On Screen Marking System

(OSM)

A Project Report
Submitted in partial fulfillment of the
Requirements for the award of the Degree of

BACHELOR OF SCIENCE

(INFORMATON TECHNOLOGY)

By

Devangi Hitesh Parmar
Seat Number : 536

Under the esteemed guidance of

Ms. Vishakha Bagwe

Assistant Professor

NAGINDAS KHANDWALA COLLEGE(Autonomous)

(Affiliated to University of Mumbai)

MUMBAI, 400 064

MAHARASHTRA

2022-23

(**Original Copy of the Approved Proforma of the Project Proposal)**

PNR **No.: ……………………**   Roll no**: <u>536</u>**

1. Name of the Student: Devangi Hitesh Parmar

2. Title of the Project: On Screen Marking System(OSM)

3. Name of the Guide: Ms. Vishakha Bagwe

4. Teaching/Industry experience of the Guide: 6 years

5. Is this your first submission?

Signature of the Student: Signature of the Guide:

Date: …………………   Date: …………………….

Signature of the Coordinator:

Date: …………………

NAGINDAS KHANDWALA COLLEGE(Autonomous)

(Affiliated to University of Mumbai)

MUMBAI, 400 064

MAHARASHTRA

DEPARTMENT OF INFORMATION TECHNOLOGY

CERTIFICATE

This is to certify that the project titled, **"On Screen Marking System (OSM)"**, is bonafied work of **DEVANGI HITESH PARMAR** bearing Seat. No: **536** submitted in partial fulfillment of the requirements for the award of degree of BACHELOR OF SCIENCE in INFORMATION TECHONOLOGY from University of Mumbai.

Internal Guide

Coordinator

External Examiner

Date:

College Seal

Abstract

This report is about our project On-Screen Marking which provides online evaluation of the scanned answer scripts ‘anytime, anywhere’. The elementary business rules of OSM ensure that each answer is evaluated along with diagrams and graphs, answer keys to do the evaluation, easy re-evaluation process, annotations to provide remarks for each answer, parallel evaluation of an answer script by multiple evaluators and most importantly multiple-level of evaluation to ensure that there’s no biased evaluation. The best part is that the evaluation can be done from a laptop, mobile or tablet as the app supports all devices on cross platforms such as Android and Windows.

ACKNOWLEDGEMENT

I want to express my gratitude and appreciation to all those who gave me the possibility to complete this project. A special thanks to my parents and my brother for providing me with my needs and their stimulating support.

I am very thankful to **Dr. (Mrs.) Ancy Jose (Director) and Prof. Dr. Moushumi Datta (Principal)** of Nagindas Khandwala College for their kind co - operation in the completion of my project.

I also Thanks **Dr. Sindhu.P.M.** Head of Department of our section in college for supporting me.

I take this opportunity to express my deep sense of gratitude towards my project guide **Ms. Vishakha Bagwe** for her help and letting me work as I wished without putting pressure on me.

I also thankful to our other staff of our department for facilitating me with the necessary requirements and support and co - operation. My project experience was satisfying, fulfilling acknowledge filled lastly, I would like to thank my college Nagindas Khandwala college of commerce, arts and science (autonomous) for providing me with proper ambience and supplying me with the right amenities that has helped me complete this project on time and satisfyingly.

DECLARATION

I hereby declare that the project entitled, "**On Screen Marking System (OSM)**" done at **Nagindas Khandwala College**, has not been in any case duplicated to submit to any other university for the award of any degree. To the best of my knowledge other than me, no one has submitted to any other university.

The project is done in partial fulfillment of the requirements for the award of degree of **BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY)** to be submitted as final semester project as part of our curriculum.

DEVANGI PARMAR

TABLE OF CONTENTS

List of Tables

List of Figures

CHAPTER 1
INTRODUCTION

1.1 Background

On-screen marking is a method of marking a candidate's response that has been produced on paper and presented to an examiner for marking on-screen using a PC/Laptop. Evaluation of descriptive answer script with scalable and robust Document Management based On-Screen Marking System, to help educational institutions & commissions increase efficiency and reduce costs around assessing physical examination documents. The System effectively automates and streamlines handling of physical documents; enabling educational institutions to transform their traditional paper based examination evaluation systems.

The process of conducting and evaluating paper-based exams is very complex and challenging. As the number of candidates appearing for these examinations is constantly increasing day by day, there is a large volume of paper scripts to be managed, requiring extensive logistics for their distribution to the evaluation centers, spread across geographies. And keeping this at a secure place engaged lots of space.

On-Screen Marking (OSM) system provides an extensive platform for evaluating pen and paper based examinations on computer screens. On-Screen Marking system completely automates the marking and scoring of descriptive answer scripts into an accurate, simple, secure and efficient process.

1.2 Objectives

- An increase in the speed of marking
- A reduction in the risks associated with transporting exam scripts between centres and examiners
- Real-time monitoring of marking progress & quality
- An improvement in the reliability and quality of marking
- Secure storage and management of scanned answer scripts
- Elimination of errors related to incorrect tabulation and recording of marks

1.3 Applicability

- There is no location constraint in such a case. Any examiner/ answer sheet checker or Moderator can verify/ evaluate answer sheets sitting at their location.
- Live tracking of number of answer scripts evaluated or pending
- Simultaneous evaluations by faculties
- Support to give individual remarks during evaluation
- Swift and easy re-evaluation process

CHAPTER 2

GAP ANALYSIS/ DRAWBACK OF EXISTING SYSTEM

1. Lack of Security: Additional security measures have to be manually taken by an assessment body to maintain the secrecy of the evaluation. This accounts for extra time, manpower and management.

2. Network Connectivity: Need of Internet Connection is required to perform all the tasks regarding evaluation.

3. Handling of re-evaluation: In the traditional method of evaluation, it is a tedious job of getting out the answer scripts and scanning it in order to issue it to the candidate for re-evaluation. The entire deal with handling the answer scripts is very risky and time-consuming.

4. Longer turn around time: The traditional evaluation process spans over months and there are also chances of delays in evaluation for various reasons. This slows down the result publication and may result in serious problems for the candidates who may be waiting to share the result with companies or higher education institutes.

CHAPTER 3

REQUIREMENTS AND ANALYSIS

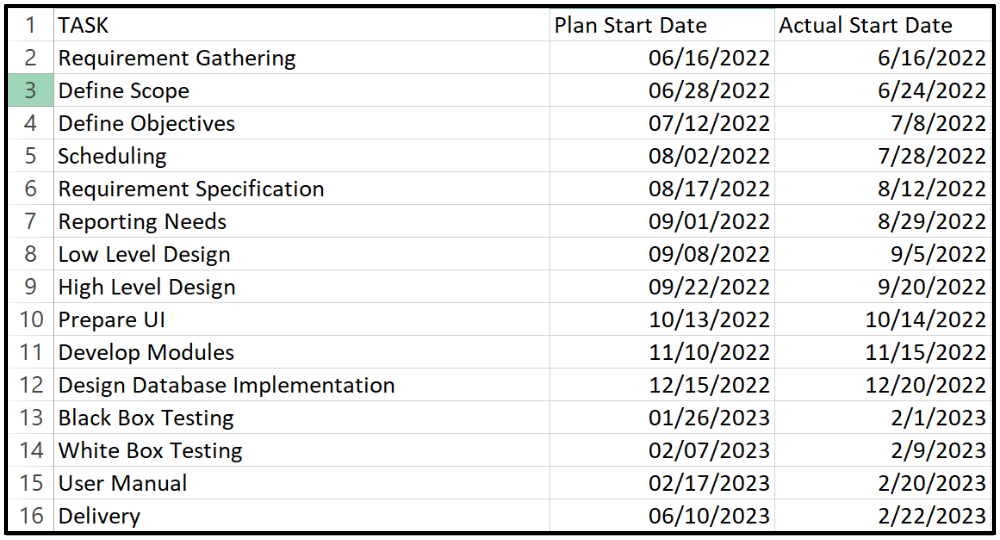
3.1 Problem Definition

1. Onscreen Marking System is useful to evaluate physical copies of the answer sheets in digital format.
2. It helps removing location and physical answer sheet handling constraint for the examiners, moderators and result processing authority.
3. Physical written answer sheet is scanned and uploaded to cloud based software system.
4. The process is completely manual and even one small mistake in marks assignment or calculation can affect a candidate's performance.
5. Model : We have used Agile model as focus on process adaptability and customer satisfaction by rapid delivery of working software product. It break the product into small incremental builds. These builds are provided in iterations.

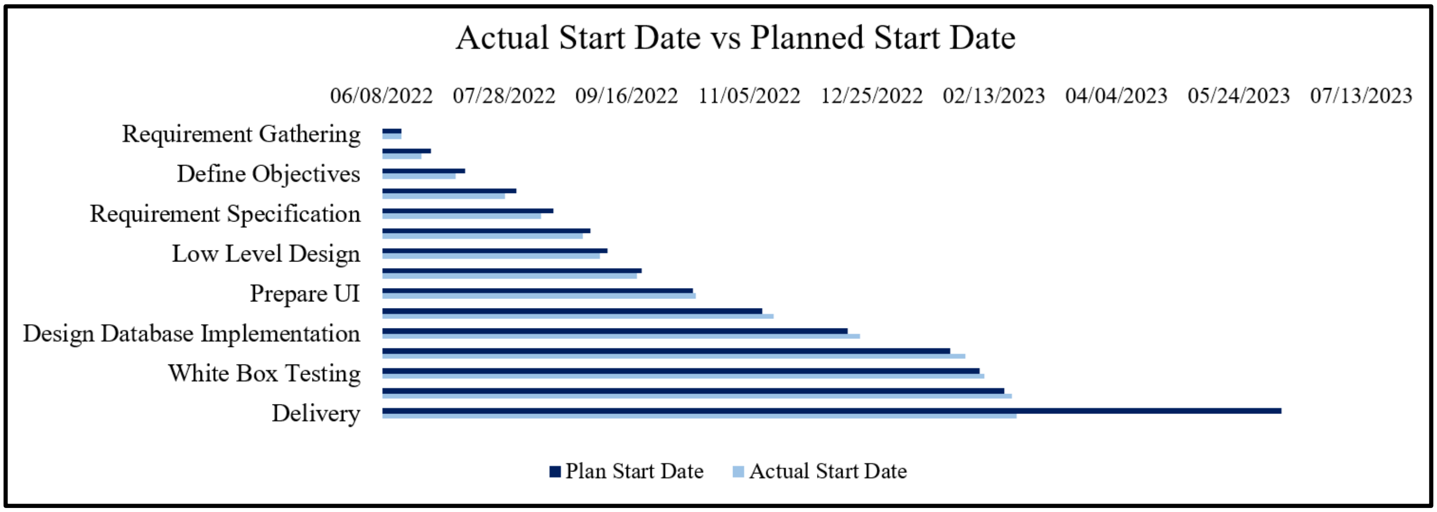
3.2 Requirements Specification

1. Our clients have gained more than 60% savings in the cost and able to process results in less than 8 days of time.
2. The cost for managing storage and handling of physical answer sheets is very high. Entire result processing may get delayed due to it.
3. Tampering of scores can happen for any reason. In the traditional evaluation process, there's only one course of evaluation and no parallel multiple evaluations. Thus, the tampering goes unnoticed.

