Sri Sivasubramaniya Nadar College of Engineering, Kalavakkam – 603 110 (An Autonomous Institution, Affiliated to Anna University, Chennai)
Department of Computer Science and Engineering
Practical End Semester Examinations

Name: Ramaprabha R
Class: CSE-B
Register Number: 185001123

Problem Statement:-

Student Mark Analysis System deals with the complete academic details of the students to analyze each student based on the marks obtained. It includes the Name, Roll number, Department, Year of studying, Mark of each subject and result whether the student is pass or fail is given. It can be accessed by the faculty who can change as well as update the mark if required. It is the duty of the faculty and the Administrator to maintain the mark records of the student and calculate the result based on the marks scored by the student.

The only role of the Student is to view their marks entered by the faculty and the Administrator. The Administrator can view all the marks of the student and also the passed and failed number of students can also be seen by him/her. This system eases the work of both faculty and student to a great extent. As the entire work is computerized, no efforts for maintaining a register to enter the marks, details of the student etc each time. Therefore no manual records need to be registered, it avoids calculation errors. So this System is an efficient one for both faculty and Student.

Identification of classes:

a) **Conceptual class category list:**

Conceptual class category	Identified classes
Physical or tangible objects	Student Mark Analysis
Specification or description of things	Student Details, Analysis, Faculty Details
Roles of people	Faculty, Admin, Student
Containers of things	Student Mark Analysis
Things in a container	Admin, Student, Faculty
Events	Login, Get Details, Analysis , View Result

b) **Identification of Noun Phrases:**

Faculty:
1. The faculty enters their login details.
2. The faculty is taken to teacher page after authentication.
3. The faculty can add students and their respective details.

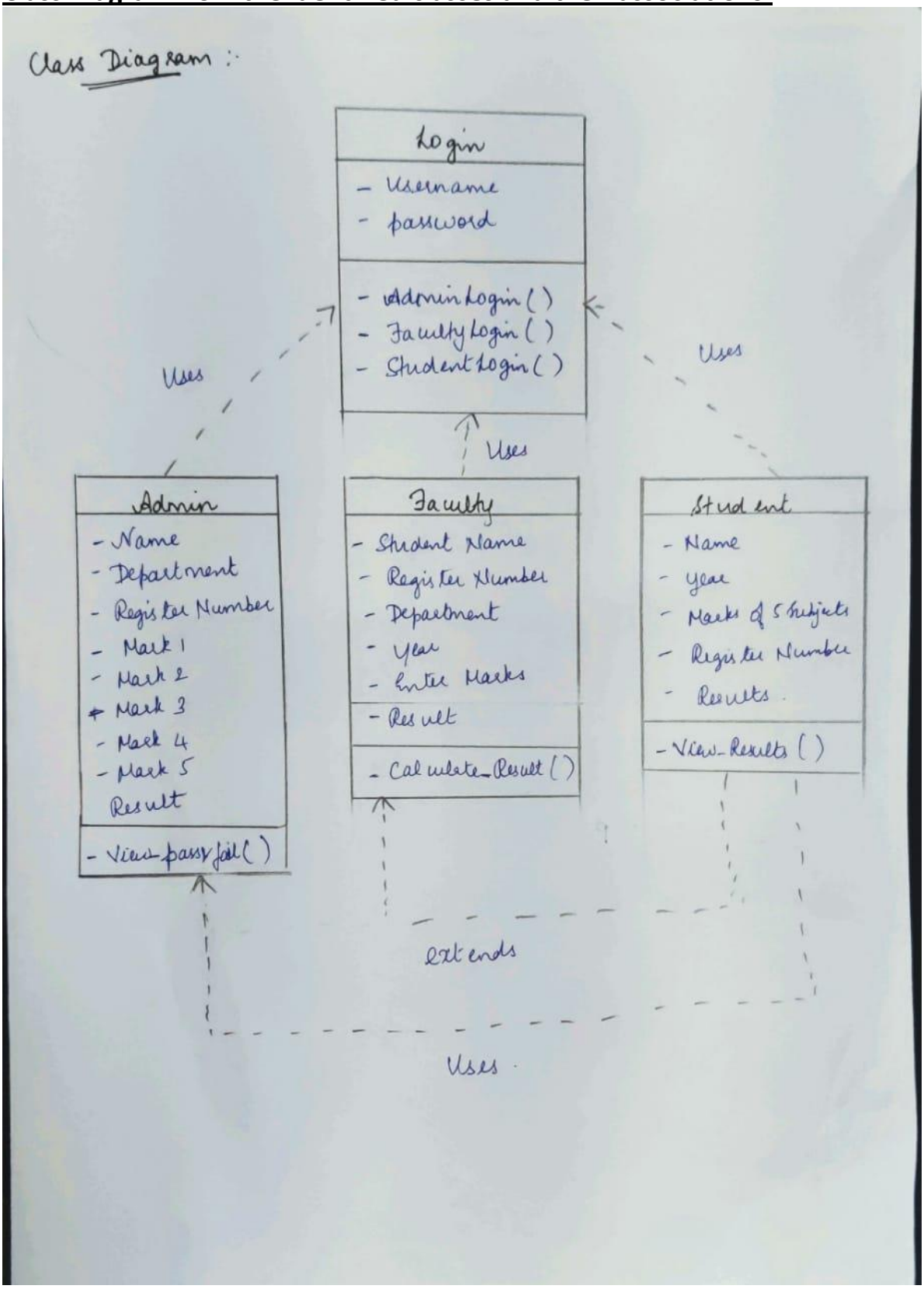
Admin:
1. The admin enters their login details.
2. The admin is taken to admin page after authentication.
3. The admin takes the student marks from the database and analyses the result of each student.
4. After the analysis is done by the admin, he will show number of failed students.

