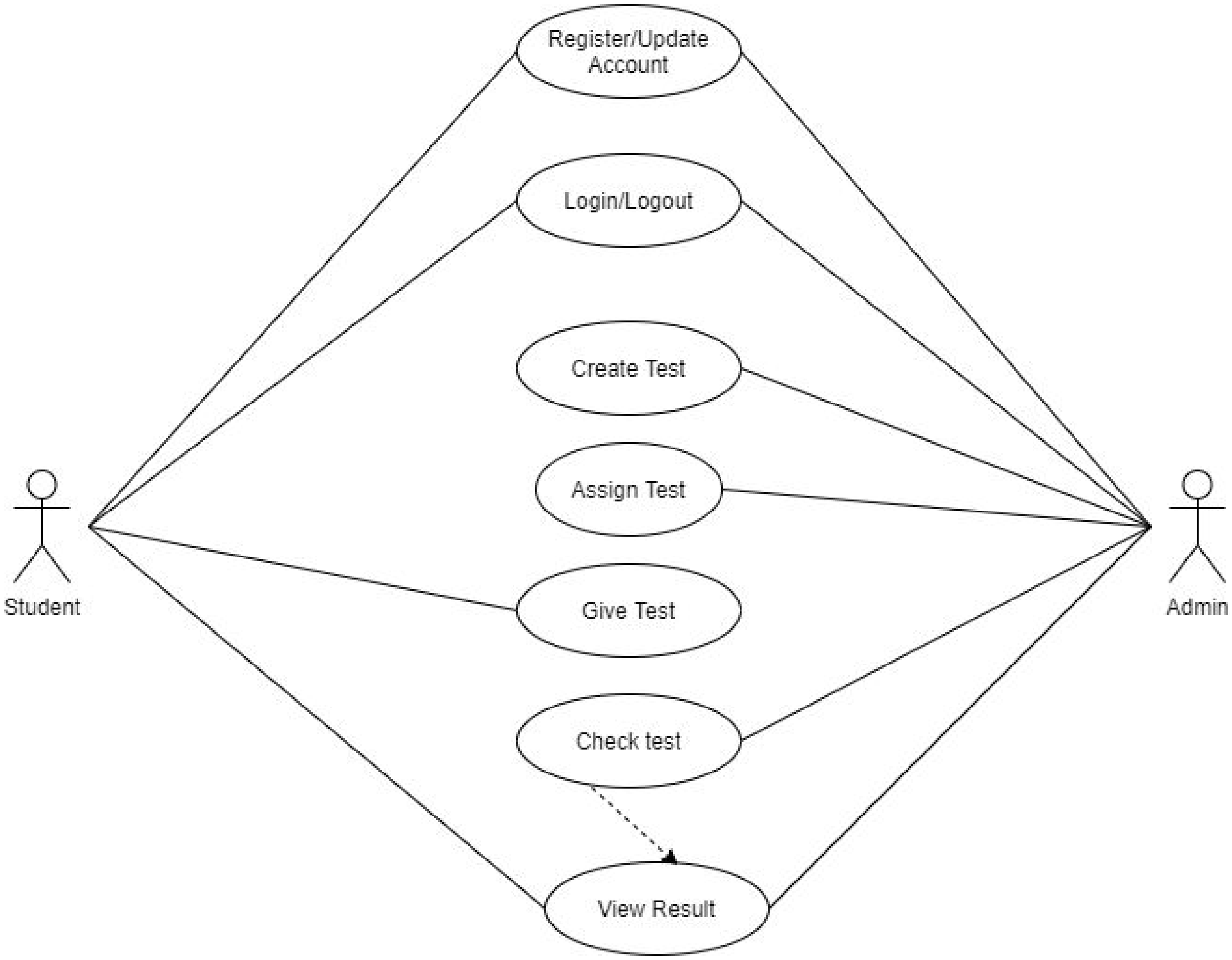
- Support for webcam access for continuous monitoring of users.
- Microphone access/ call over computer audio for monitoring malpractices over audio.
- Screen resolution of at least 800X600 is required for proper and complete viewing of screens. Higher resolution will be accepted.