3.3 Planning and Scheduling



1	TASK	Plan Start Date	Actual Start Date
2	Requirement Gathering	06/16/2022	6/16/2022
3	Define Scope	06/28/2022	6/24/2022
4	Define Objectives	07/12/2022	7/8/2022
5	Scheduling	08/02/2022	7/28/2022
6	Requirement Specification	08/17/2022	8/12/2022
7	Reporting Needs	09/01/2022	8/29/2022
8	Low Level Design	09/08/2022	9/5/2022
9	High Level Design	09/22/2022	9/20/2022
10	Prepare UI	10/13/2022	10/14/2022
11	Develop Modules	11/10/2022	11/15/2022
12	Design Database Implementation	12/15/2022	12/20/2022
13	Black Box Testing	01/26/2023	2/1/2023
14	White Box Testing	02/07/2023	2/9/2023
15	User Manual	02/17/2023	2/20/2023
16	Delivery	06/10/2023	2/22/2023



3.4 Software and Hardware Requirements

Hardware Requirements

- Processor : Minimum 1 GHz; Recommended 2Ghz or more
- Hard drive : Minimum 32GB; Recommended 64GB or more
- Memory(RAM) : Minimum 1GB; Recommended 4GB or above

Software Requirements

- Frontend : HTML, CSS, Javascript, jQuery
- Backend : Python, Mysql
- MS word
- Web browser : Google chrome, Microsoft Edge

3.5 Preliminary Product Description

- Timely Updates are sent to evaluators
- Controller of Examination (COE) can monitor status in real time
- Marking schemes can be created from web UI
- Counting and other human errors during evaluation are unavoidable
- No way to track overall status of evaluation in real-time
- No way to assess evaluator performance (time taken etc.)
- Evaluation is possible only during regular working hours (no exceptions)

CHAPTER 4

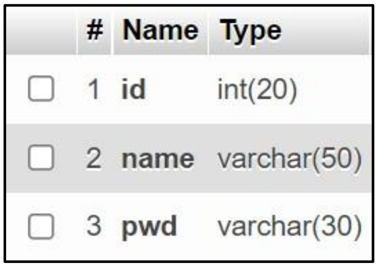
SYSTEM DESIGN

4.1 Basic Modules

- Admin login : A login system which helps admin to access system
- Teacher login : a login system provides login for all the teachers to access papers and other details.
- Admin Dashboard : In the dashboard Admin can perform following tasks
  - It can add teachers respective to their subject and course
  - It can store update of teacher's check-in checkout history
  - It assigns papers to the respective teachers
- Teacher dashboard: helps teachers to stay updated with all the paper details and its checking

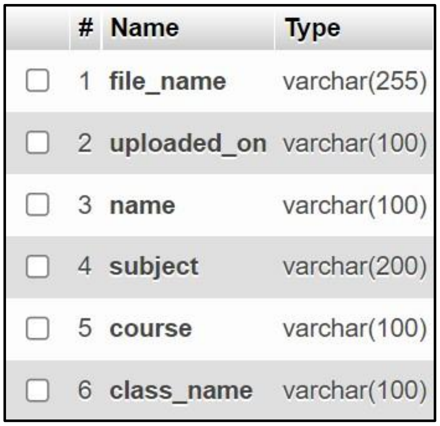
4.2 Schema Design

1. Admin



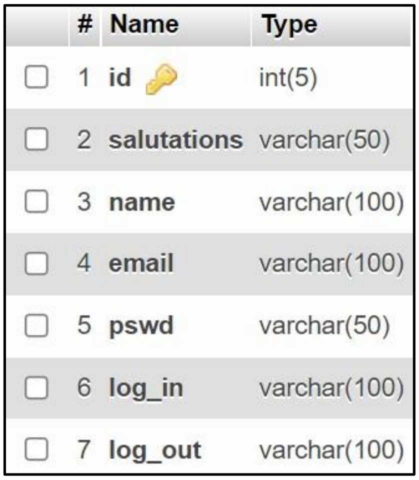
	#	Name	Type
□	1	**id**	int(20)
□	2	**name**	varchar(50)
□	3	**pwd**	varchar(30)

2. Assign



	#	Name	Type
□	1	**file_name**	varchar(255)
□	2	**uploaded_on**	varchar(100)
□	3	**name**	varchar(100)
□	4	**subject**	varchar(200)
□	5	**course**	varchar(100)
□	6	**class_name**	varchar(100)