Student:
1. The student enters their login details.
2. The student is taken to student page after authentication.
3. The result of analysis can be viewed by the student.

c) Association category list:

Category	Examples
A is a member of B	Admin – Student Mark Analysis Faculty - Student Mark Analysis Student - Student Mark Analysis
A uses or manages B	Faculty - Details Admin - Analysis Student Mark Analysis - Result
A communicates with B	Admin -- Student
A related to a transaction B	Faculty – Mark entering Admin – Analysis Admin – Result Student - View
A is a transaction related to another transaction B	Analysis - Result

Class Diagram from the identified classes and their associations:



Identification of Test case Scenarios:

- ✓ Faculty Login
- ✓ Entering the Details of the Student
- ✓ Calculating the result based on the mark above or below 50
- ✓ Admin Login
- ✓ Student viewing the result

Test Cases:-

Test_ID	Test_Scenario	Test Step	Test Data	Predicted Result	Actual Result	Pass/Fail
T01	Faculty Login	Enter Username &Password	Username: Varun Password:1234	Logged in	Logged in	Pass
T02	Faculty Login	Enter Username &Password	Username: Varun Password:5678	Invalid, Try Again	Logged in	Fail
T03	Entering Student Details	Once the faculty is logged in he/she can enter the student details	Name: Devi Dept: CSE Year: 2021 Register Number: 123 Mark1: 65 Mark2: 98 Mark3: 84	Student Details entered successfully	Student Details entered successfully	Pass
T04	Entering Student Details	Once the faculty is logged in he/she can enter the student details	Name: Dept: CSE Year: 2021 Register Number: 123 Mark1: 65 Mark2: 98 Mark3: 84	Fill the required field	Student Details entered successfully	Fail
T05	Calculating mark of each Student	If the student scores less than 50 then the student is failed	Name: Devi Dept: CSE Year: 2021 Register Number: 123	Student failed to pass the exam	The result is sent to the Admin as Pass	Pass