3.3 Software Interfaces

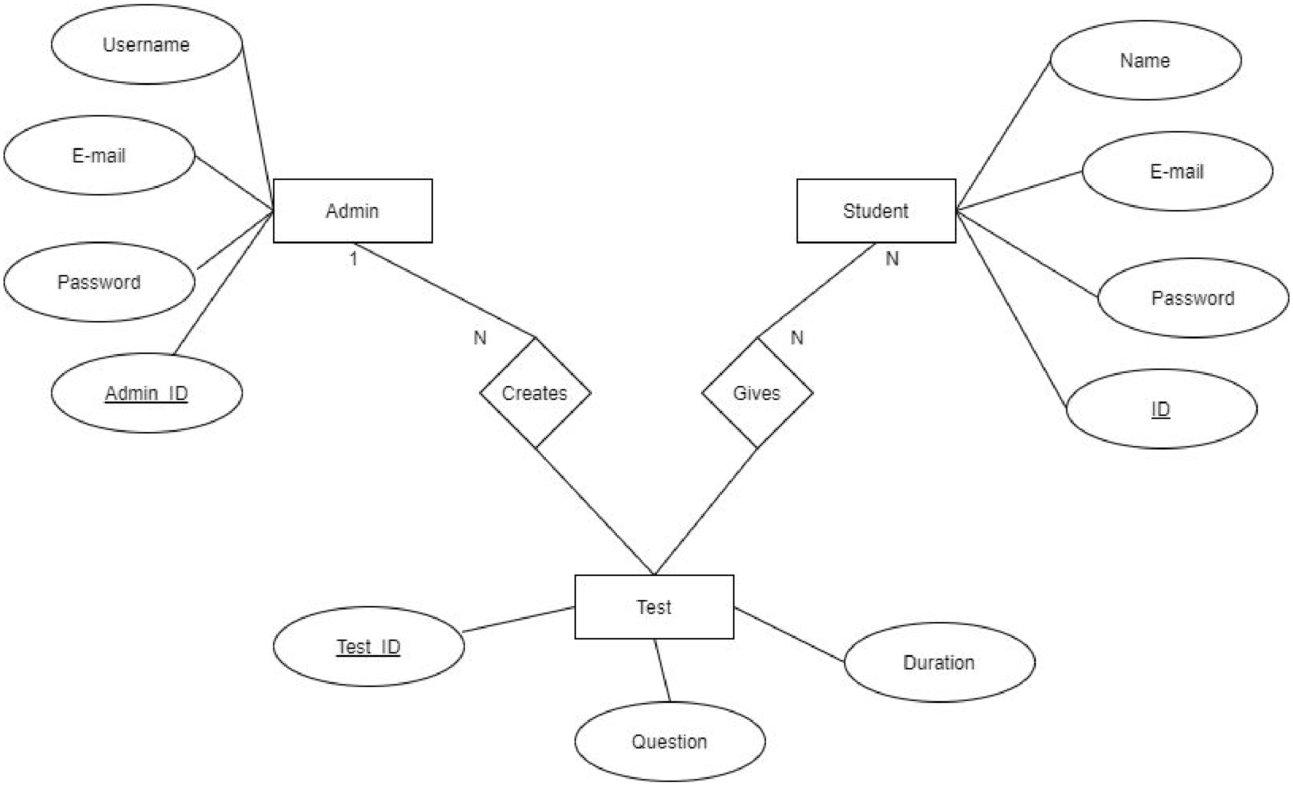
- Any windows based operating system.
- Oracle cloud for NOSQL DBMS-for database.
- IDE (Visual Studio Code) and Jupyter Notebook for developing code.