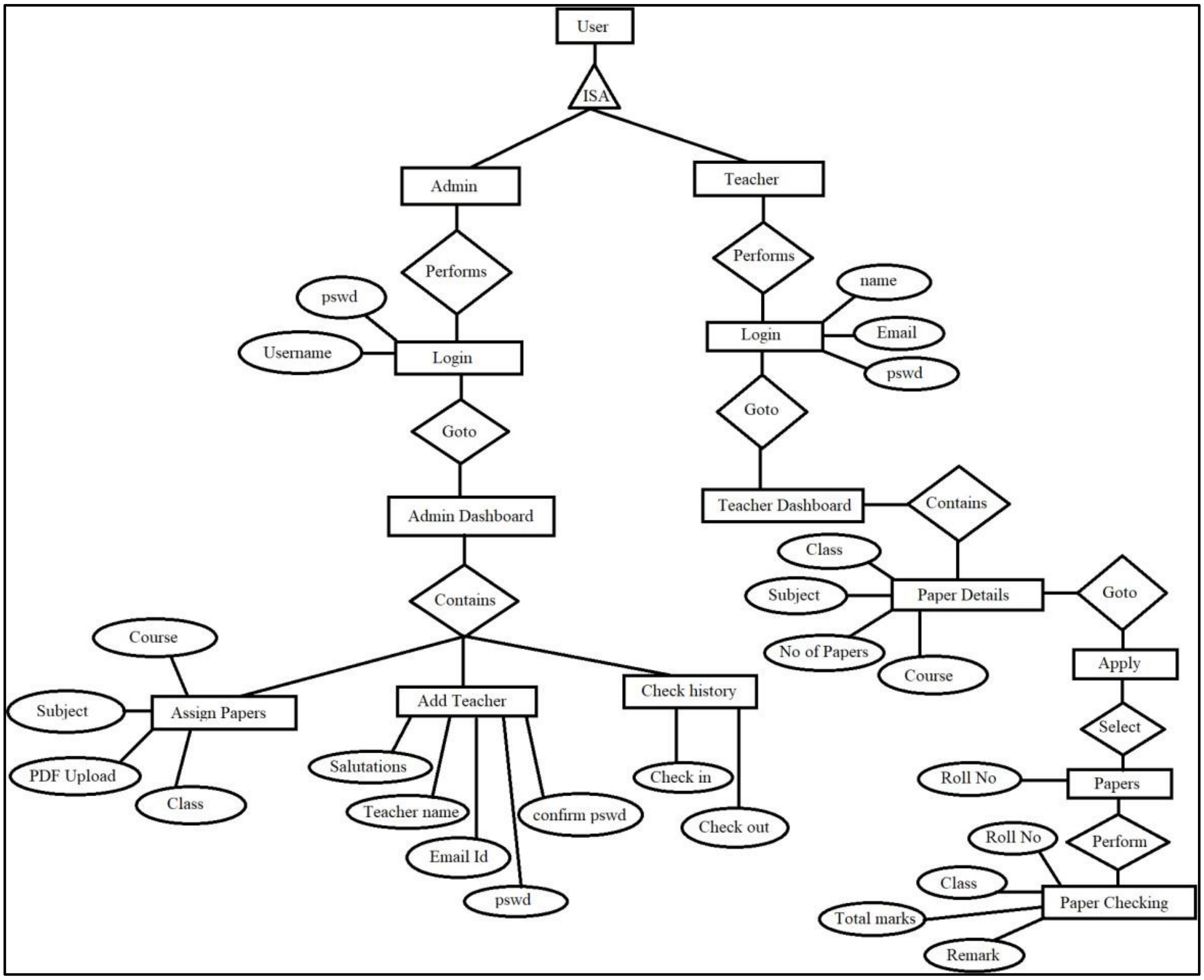
3. Teacher



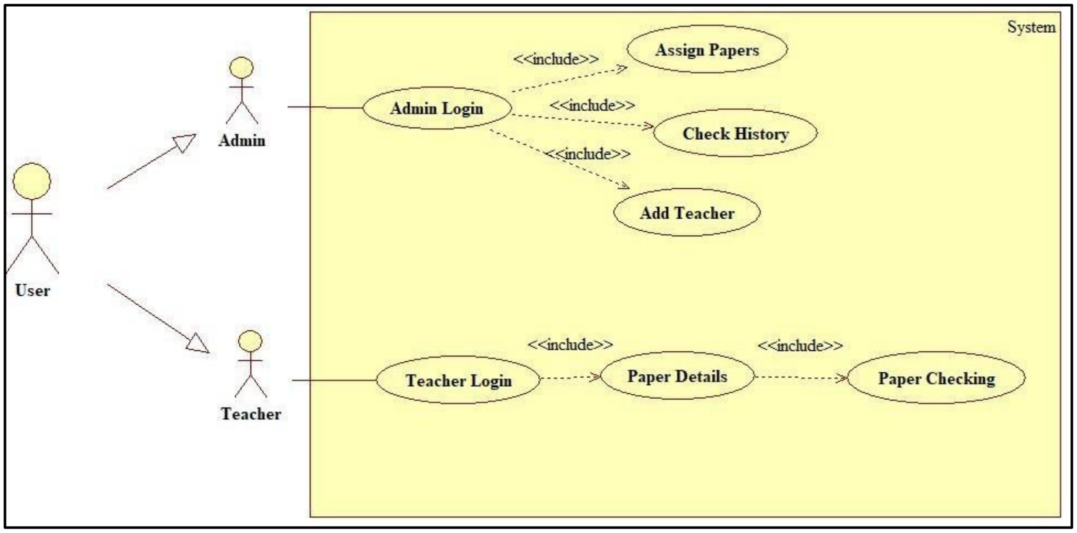
	#	Name	Type
□	1	id 🔑	int(5)
□	2	salutations	varchar(50)
□	3	name	varchar(100)
□	4	email	varchar(100)
□	5	pswd	varchar(50)
□	6	log_in	varchar(100)
□	7	log_out	varchar(100)