		else he is passed	Mark1: 65 Mark2: 26 Mark3: 84			
T06	Calculating mark of each Student	If the student scores less than 50 then the student is failed else he is passed	Name: Devi Dept: CSE Year: 2021 Register Number: 123 Mark1: 65 Mark2: 95 Mark3: 84	Student has passed the exam	The result is sent to the Admin as Fail	Fail
T07	Admin Login	Enter Username &Password	Username: admin Password: root	Logged in	Logged in	Pass
T08	Admin Login	Enter Username &Password	Username: admin Password:1234	Invalid, Try Again	Logged in	Fail
T09	Student viewing mark details	Once the student is logged in he/she and only view the marks obtained and the result pass (or) fail is displayed	Name: Devi Dept: CSE Year: 2021 Register Number: 123 Mark1: 65 Mark2: 95 Mark3: 84 Result: Pass	Student result viewed Successfully	Student result viewed Successfully	Pass
T10	Student viewing mark details	Once the student is logged in he/she and only view the marks obtained and the result pass (or) fail is displayed	Name: Devi Dept: CSE Year: 2021 Register Number: 123 Mark1: 65 Mark2: 95 Mark3: 84 Result: Pass	Student result viewed Successfully	Database Retrieval error Student result not displayed	Fail

Improvement:-
With the help of test cases we ensured that if different features mentioned above within an application are working as expected. It validates whether our software is free of defects and is working as per the expectations of the end users. With Test case scenarios we also improved the quality of our Project.

Code:

Index:

```
<html>
    <head></head>
    <body style="background:pink;">
        <center>
             <h2 style="padding-top:200px;">SSN MARK ANALYSIS</h2><br>
            <form action="/" method="POST">

            <p>Please enter your username:</p>
            <input type="text" id="email" name="email">

                <p>Please enter your password:</p>
                <input type="text" id="pass" name="pass">
                <br><br><br>

                <input type="submit" value="Go to login">
            </form>

        </center>
    </body>
</html>
```

Login:

```
<html>
    <head></head>
    <body style="background:pink;">
        <center>
<h2 style="padding-top:200px;">Login page:</h2><br>
<form method="post" action="/">
    <p>Please enter your E-mail ID:</p>
    <input type='text' id='mail' name='mail' placeholder="Email id" required> <br
>
     <p>Please enter your Password:</p>
    <input type='password' id='pwd' name='pwd' placeholder="password" required> <
br><br>
    <button type="submit" id='button' name="button"    >login</button><br><br>
```

```
    <a href="/register"">Register</a><br>
</form>
</center>
    </body>
</html>
```

Admin:

```
<html>
<head>
<style>
table {
  font-family: arial, sans-serif;
  border-collapse: collapse;
  width: 100%;
}

td, th {
  border: 2px solid #000000;
  border-style : groove;
  text-align: left;
  padding: 8px;
}

tr:nth-child(even) {
  background-color: #dddddd;
}
</style>
</head>
<body  style="background:pink;">
    <center>
    <h2 style="padding-top:200px;">The Student Details are as follows:</h2><br>
<table>
  <tr>
    <th>Register Number</th>

    <th>Name</th>

    <th>Department</th>
    <th>Mark1</th>
    <th>Mark2</th>
    <th>Mark3</th>
    <th>Mark4</th>
    <th>Mark5</th>
    <th>Result</th>
  </tr>
```

```
{%for i in items%}
<tr>
<td>{{i.Regno}}</td>

<td>{{i.Name}}</td>

<td>{{i.Year}}</td>

<td>{{i.Mark_1}}</td>

<td>{{i.Mark_2}}</td>

<td>{{i.Mark_3}}</td>

<td>{{i.Mark_4}}</td>

<td>{{i.Mark_5}}</td>

<td>{{i.Result}}</td>
</tr>
{%endfor%}
</table>
<a href="/view_stats">View stats</a>
</center>
</body>
</html>
```

View:-

```
<html>
    <head></head>
<body style="background:pink;">
        <center>
<h3> The Analysis of the results are as follows:</h3>
<h5>Total no of students passed =  {{items}}<h5>
<h5>Total no of students failed=  {{it}}<h5>
</center>
    </body>
</html>
```