3.4 Behaviour Requirements

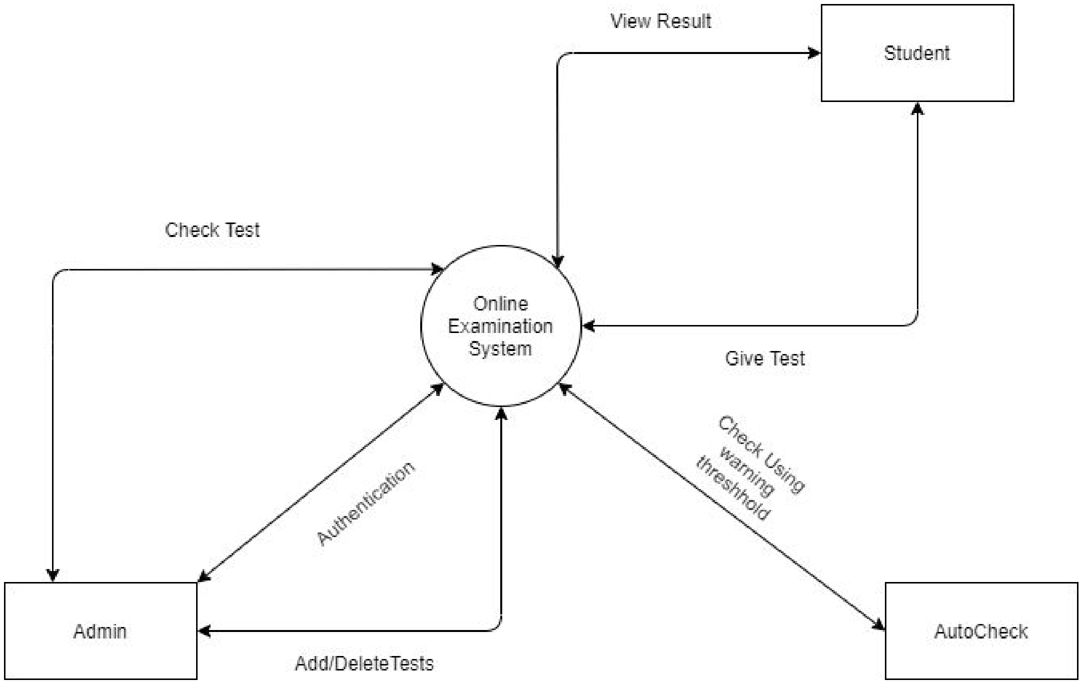
Use Case Diagram:



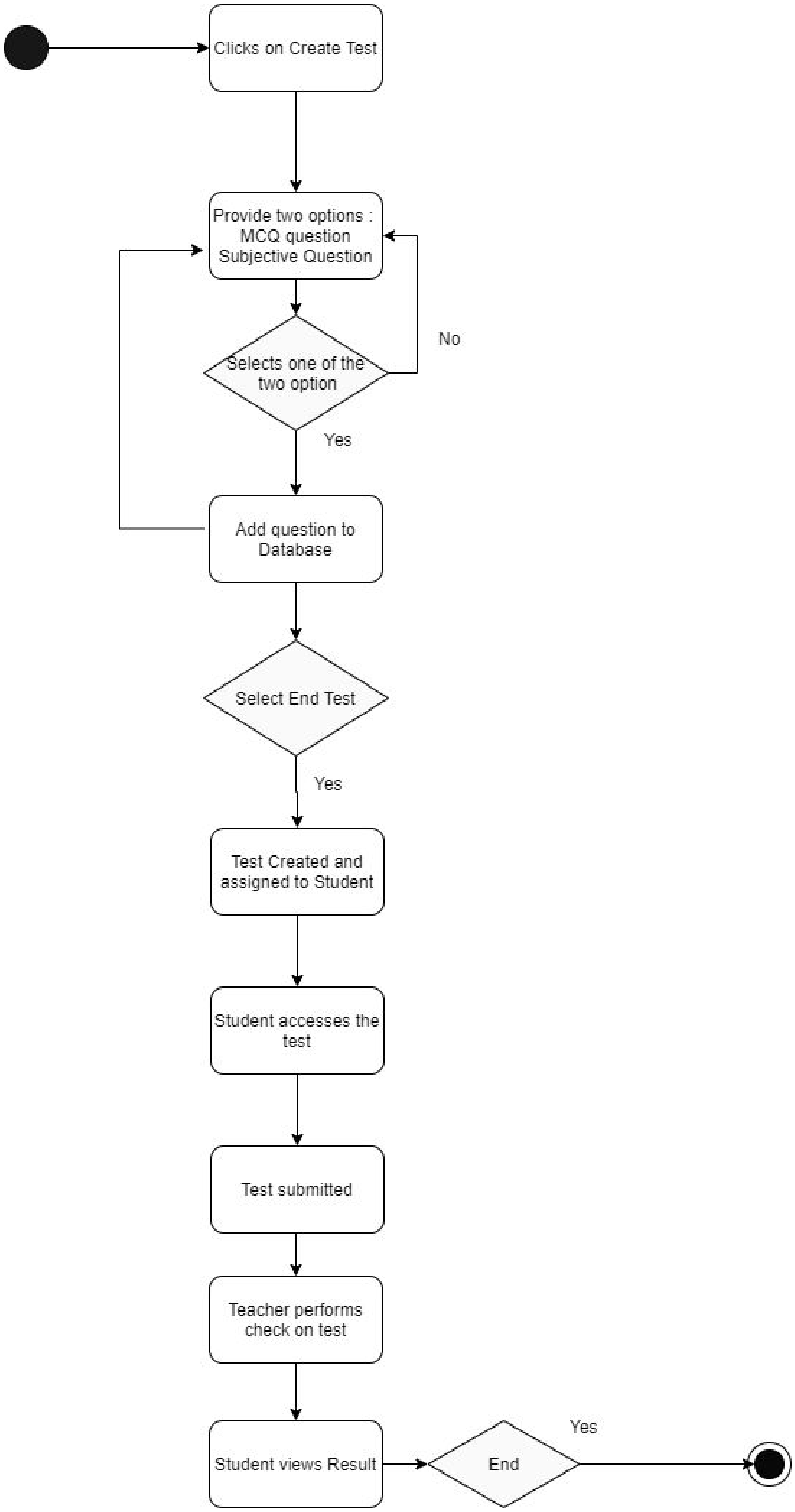
Entity-Relationship Diagram:



Data Flow Diagram:

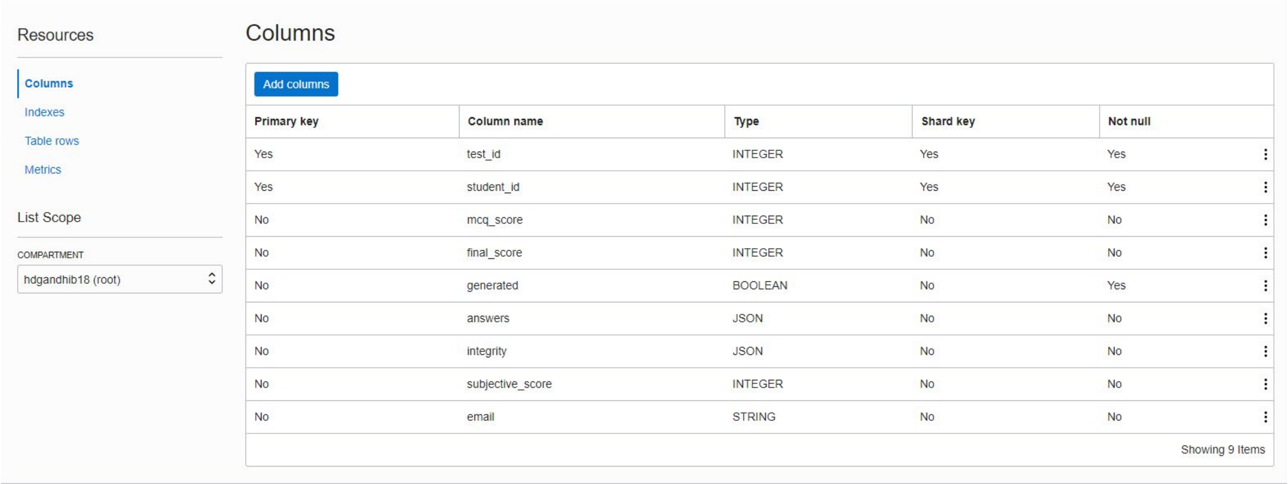


Activity Diagram:


Clicks on Create Test
Provide two options : MCQ question Subjective Question
Selects one of the two option
No
Yes
Add question to Database
Select End Test
Yes
Test Created and assigned to Student
Student accesses the test
Test submitted
Teacher performs check on test
Student views Result
End
Yes

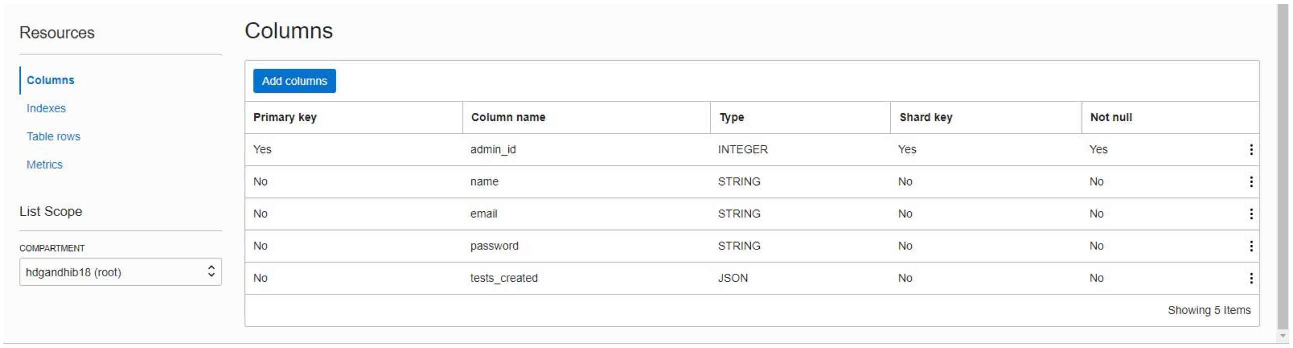
3.5 Database Schema

Result Table:



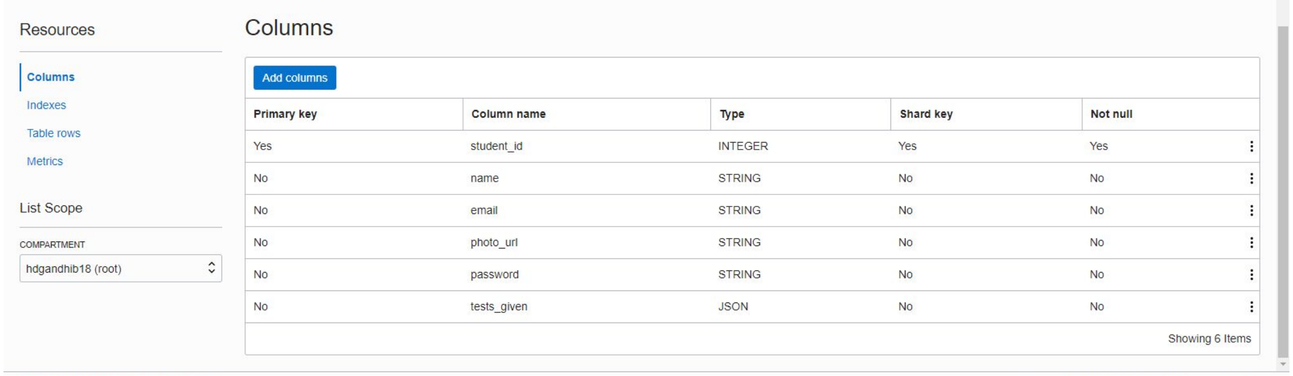
Primary key	Column name	Type	Shard key	Not null
Yes	test_id	INTEGER	Yes	Yes
Yes	student_id	INTEGER	Yes	Yes
No	mcq_score	INTEGER	No	No
No	final_score	INTEGER	No	No
No	generated	BOOLEAN	No	Yes
No	answers	JSON	No	No
No	integrity	JSON	No	No
No	subjective_score	INTEGER	No	No
No	email	STRING	No	No