4.3 UML Diagrams

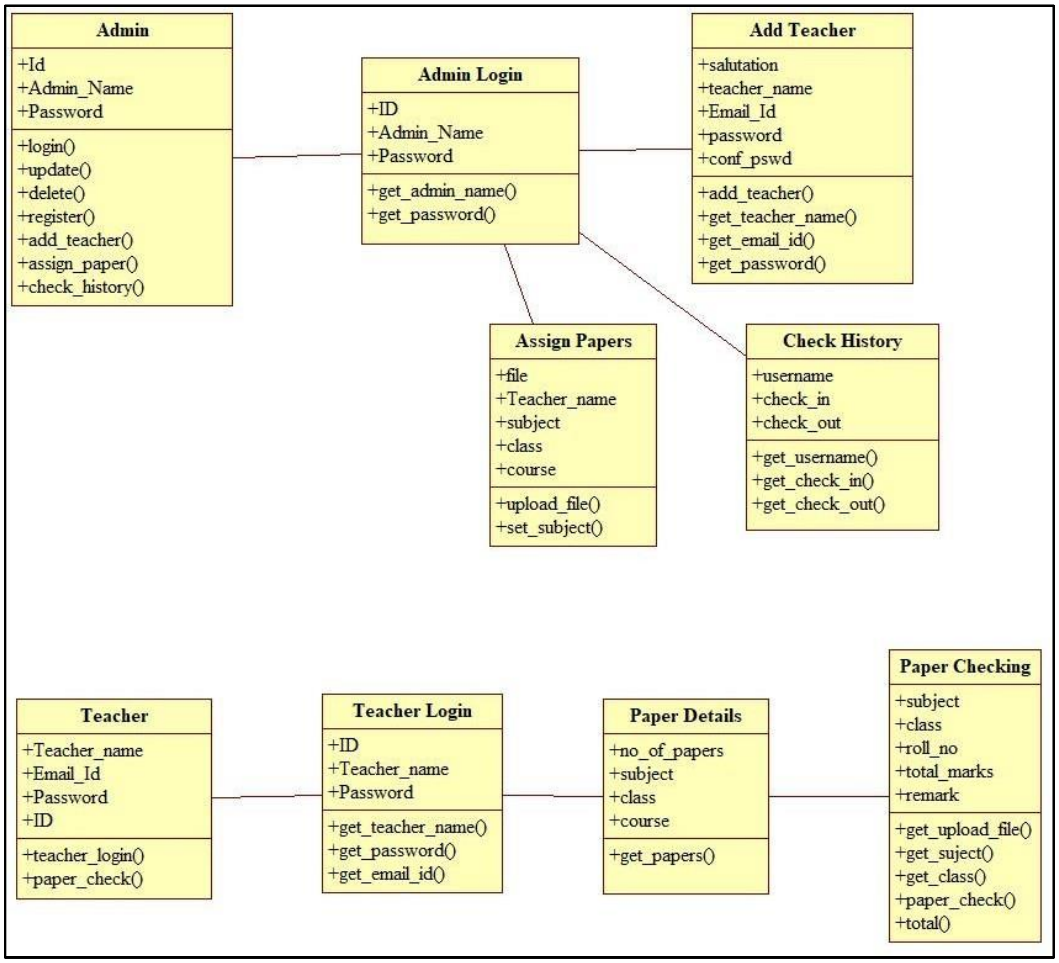
1. ER Diagram



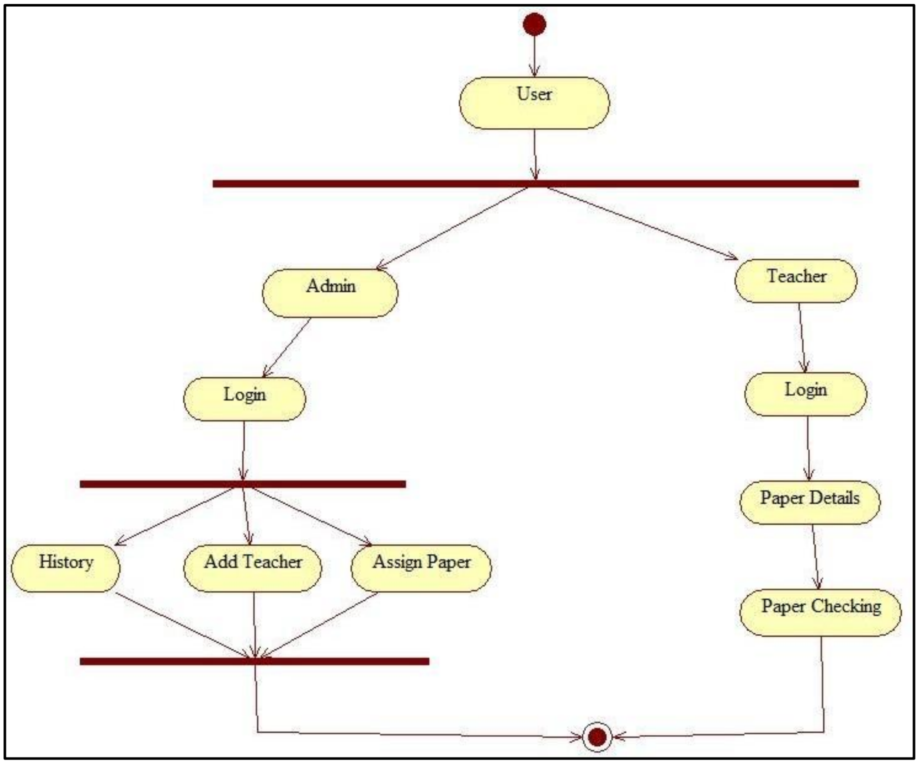
2. Use Case Diagram



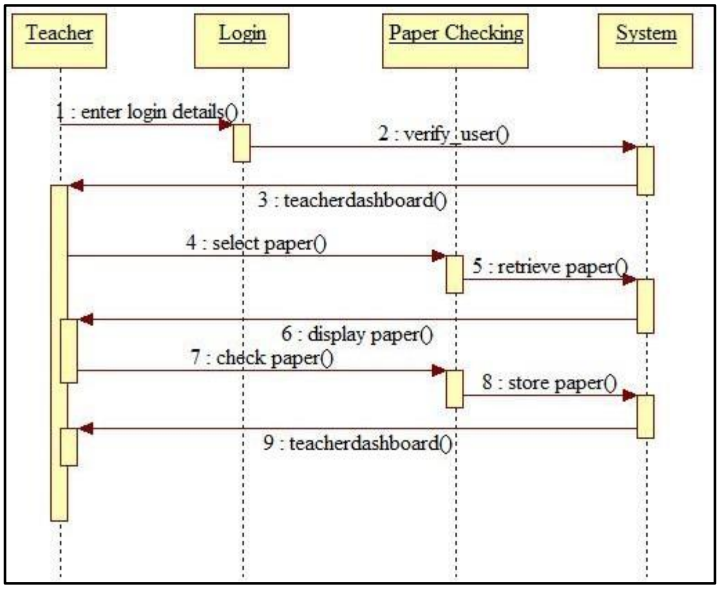
3. Class Diagram

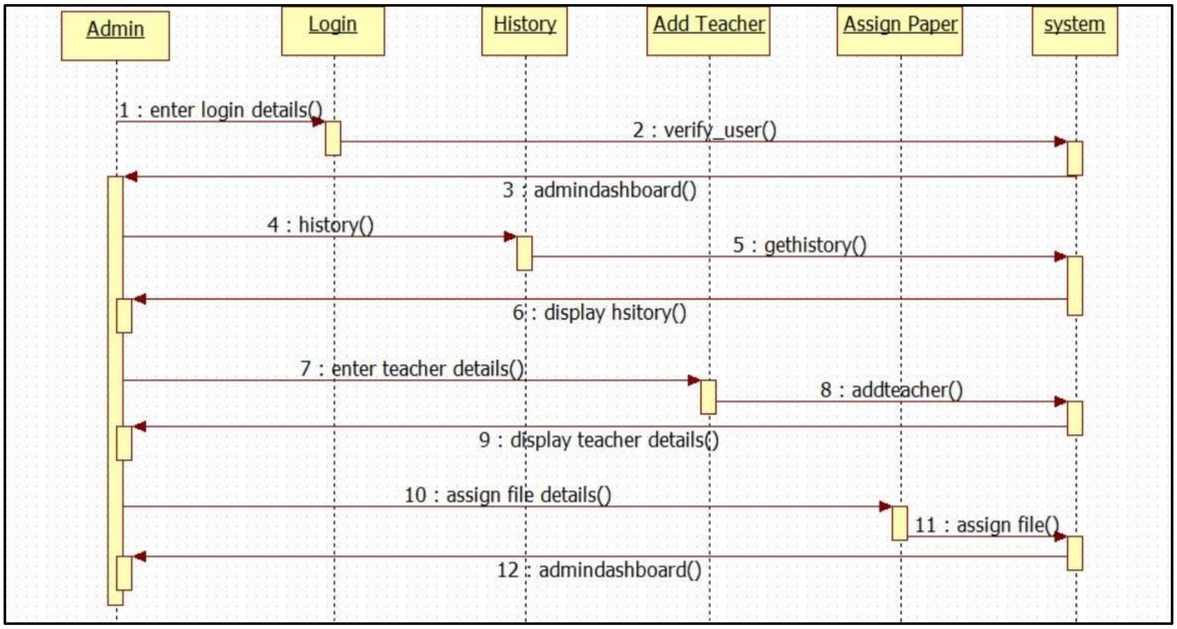


4. Activity Diagram



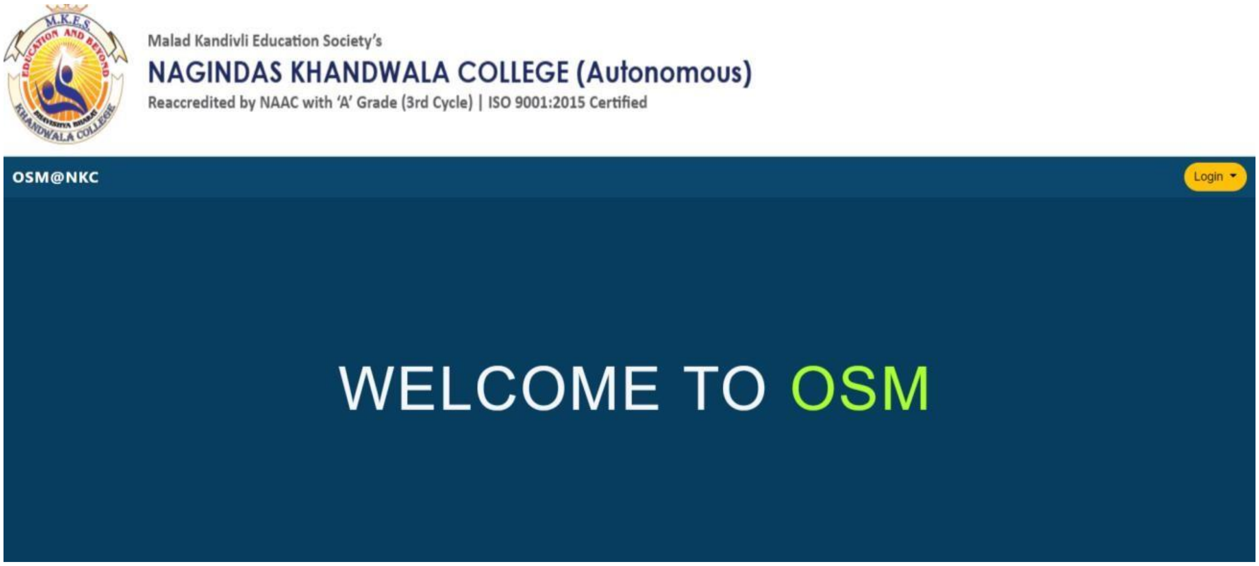
5. Sequence Diagram





4.4 User interface design

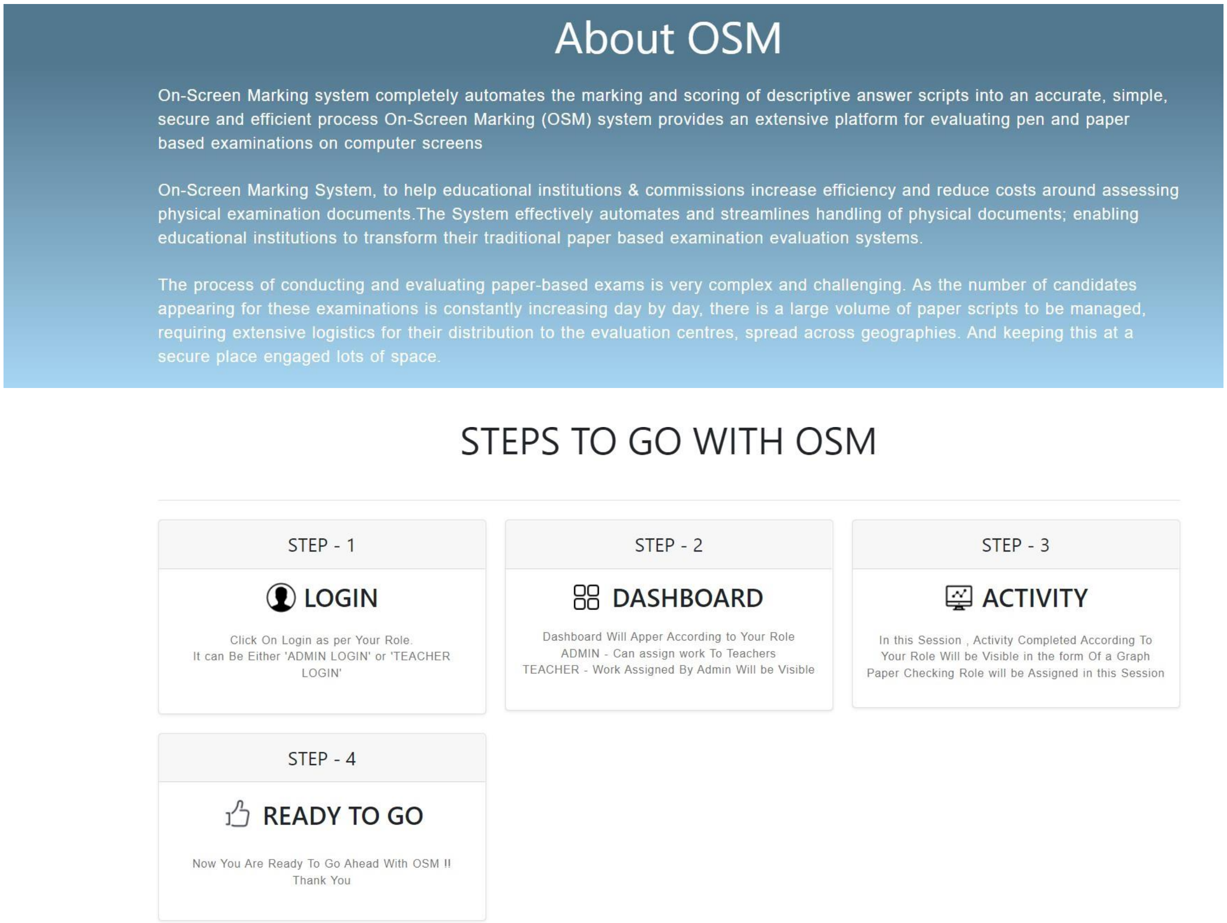
1. Home Page


Malad Kandivli Education Society's

NAGINDAS KHANDWALA COLLEGE (Autonomous)

Reaccredited by NAAC with 'A' Grade (3rd Cycle) | ISO 9001:2015 Certified

WELCOME TO OSM



About OSM

On-Screen Marking system completely automates the marking and scoring of descriptive answer scripts into an accurate, simple, secure and efficient process On-Screen Marking (OSM) system provides an extensive platform for evaluating pen and paper based examinations on computer screens

On-Screen Marking System, to help educational institutions & commissions increase efficiency and reduce costs around assessing physical examination documents.The System effectively automates and streamlines handling of physical documents; enabling educational institutions to transform their traditional paper based examination evaluation systems.

The process of conducting and evaluating paper-based exams is very complex and challenging. As the number of candidates appearing for these examinations is constantly increasing day by day, there is a large volume of paper scripts to be managed, requiring extensive logistics for their distribution to the evaluation centres, spread across geographies. And keeping this at a secure place engaged lots of space.

STEPS TO GO WITH OSM

STEP - 1

LOGIN

Click On Login as per Your Role.
It can Be Either 'ADMIN LOGIN' or 'TEACHER LOGIN'

STEP - 2

DASHBOARD

Dashboard Will Apper According to Your Role
ADMIN - Can assign work To Teachers
TEACHER - Work Assigned By Admin Will be Visible