Student:

```
<html>
<head>
<style>
```

```
table {
  font-family: arial, sans-serif;
  border-collapse: collapse;
  width: 100%;
}

td, th {
  border: 2px solid #000000;
  border-style : groove;
  text-align: left;
  padding: 8px;
}

tr:nth-child(even) {
  background-color: #dddddd;
}
</style>
</head>
<body  style="background:pink;">
    <center>
    <h2 style="padding-top:200px;">The Student Details are as follows:</h2><br>
<table>
  <tr>
    <th>Register Number</th>

    <th>Name</th>

    <th>Year</th>
    <th>Mark1</th>
    <th>Mark2</th>
    <th>Mark3</th>
    <th>Mark4</th>
    <th>Mark5</th>
    <th>Result</th>
  </tr>
<tr>
<td>{{items.Regno}}</td>

<td>{{items.Name}}</td>

<td>{{items.Year}}</td>

<td>{{items.Mark_1}}</td>

<td>{{items.Mark_2}}</td>
```

```
<td>{{items.Mark_3}}</td>

<td>{{items.Mark_4}}</td>

<td>{{items.Mark_5}}</td>

<td>{{items.Result}}</td>
</tr>
</table>
</center>
</body>
</html>
```

Faculty:

```
<html>
    <head>

    </head>
    <body  style="background:pink;">
    <center>
    <h2 style="padding-top:200px;">Please enter the Student Details:</h2><br>
        <form action="#" method="POST">
                <label for="name">Name:</label><br>
                <input type="text" id="name" name="name"><br>
                <label for="numer">Register Number:</label><br>
                <input type="text" id="numer" name="numer"><br>
                <label for="sem">SEM</label><br>
                <input type="text" id="sem" name="sem"><br>
                <label for="dept">Department:</label><br>
                <input type="text" id="dept" name="dept"><br>
                <label for="year">Year:</label><br>
                <input type="text" id="year" name="year"><br>
                <h2> Enter Marks:</h2>
                <div class="mark" style="border: 2 rem; width: 20rem;">
                <label for="Mark1">Mark1:</label><br>

                <input type="text" id="Mark1" name="Mark1"><br>
                <label for="Mark1">Mark2:</label><br>
                <input type="text" id="Mark2" name="Mark2"><br>
                <label for="Mark1">Mark3:</label><br>
                <input type="text" id="Mark3" name="Mark3"><br>
                <label for="Mark1">Mark4:</label><br>
                <input type="text" id="Mark4" name="Mark4"><br>
```

```
                <label for="Mark1">Mark5:</label><br>
                <input type="text" id="Mark5" name="Mark5"><br><br>
            </div>
                <br>
                <input type="submit" value="Submit">
        </form>
    </center>
    </body>
</html>
```

Python code file:

```
from flask import Flask, url_for , request, redirect , session, g
from flask import render_template
from flask_sqlalchemy import SQLAlchemy
from datetime import datetime,date,timedelta
from sqlalchemy.orm import backref
import sqlite3

app=Flask(__name__)
app.config['SQLALCHEMY_DATABASE_URI'] = 'sqlite:///practicals.db'
app.secret_key = 'ramaprabha'

db = SQLAlchemy(app)

class Student(db.Model):
    __tablename__ = 'student'
    id = db.Column(db.Integer,primary_key=True)
    Regno=db.Column(db.Integer)
    Name = db.Column(db.String(50))
    Year=db.Column(db.Integer)
    Dept=db.Column(db.String(50))
    Mark_1=db.Column(db.Integer)

    Mark_2=db.Column(db.Integer)

    Mark_3=db.Column(db.Integer)

    Mark_4=db.Column(db.Integer)

    Mark_5=db.Column(db.Integer)

    Result=db.Column(db.String(50))

class Staff(db.Model):
```