Admin Table:



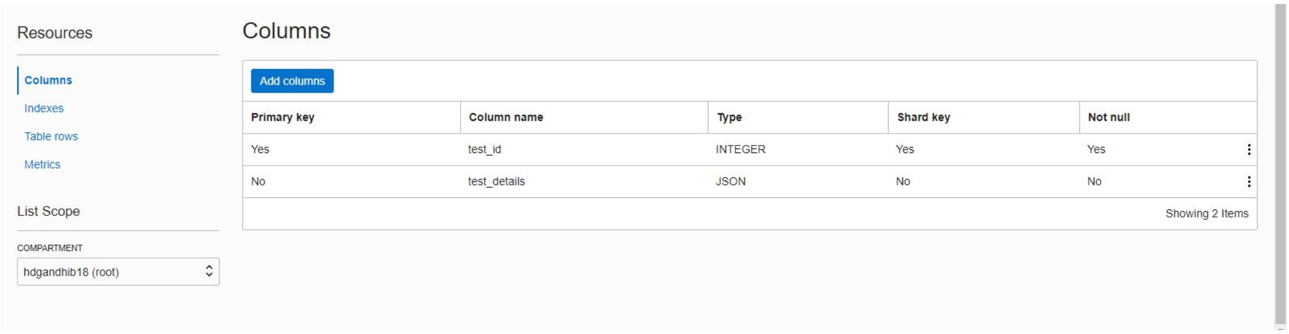
Primary key	Column name	Type	Shard key	Not null
Yes	admin_id	INTEGER	Yes	Yes
No	name	STRING	No	No
No	email	STRING	No	No
No	password	STRING	No	No
No	tests_created	JSON	No	No

Student Table:



Primary key	Column name	Type	Shard key	Not null
Yes	student_id	INTEGER	Yes	Yes
No	name	STRING	No	No
No	email	STRING	No	No
No	photo_url	STRING	No	No
No	password	STRING	No	No
No	tests_given	JSON	No	No

Test Table:



Primary key	Column name	Type	Shard key	Not null
Yes	test_id	INTEGER	Yes	Yes
No	test_details	JSON	No	No

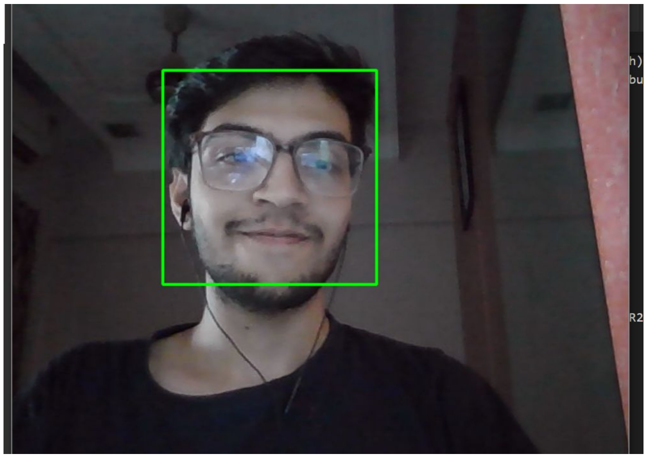
Showing 2 Items

4. System Features

The functional requirements for the product are described by system features, including all the major services provided by the product. Below mentioned are the innovative features that we have integrated into our project.

4.1 Face detection / Verification

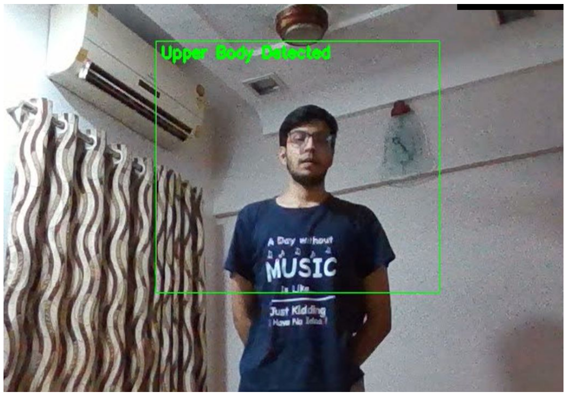
Dataset: Labeled Faces in the Wild
Implemented Algo: Haar-Cascade Classifier
Accuracy : 95%