STEP - 3

ACTIVITY

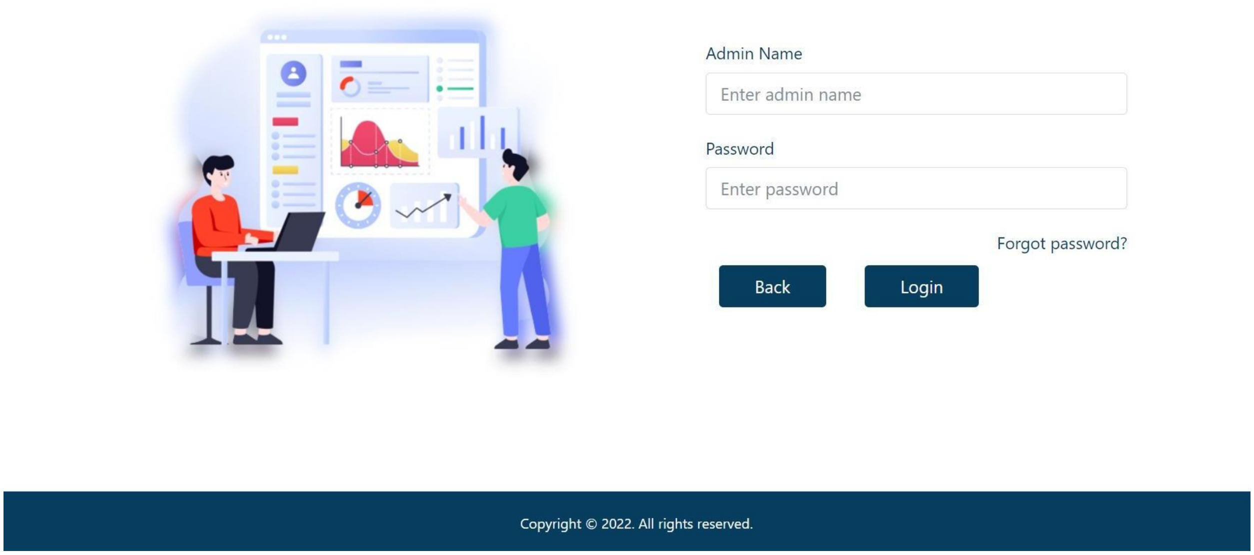
In this Session , Activity Completed According To Your Role Will be Visible in the form Of a Graph
Paper Checking Role will be Assigned in this Session

STEP - 4

READY TO GO

Now You Are Ready To Go Ahead With OSM !!
Thank You

2. Admin Login



Admin Name

Enter admin name

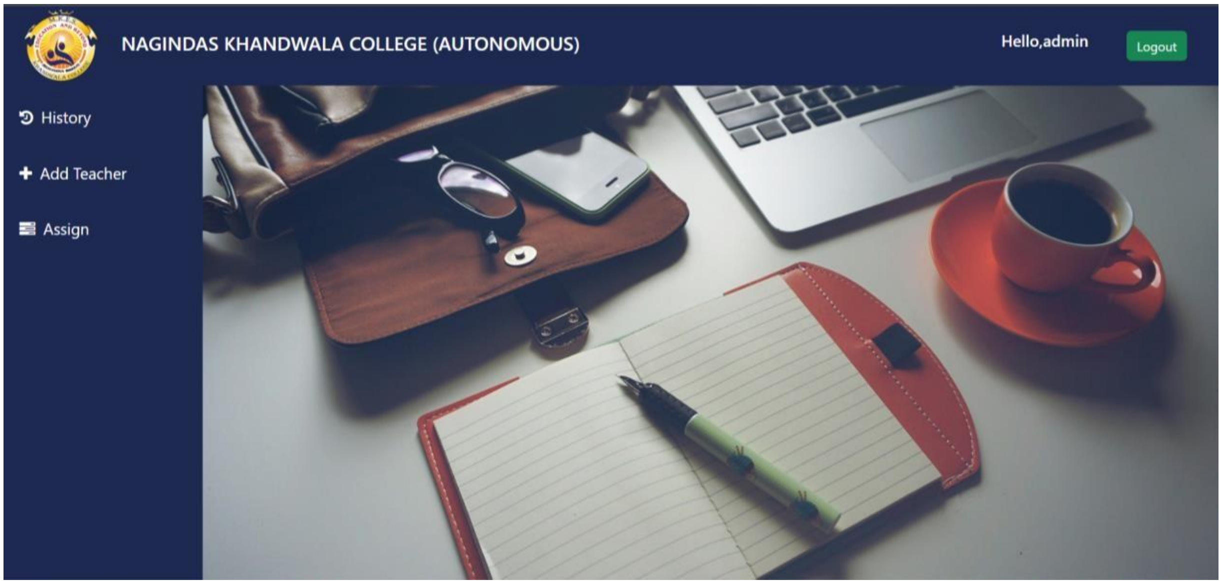
Password

Enter password

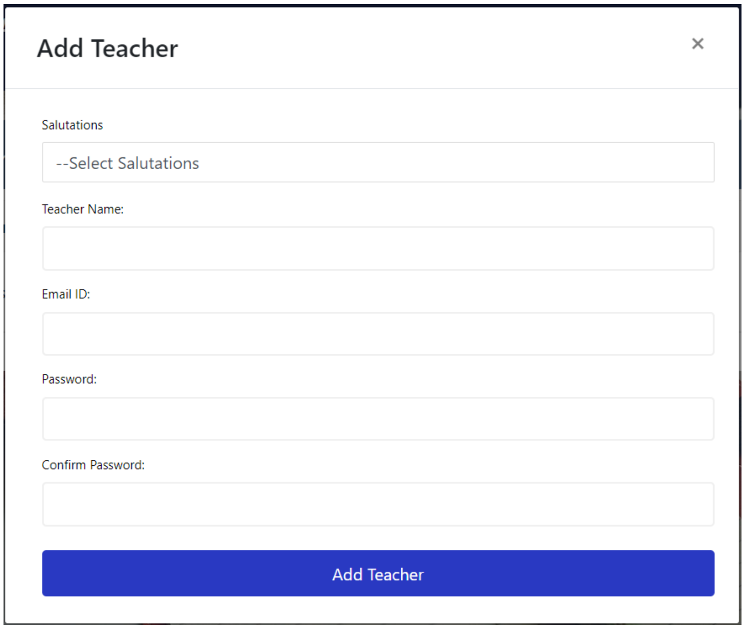
Forgot password?

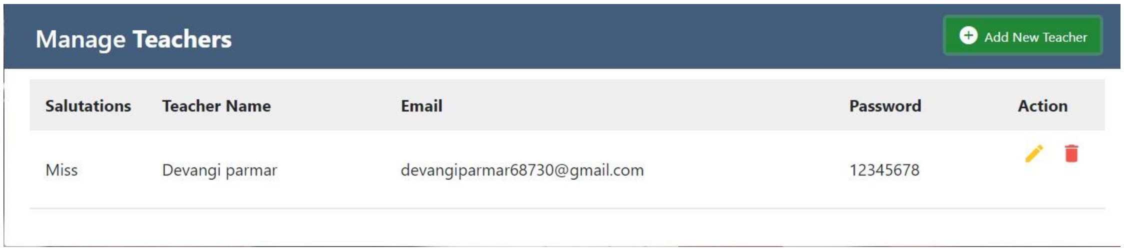
Copyright © 2022. All rights reserved.

3. Admin Dashboard



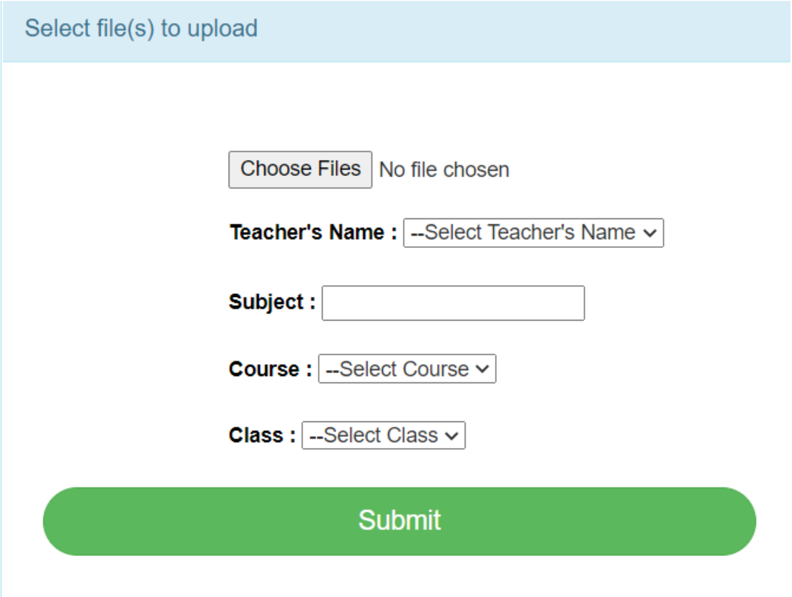
4. Add Teacher





Salutations	Teacher Name	Email	Password	Action
Miss	Devangi parmar	devangiparmar68730@gmail.com	12345678	✎ 🗑

5. Assign Papers



Select file(s) to upload

No file chosen

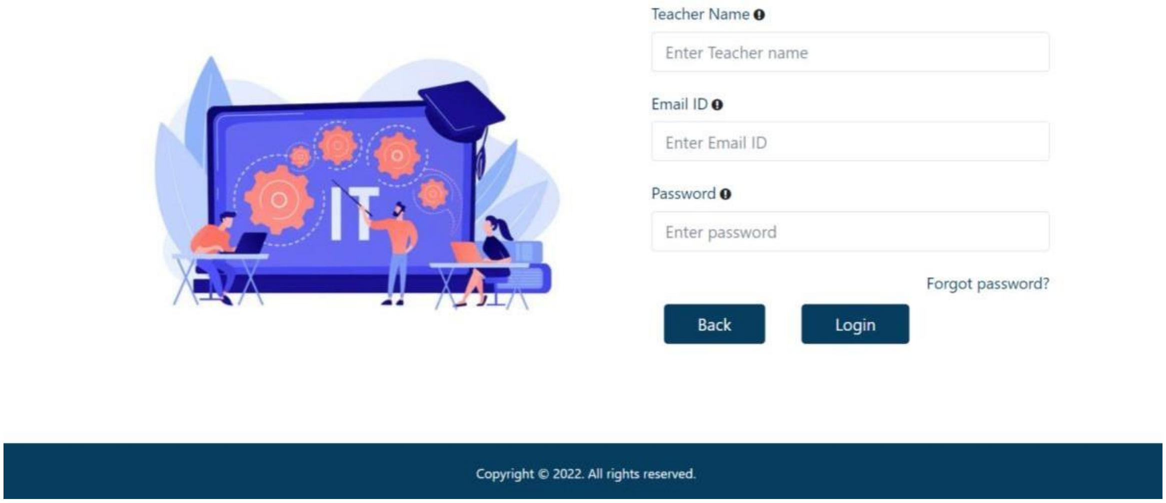
Teacher's Name : --Select Teacher's Name

Subject :

Course : --Select Course

Class : --Select Class

6. Teacher Login



Teacher Name

Enter Teacher name

Email ID

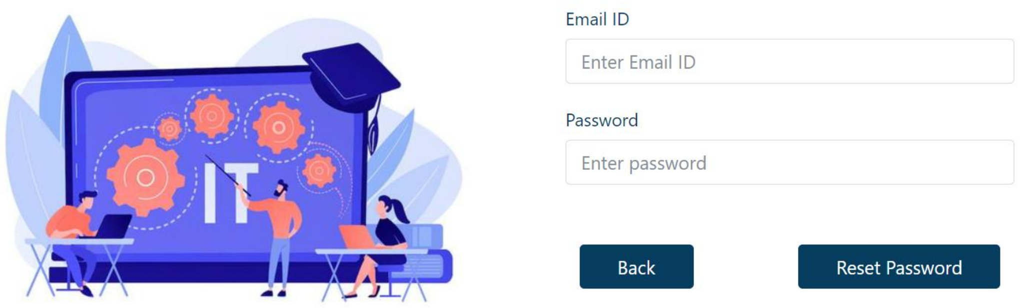
Enter Email ID

Password

Enter password

Copyright © 2022. All rights reserved.