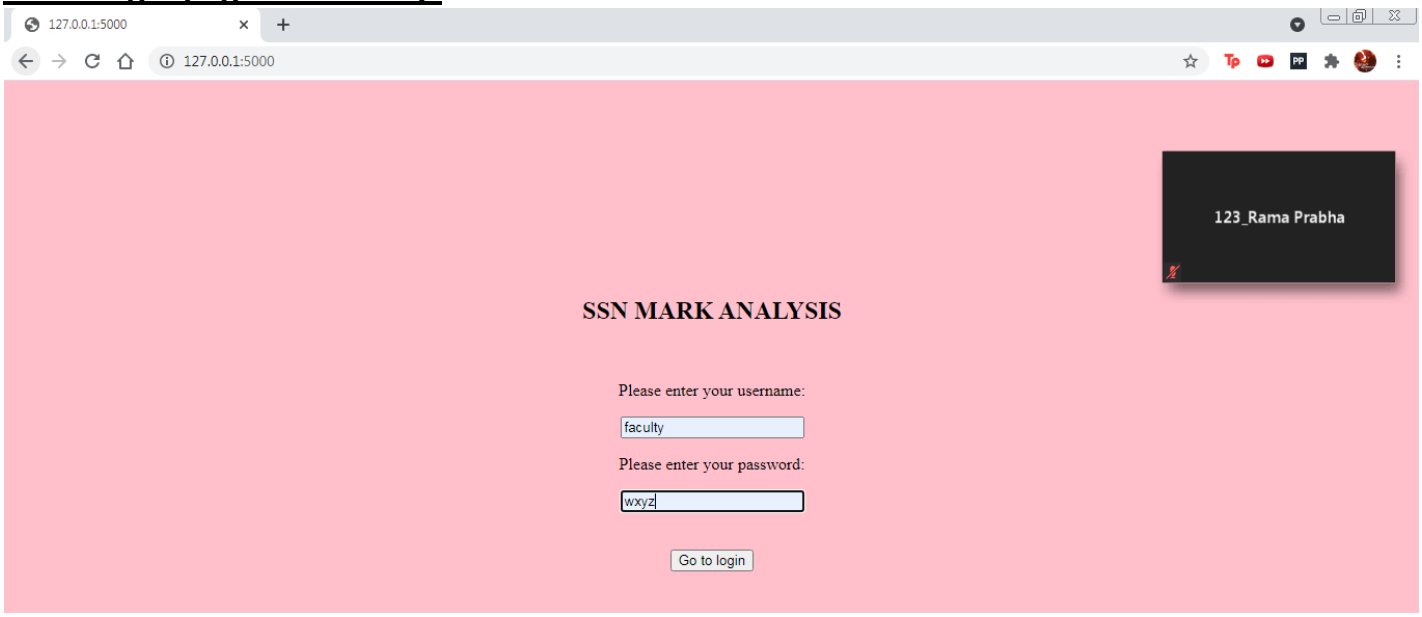
```
    __tablename__ = 'staff'
    id = db.Column(db.Integer,primary_key=True)
    Email = db.Column(db.String(50))
    Password = db.Column(db.String(50))
class COE(db.Model):
    __tablename__ = 'coe'
    id = db.Column(db.Integer,primary_key=True)
    Emailid = db.Column(db.String(50))
    Password = db.Column(db.String(50))




def connect_db():
    return sqlite3.connect('practicals.db')

@app.before_request
def before_request():
    if 'username' in session:
        g.user = session['username']

@app.route('/',methods=['POST', 'GET'])
def login():
    if request.method == 'POST':
        emailid = request.form.get('email')
        print(emailid)
        passwd = request.form.get('pass')
        print(passwd)
        selectapplicant = Student.query.filter_by(Regno=emailid,Name=passwd).firs
t()
        #selectapplicant1 = COE.query.filter_by(Emailid=emailid,Password=passwd).
first()
        if emailid=="faculty" and passwd=="wxyz":
            return redirect('/teacherpage')
        if emailid=="admin" and passwd=="root":
            #session['username']=selectapplicant.id
            return redirect('/coepage1')
        if selectapplicant:

            session['username']=selectapplicant.Regno
            print(session['username'])
            print("Bye",selectapplicant.Regno)
            return redirect('/studentpage')

    return render_template('index.html')
```

```python
@app.route('/studentpage',methods=['POST', 'GET'])
def studentpage():
    givenstu=Student.query.filter_by(Regno=session['username']).first()
    print(session['username'])
    print(givenstu)
    return render_template('studentpage.html',items=givenstu)


@app.route('/teacherpage',methods=['POST', 'GET'])
def teacherpage():
    if request.method=='POST':
        regno=request.form['numer']
        name=request.form['name']
        sem=request.form['sem']
        dept=request.form['dept']
        year=request.form['year']
        mark1=request.form['Mark1']
        mark2=request.form['Mark2']
        mark3=request.form['Mark3']
        mark4=request.form['Mark4']
        mark5=request.form['Mark5']
        if(int(mark1)>=35 and int(mark2)>=35 and int(mark3)>=35 and int(mark4)>=3
5 and int(mark5)>=35):
            result="pass"
        else:
            result="fail"
        data = Student(Regno=regno,Name=name,Year=year,Dept=dept,Mark_1=mark1,Mar
k_2=mark2,Mark_3=mark3,Mark_4=mark4,Mark_5=mark5,Result=result)
        db.session.add(data)
        db.session.commit()
        return "Sucessfully added"
    else:
        return render_template('teacherpage.html')


@app.route('/coepage1',methods=['POST', 'GET'])
def coepage():
    data = Student.query.all()
    print(data)
    return render_template('coepage1.html',items=data)
@app.route('/view_stats',methods=['POST', 'GET'])
def view_pass_stats():
    passed = Student.query.filter_by(Result="pass").count()
```