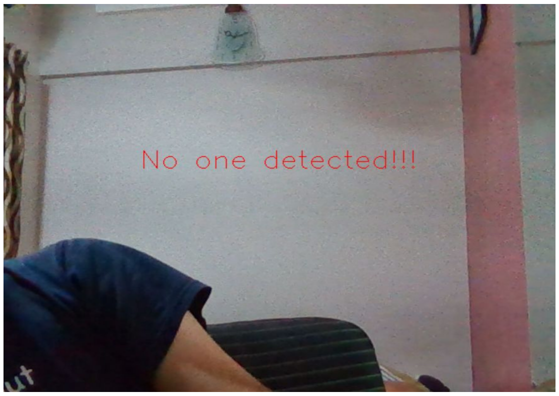


4.2 Anomaly Detection Services

Human Body Detection

Dataset & Accuracy: INRIA Person Dataset
Implemented Algo**:** HOG Descriptor & SVM

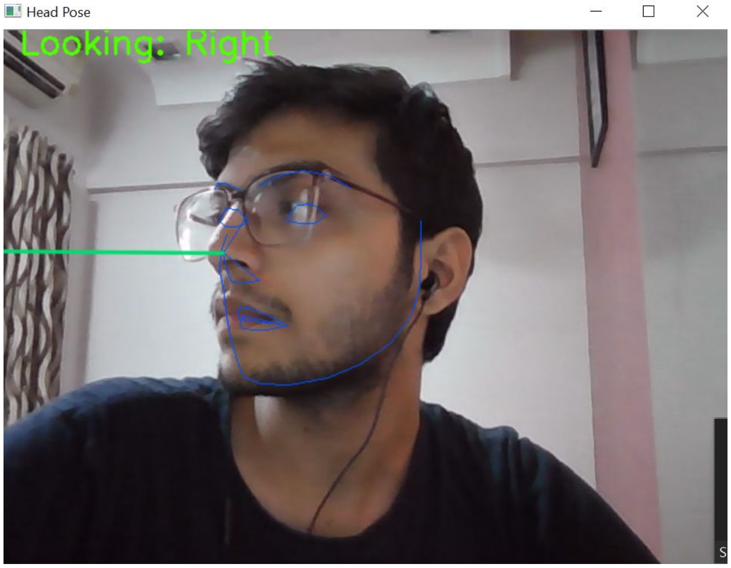


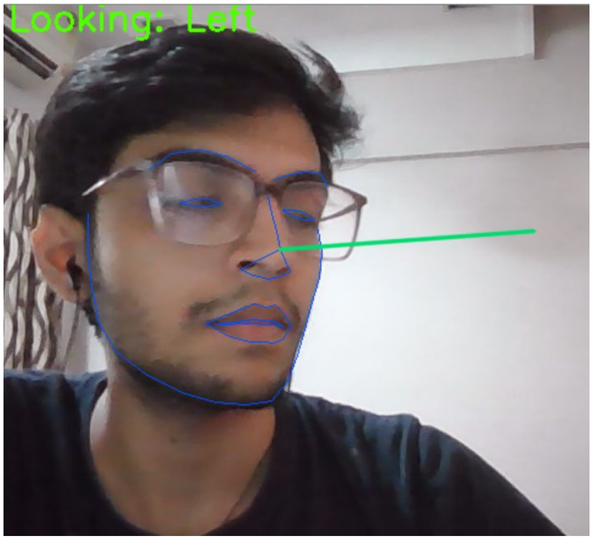


Face Landmark Detection

Dataset: Face Images with Marked Landmark Points

Implemented Algo: HOG + Linear SVM

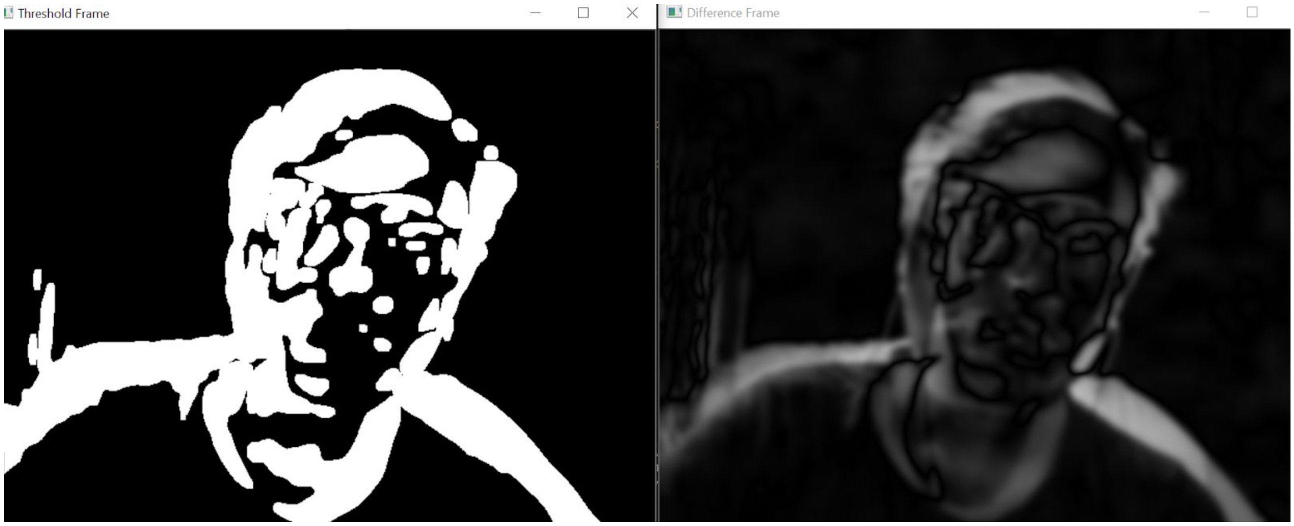




Motion Detection

Dataset: UCI Machine Learning Repository

Implemented Algo: Background Subtraction



4.3 Plagiarism Checker

One way we might go about detecting plagiarism is by comparing two passages and basing their decision off of how similar they appear. Before jumping to solutions, we must first understand how we define similarity and ultimately how it relates to plagiarism. We might decide that two texts are plagiarized if a large percentage of words in both passages are the same or if a set of words occur in the same order. This is exactly how we will tackle the plagiarism problem; by computing the similarity of two text answers.

This can be achieved by **Edit distance algorithm** or **containment algorithm**.

4.4 Key-Board monitoring

Using a software plugin we can check if the user is trying to copy and paste text on the test screen. For each of these actions specific warnings will be generated. These warnings will have a specific amount of threshold in place. If this threshold is achieved the test will automatically be closed.

4.5 Result Calculation

Validity Checks: JavaScript provides validity checks for various fields in the forms.

Sequencing Information: All the information regarding exam details, student list, question details, display of result should be handled sequentially that is data should be stored only in a particular sequence to avoid any inconvenience

Error Handling: If any of the validations or sequencing flows does not hold true then appropriate error messages will be prompted to the user for doing the needful.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

System should have:

- Low latency
- High throughput

5.2 Safety Requirements

The Cloud database may get crashed at any certain time due to virus or operating system failure. Therefore taking backup of the data will be necessary.

5.3 Security Requirements

Some of the factors that are identified to protect the software from accidental or malicious access, use, modification below. Keep specific log or history data sets

- Assign certain functions to different modules

5.4 Software System Attributes

Security: Only authorized users will be able to access the website by entering the correct login name and corresponding password.

Maintainability: The website can be maintained in the present or future. It will be easy to incorporate new requirements in the individual modules.

Portability: As the website is online so will be easily portable on various systems.

The website will be also easily portable on any windows based system that has MS ACCESS installed.