7. Teacher Reset Password



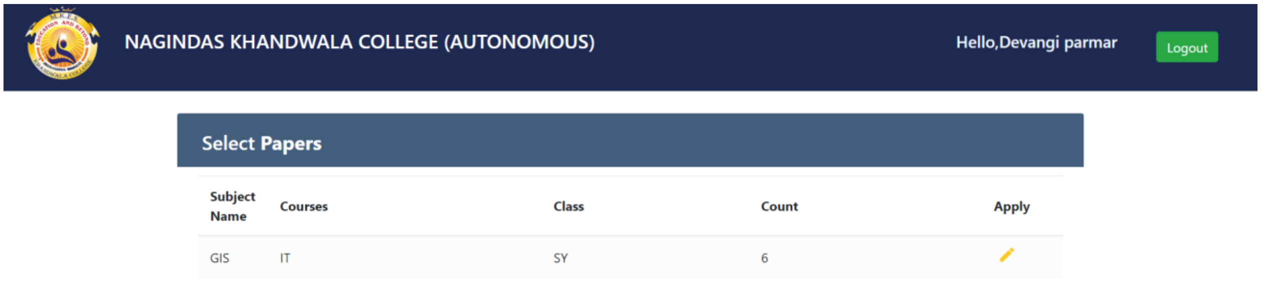
Email ID

Enter Email ID

Password

Enter password

8. Teacher Dashboard



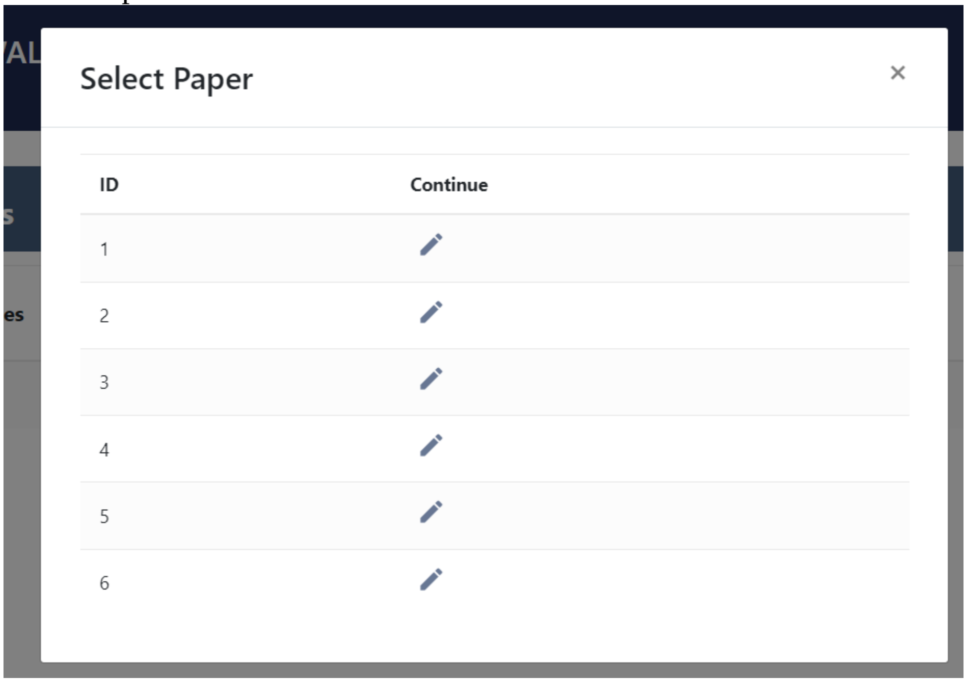
NAGINDAS KHANDWALA COLLEGE (AUTONOMOUS)

Hello,Devangi parmar

Select Papers

Subject Name	Courses	Class	Count	Apply
GIS	IT	SY	6	✎

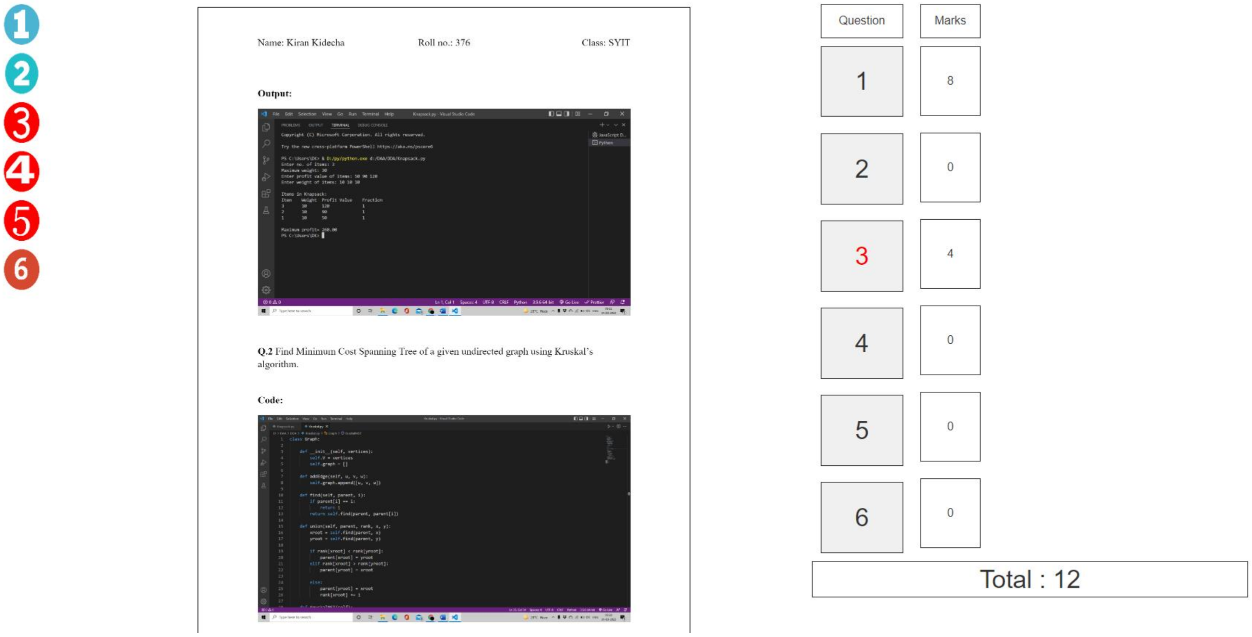
9. Select Paper



Select Paper

ID	Continue
1	✎
2	✎
3	✎
4	✎
5	✎
6	✎

10. Paper checking



Name: Kiran Kidecha Roll no.: 376 Class: SYIT

Output:

```
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\DK> & D:/py/python.exe d:/DAA/DDA/Knapsack.py
Enter no. of items: 3
Maximum weight: 30
Enter profit value of items: 50 90 120
Enter weight of items: 10 10 10

Items in Knapsack:
Item    Weight  Profit Value    Fraction
3       10      120             1
2       10      90              1
1       10      50              1

Maximum profit= 260.00
PS C:\Users\DK>
```

Q.2 Find Minimum Cost Spanning Tree of a given undirected graph using Kruskal's algorithm.

Code:

```python
class Graph:

    def __init__(self, vertices):
        self.V = vertices
        self.graph = []

    def addEdge(self, u, v, w):
        self.graph.append([u, v, w])

    def find(self, parent, i):
        if parent[i] == i:
            return i
        return self.find(parent, parent[i])

    def union(self, parent, rank, x, y):
        xroot = self.find(parent, x)
        yroot = self.find(parent, y)

        if rank[xroot] < rank[yroot]:
            parent[xroot] = yroot
        elif rank[xroot] > rank[yroot]:
            parent[yroot] = xroot

        else:
            parent[yroot] = xroot
            rank[xroot] += 1
```