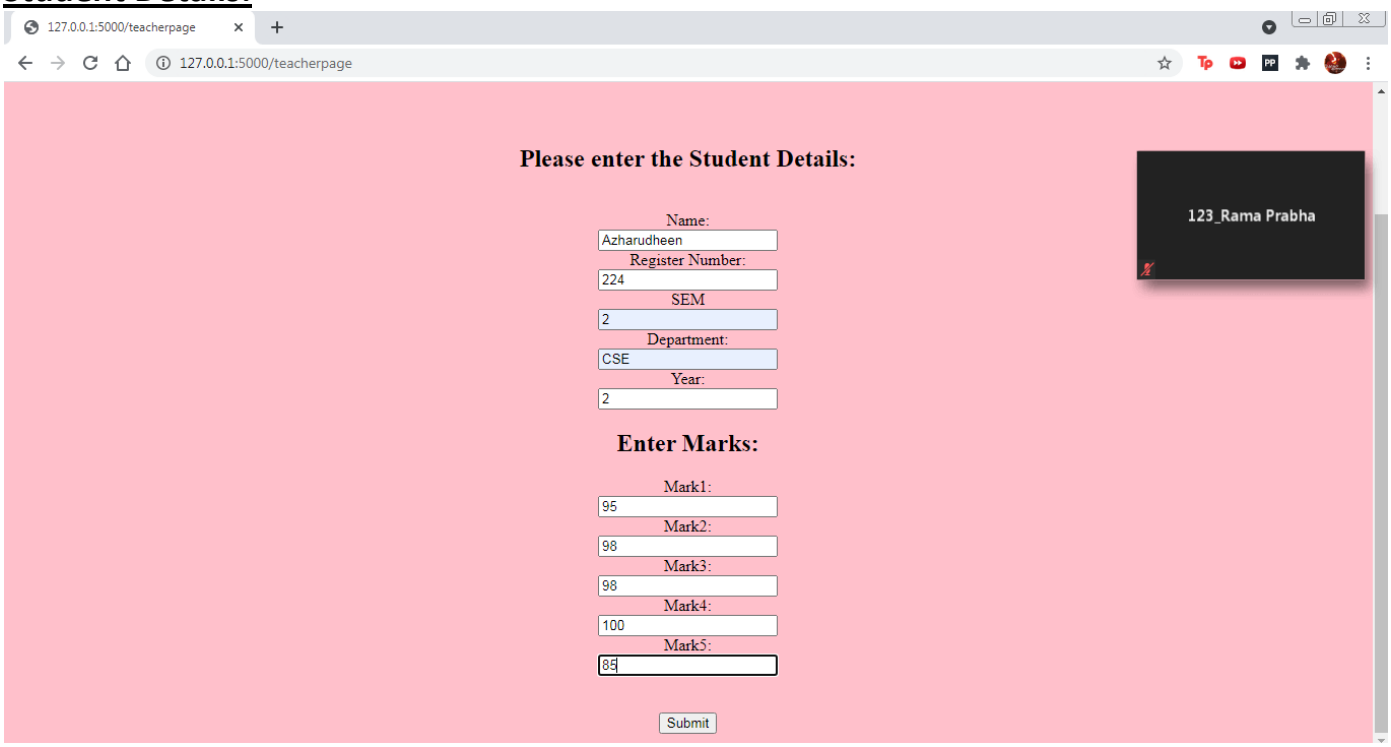
```
    failed= Student.query.filter_by(Result="fail").count()
    return render_template('pass.html',items=passed,it=failed)
if __name__ == "__main__":
    app.run(debug=True)
```