Question	Marks
1	8
2	0
3	4
4	0
5	0
6	0

Total : 12

4.5 Security Issues

- SQL INJECTIONS
- BROKEN AUTHENTICATION & SESSION MANAGEMENT

CHAPTER 5

IMPLEMENTATION AND TESTING

5.1 Code (Place Core Segments)

app.py

```
from datetime import datetime
from flask import Flask, flash, render_template, request, redirect, url_for, session
from flask_mysqldb import MySQL, MySQLdb
import os
from werkzeug.utils import secure_filename
from flask_mail import Mail

app = Flask(_name_)
app.secret_key = 'admin1234'

app.config['MYSQL_HOST'] = 'localhost'
app.config['MYSQL_USER'] = 'root'
app.config['MYSQL_PASSWORD'] = ''
app.config['MYSQL_DB'] = 'osm'
mysql = MySQL(app)

app.config["MAIL_SERVER"]='smtp.gmail.com'
app.config["MAIL_PORT"] = 465
app.config["MAIL_USERNAME"] = 'osmnkc2022@gmail.com'
app.config['MAIL_PASSWORD'] = ''
app.config['MAIL_USE_TLS'] = False
app.config['MAIL_USE_SSL'] = True
```

```
mail = Mail(App)
UPLOAD_FOLDER = './static/upload'
app.config['UPLOAD_FOLDER'] = UPLOAD_FOLDER
app.config['MAX_CONTENT_LENGTH'] = 16 * 1024 * 1024
ALLOWED_EXTENSIONS = set(['pdf'])

@app.route('/')
def home():
    return render_template('home.html')


@app.route('/add/teacher')
def addteacher():
    cursor = mysql.connection.cursor()
    cursor.execute("SELECT * FROM teacher")
    data = cursor.fetchall()
    return render_template('addteacher.html', data=data)

@app.route('/register', methods=['POST', 'GET'])
def register():
    if request.method == 'GET':
        return "Login via the login Form"
    if request.method == 'POST':
        salutations = request.form['salutations']
        name = request.form['name']
        email = request.form['email']
        pswd = request.form['pswd']
        cursor = mysql.connection.cursor()
        cursor.execute(''' INSERT INTO teacher (salutations,name,email, pswd, log_in, log_out)
VALUES(%s,%s,%s,%s,%s,%s)''',(salutations,name,email,pswd,datetime.now(), ''))

        mysql.connection.commit()


        mail.send_message(

                    sender="osmnkc2022@gmail.com",

                    recipients = [email],

                    subject = "Login Details",

                    body = "Username : " + name + "\n" + "Password : " + pswd

                )

        flash("Teacher Added Successfully")

        return redirect('/add/teacher')


@app.route('/update', methods=["POST"])

def update():

    id = request.form['id']

    salutations = request.form['salutations']

    name = request.form['name']

    email = request.form['email']

    pswd = request.form['pswd']

    cursor = mysql.connection.cursor()

    cursor.execute("""UPDATE teacher SET salutations=%s,name=%s, email=%s, pswd=%s
WHERE id=%s""",(salutations,name, email, pswd, id))

    mysql.connection.commit()

    flash("Teacher updated Successfully")

    return redirect('/add/teacher')


@app.route('/delete/<id>')
```

```
def delete(id):
    id = request.form['id']
    cursor = mysql.connection.cursor()
    cursor.execute("""DELETE FROM teacher where id=%s""", (id,))
    mysql.connection.commit()
    flash("Teacher Deleted Successfully")
    return redirect('/add/teacher')


@app.route('/admin/login', methods=['GET', 'POST'])
def login():
    if request.method == 'POST' and 'admin' in request.form and 'pass' in request.form:
        username = request.form['admin']
        password = request.form['pass']
        cursor = mysql.connection.cursor(MySQLdb.cursors.DictCursor)
        cursor.execute('SELECT * FROM admin WHERE name = %s AND pwd = %s', (username,
password,))
        account = cursor.fetchone()
        if account:
            session['admin_loggedin'] = True
            session['id'] = account['id']
            session['username'] = account['name']
            return redirect('/admin/dashboard')
        else:
            return 'Incorrect username/password!'
    return render_template('admin_login.html')



@app.route('/teacher/login', methods=['GET', 'POST'])
```

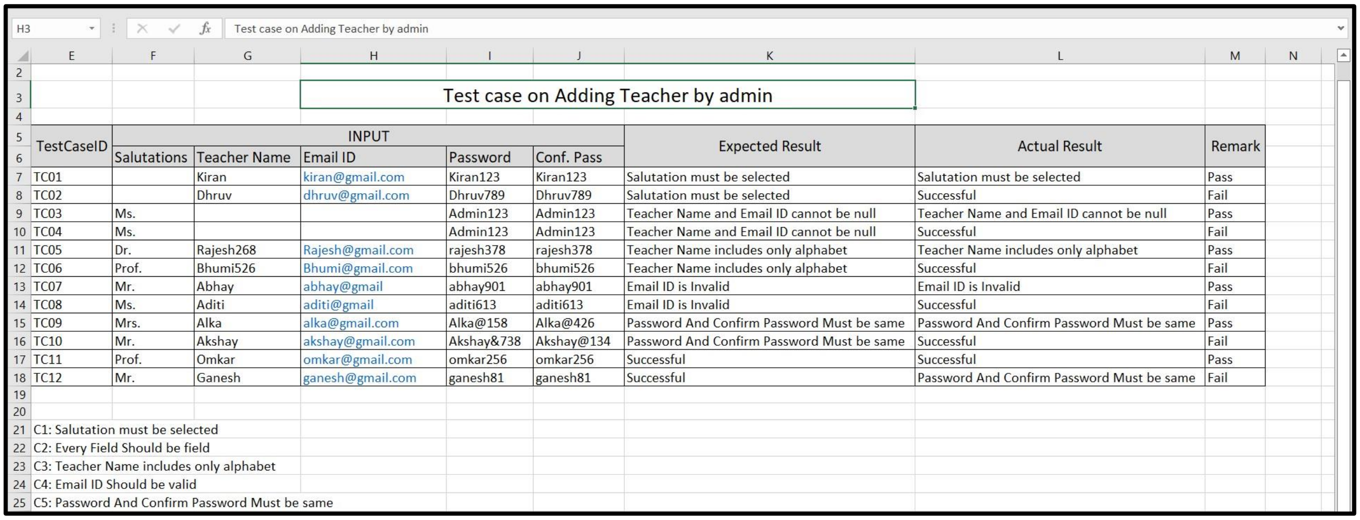
```
def teacher_login():
    if request.method == 'POST':
        teacher_name = request.form['teacher_name']
        email = request.form['teacher_email']
        password = request.form['pass']
        cursor = mysql.connection.cursor(MySQLdb.cursors.DictCursor)
        cursor.execute('SELECT * FROM teacher WHERE name = %s AND email = %s AND 
pswd = %s', (teacher_name, email, password))
        account = cursor.fetchone()
        cursor.execute('UPDATE teacher SET log_in = %s WHERE name = %s', 
(datetime.now(),teacher_name))
        mysql.connection.commit()
        if account:
            session['teacher_loggedin'] = True
            session['id'] = account['id']
            session['teacher_name'] = account['name']
            return redirect('/teacher/dashboard')
        else:
            return 'Invalid username or password'
    return render_template('teacherlogin.html')


@app.route('/teacher/dashboard', methods=['GET', 'POST'])
def teacherdashboard():
    if session.get('teacher_name') != None:
        var1 = session['teacher_name']
        cursor = mysql.connection.cursor()
        cursor.execute("SELECT subject,course,class_name,count(file_name) FROM assign 
WHERE name=%s GROUP BY subject",(var1,))
        cursor.close()
        else:
        data = ''
    return render_template('teacher_dash.html', data=data)



@app.route('/admin/dashboard')
def admindashboard():
    return render_template('admin_dash.html')

def allowed_file(filename):
    return '.' in filename and filename.rsplit('.', 1)[1].lower() in ALLOWED_EXTENSIONS

@app.route('/admin/dashboard/assign',methods=['GET', 'POST'])
def upload_file():
    cur = mysql.connection.cursor()
    now = datetime.now()
    cur.execute('SELECT name FROM teacher')
    teachers_name = cur.fetchall()
    if request.method == 'POST':
        files = request.files.getlist('files[]')
        for file in files:
            if file and allowed_file(file.filename):
                filename = secure_filename(file.filename)
                # new_path = os.path.realpath(filename)
                file.save(os.path.join(app.config['UPLOAD_FOLDER'], filename))
                check_path = os.path.dirname(app.config['UPLOAD_FOLDER'])+ '/upload/'
+filename
```

```
                check_path = check_path[1:]

                name = request.form['teacher_name']

                subject = request.form['subject']

                course = request.form['course']

                class_name = request.form['class_name']

                # flash(new_path)

                cur.execute("INSERT INTO assign (file_name,uploaded_on, name, subject, course,
class_name) VALUES (%s, %s,%s, %s, %s, %s)", [check_path,now,name, subject,course,
class_name])

                mysql.connection.commit()

            else:

                flash("File can't be uploaded")

                break

        cur.close()

    return render_template('assign.html',teacher_name = teachers_name)


@app.route("/logout", methods=['GET', 'POST'])

def logout():

    cursor = mysql.connection.cursor()

    if session.get('teacher_name') != None:

        teacher_name = session['teacher_name']

    if session.get('username') != None and session.get('admin_loggedin') != False:

        session.pop('username',None)

        session['admin_loggedin'] = False

    elif session.get('teacher_name') != None and session.get('teacher_loggedin') != False:

        session.pop('teacher_name',None)

        session['teacher_loggedin'] = False

        cursor.execute('UPDATE teacher SET log_out = %s WHERE name = %s',
(datetime.now(),teacher_name))
```

```
        mysql.connection.commit()
    return redirect(url_for("home"))

@app.route("/teacher/paper")
def paper_check():
    if session.get('teacher_name') != None:
        var2 = session['teacher_name']
        cursor = mysql.connection.cursor(MySQLdb.cursors.DictCursor)
        cursor.execute("SELECT file_name FROM assign WHERE name=%s",(var2,))
        path = cursor.fetchone()
        cursor.close()
    return render_template("file_preview.html",path=path)

@app.route("/admin/dashboard/history")
def history():
    if session.get('username') != None:
        cursor = mysql.connection.cursor()
        cursor.execute("SELECT * FROM teacher")
        history = cursor.fetchall()
    return render_template("history.html", history = history)

if__name__== '_main_':
    app.secret_key = 'super secret key'
    app.run(debug=True)
```

5.2 Testing Approach And Test Cases

Black Box Testing



Test case on Adding Teacher by admin

TestCaseID	INPUT					Expected Result	Actual Result	Remark
	Salutations	Teacher Name	Email ID	Password	Conf. Pass			
TC01		Kiran	kiran@gmail.com	Kiran123	Kiran123	Salutation must be selected	Salutation must be selected	Pass
TC02		Dhruv	dhruv@gmail.com	Dhruv789	Dhruv789	Salutation must be selected	Successful	Fail
TC03	Ms.			Admin123	Admin123	Teacher Name and Email ID cannot be null	Teacher Name and Email ID cannot be null	Pass
TC04	Ms.			Admin123	Admin123	Teacher Name and Email ID cannot be null	Successful	Fail
TC05	Dr.	Rajesh268	Rajesh@gmail.com	rajesh378	rajesh378	Teacher Name includes only alphabet	Teacher Name includes only alphabet	Pass
TC06	Prof.	Bhumi526	Bhumi@gmail.com	bhumi526	bhumi526	Teacher Name includes only alphabet	Successful	Fail
TC07	Mr.	Abhay	abhay@gmail	abhay901	abhay901	Email ID is Invalid	Email ID is Invalid	Pass
TC08	Ms.	Aditi	aditi@gmail	aditi613	aditi613	Email ID is Invalid	Successful	Fail
TC09	Mrs.	Alka	alka@gmail.com	Alka@158	Alka@426	Password And Confirm Password Must be same	Password And Confirm Password Must be same	Pass
TC10	Mr.	Akshay	akshay@gmail.com	Akshay&738	Akshay@134	Password And Confirm Password Must be same	Successful	Fail
TC11	Prof.	Omkar	omkar@gmail.com	omkar256	omkar256	Successful	Successful	Pass
TC12	Mr.	Ganesh	ganesh@gmail.com	ganesh81	ganesh81	Successful	Password And Confirm Password Must be same	Fail

C1: Salutation must be selected

C2: Every Field Should be field

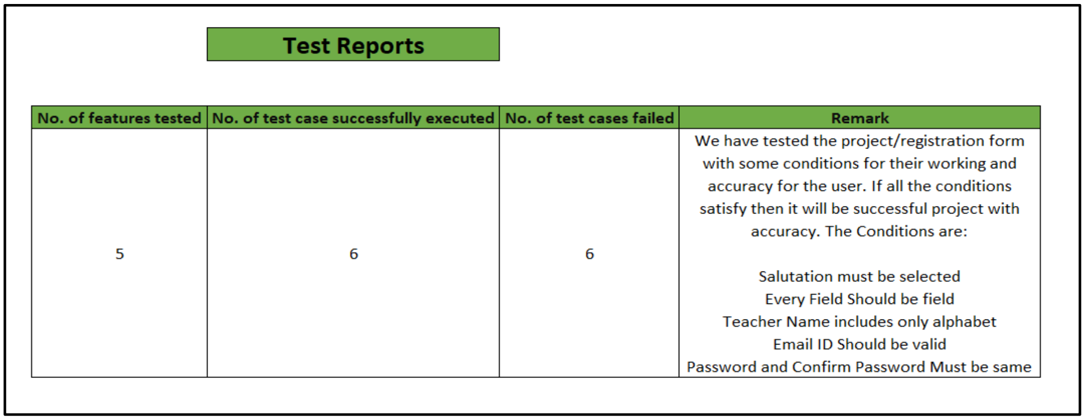
C3: Teacher Name includes only alphabet

C4: Email ID Should be valid

C5: Password And Confirm Password Must be same

CHAPTER 6
RESULTS AND DISCUSSION

6.1 Test Reports



Test Reports

No. of features tested	No. of test case successfully executed	No. of test cases failed	Remark
5	6	6	We have tested the project/registration form with some conditions for their working and accuracy for the user. If all the conditions satisfy then it will be successful project with accuracy. The Conditions are: Salutation must be selected Every Field Should be field Teacher Name includes only alphabet Email ID Should be valid Password and Confirm Password Must be same

6.2 User Documentation

Welcome to the user documentation for our Onscreen Marking System. This website is designed to help you with marking assessments online. In this guide, we will take you through the features of the website and how to use them.

Getting Started:

To get started, you will need to log in to the website using your username and password which will be provided by admin. Once you have logged in, you will be taken to the teacher's dashboard.

Teacher's Dashboard:

The dashboard will display all the assessments that you need to mark. You can select an assessment by clicking on it. Once you have selected an assessment, you will be taken to the marking page.

Assessment Page:

The marking page will display the assessment that you have selected. You will be able to see the questions that need to be marked and the answers that have been provided by the students. To mark a question, click on the question number. This will make you to the assign marks to that particular question and drag-drop the marks to assign marks.

Admin Dashboard:

The admin dashboard allows admin to add teacher account and assign papers to that subject teacher. And it will help admin to check the record of the teacher's login. To logout, click on the logout button.

That's it! We hope that this user documentation has helped you to get started with our Onscreen Marking System. If you have any questions or need further assistance, please contact our support team.

CHAPTER 7
CONCLUSION

7.1 Conclusion

In conclusion, the On Screen Marking System is an innovative and efficient tool that allows teachers and instructors to grade exams digitally. This system has many benefits, including improving the accuracy and consistency of grading, reducing the workload and time required for grading, enhancing security and confidentiality, and improving data analysis and reporting.

By using the On Screen Marking System, educational institutions can move away from paper-based grading and embrace digital methods. This can lead to a more efficient, effective, and transparent grading process that benefits both students and teachers.

Overall, the On Screen Marking System software is a valuable tool that can help educational institutions improve their grading process and provide a better learning experience for their students. It is a worthwhile investment for any institution that values accuracy, efficiency, and transparency in its grading process.

7.1.1 Significance of the System

The on-screen marking system is a digital tool that enables examiners to grade answer scripts on a computer rather than using traditional paper-based methods. This software has several significant benefits:

- Increased efficiency: On-screen marking system reduces the time and effort required for grading exam papers. It allows examiners to access and grade answer scripts from anywhere, and grading can be done much more quickly than traditional paper-based methods.

- Improved accuracy: The system reduces errors caused by illegible handwriting and calculation mistakes, resulting in more accurate grading. It also enables standardization in grading, making it easier to ensure that all examiners are grading in a consistent manner.

- Enhanced security: On-screen marking system is more secure than traditional paper-based methods as it reduces the risk of answer scripts getting lost or stolen. It also enables easier tracking and monitoring of grading activity, ensuring that the process is transparent and accountable.

- Greater accessibility: The system makes it easier for examiners with disabilities to grade answer scripts, as it can be customized to suit their needs. It also enables remote working, allowing examiners to grade answer scripts from anywhere in the world.

7.1.2 Limitations of the System

While on-screen marking system software offers several advantages over traditional paper-based grading methods, there are also some limitations to be aware of:

- Cost: Implementing an on-screen marking system can be costly, particularly if a school or institution needs to invest in new hardware, software licenses, and training for examiners. This may make it difficult for some organizations to adopt this technology.

- Technical issues: The on-screen marking system relies heavily on technology, which can sometimes be prone to technical issues such as software glitches, internet connectivity problems, and hardware failures. These issues can disrupt the grading process and lead to delays.

- Training requirements: Using the on-screen marking system requires specialized training, particularly for examiners who are used to paper-based methods. This can add to the overall cost of implementation and may take time to fully integrate into the grading process.

- Limited flexibility: Some examiners may find that using on-screen marking system is less flexible than traditional paper-based methods. For example, they may not be able to make as many notes or annotations on the answer scripts, or they may be limited in their ability to grade papers in a way that suits their individual preferences.

- Security concerns: While on-screen marking system is generally more secure than traditional paper-based methods, there is still a risk of data breaches and cyber attacks. Schools and institutions need to have robust security protocols in place to mitigate these risks.

7.1.3 Future Scope of the Project

The on-screen marking system has already made a significant impact on the grading process, but there is still room for further development and expansion. Here are some potential areas for further scope:

- Integration with AI: Artificial intelligence (AI) can be used to enhance the on-screen marking system further. For example, AI could be used to detect and flag potential cases of plagiarism, or to help identify patterns in student responses that could inform future teaching practices.

- Integration with learning management systems: On-screen marking system software could be integrated with learning management systems (LMS) to provide a more streamlined grading process. This would allow examiners to access student data and grading information in a single system, reducing the need for manual data entry.

REFERENCES

- https://flask.palletsprojects.com/en/1.1.x/
- https://stackoverflow.com/questions/65832845/how-to-drag-and-drop-same-element-multiple-times- into-another-div-box-in-javascr
- https://www.youtube.com/watch?v=krcyh42ShLg
- https://www.javatpoint.com/flask-session
- https://www.researchgate.net/publication/305781060_Review_of_Research_into_Onscreen_Marking_and_a_Description_of_the_Methodological_Approaches_and_Analytical_Tools_Used_in_the_OSM_Validation_Studies
- https://www.mindlogicx.com/onscreen-marking-system/
- https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/605688/0613_JoTisi_et_al-nfer-a-review-of-literature-on-marking-reliability.pdf
- https://www.testreach.com/blog-post/online-marking-software-why-use-on-screen-exam-marking-tools.html
- https://news.bostes.nsw.edu.au/blog/2010/3/22/onscreen-marking
- https://www.researchgate.net/publication/230897314_A_Review_of_the_Status_of_Online_Semi-Automated_Marking_and_Feedback_Systems

APPENDIX

- Database : The database for an onscreen marking system typically includes information about students, markers, assessments, and marking criteria. It should be designed to store and manage assessment data securely, efficiently, and reliably.
- Student data: This includes information such as student names, IDs, courses, and grades.
- Assessment data: This includes the actual assessment files, such as exam papers or assignments, as well as metadata about the assessment, such as assessment types, deadlines, and marking criteria.
- Test case: To ensure that an onscreen marking system works as intended, it is important to test it rigorously using a range of test cases.
- Security and privacy: This test case ensures that the system complies with data protection and security standards, such as encryption, access control, and audit trails.
- Validation: Validation is the process of ensuring that an onscreen marking system meets the requirements and standards of its intended users and stakeholders.