Output:-

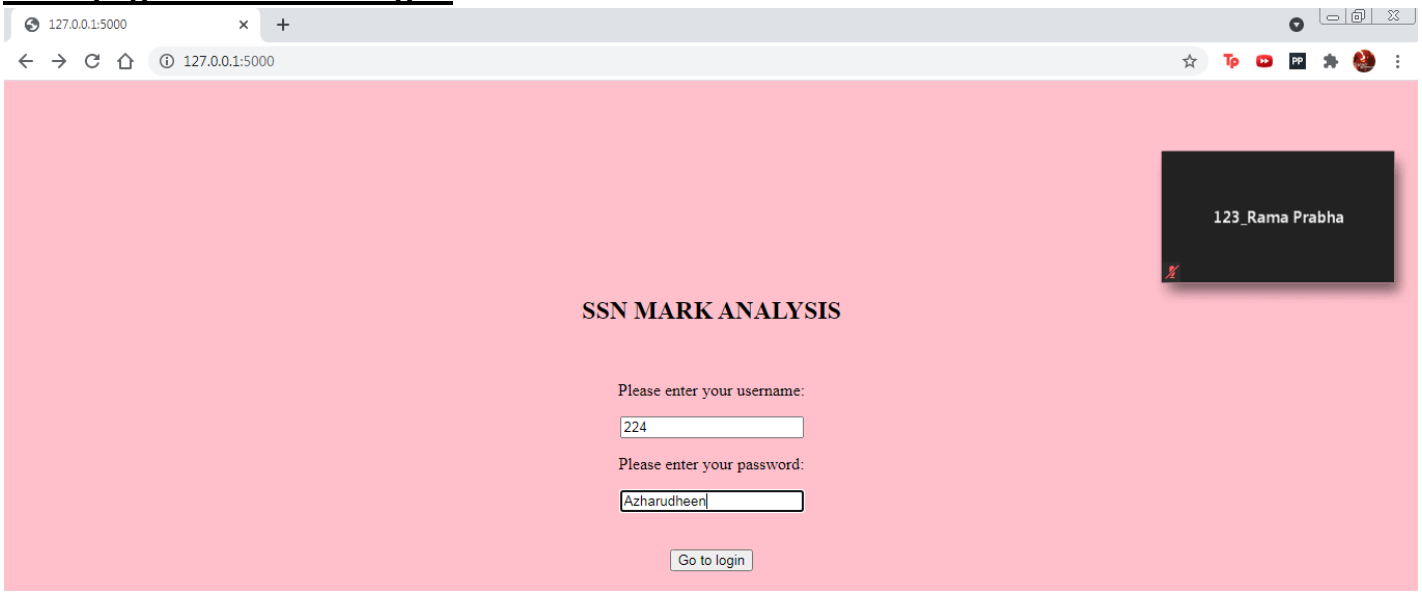
Main Login page for Faculty:



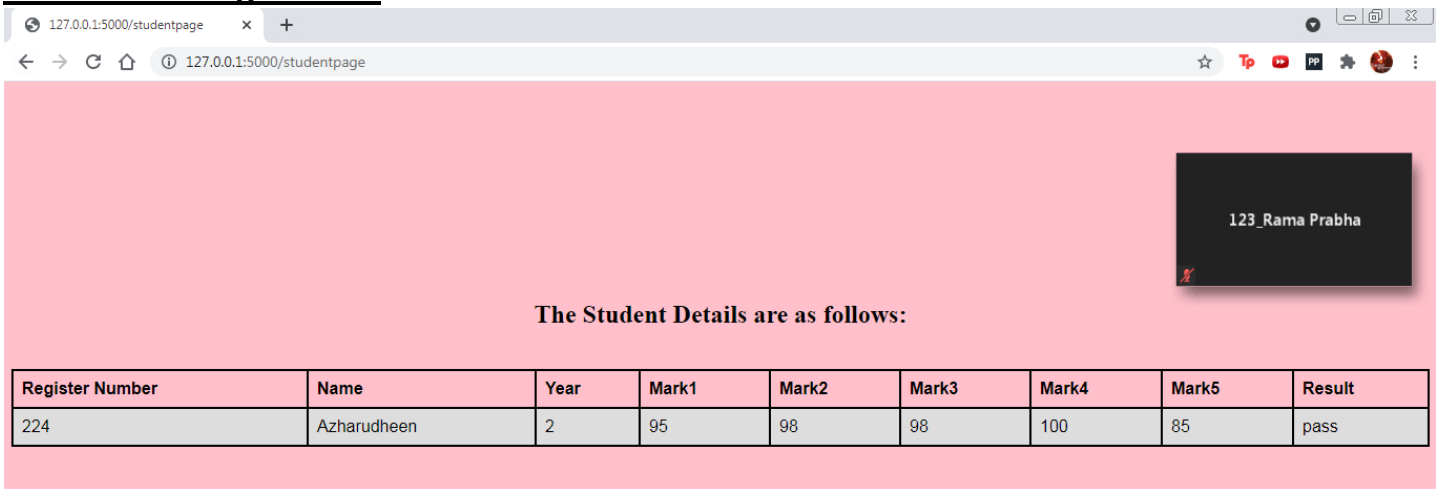
Student Details:



Main page for Student Login:

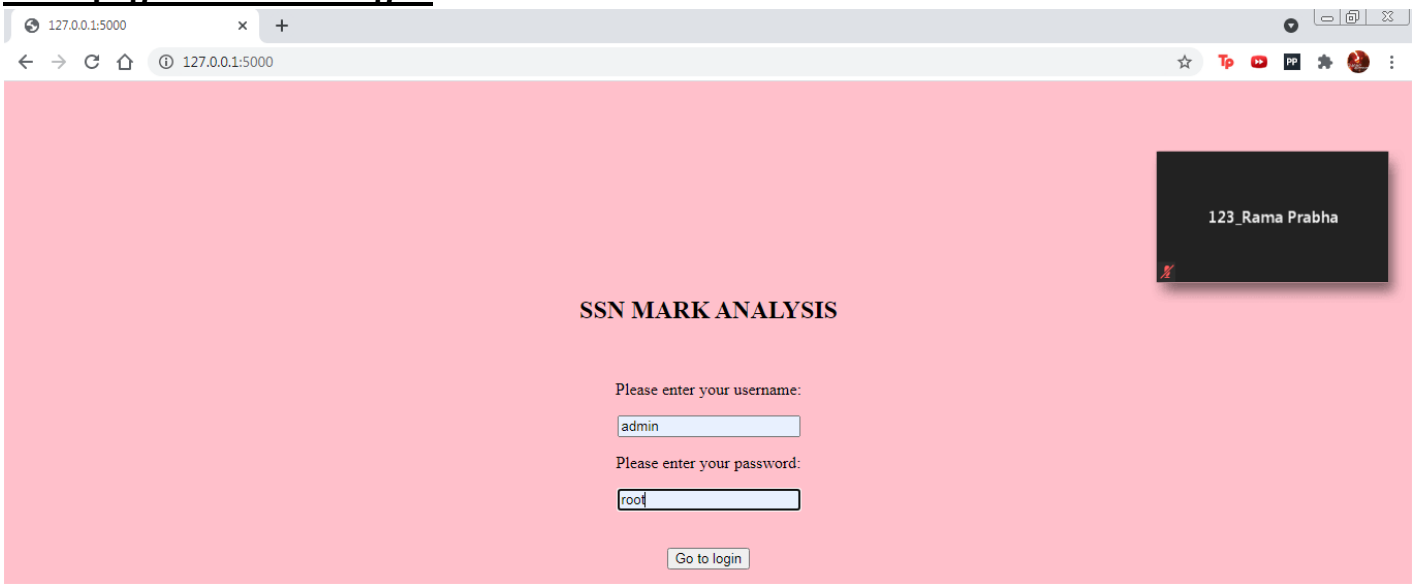


Student viewing Details:

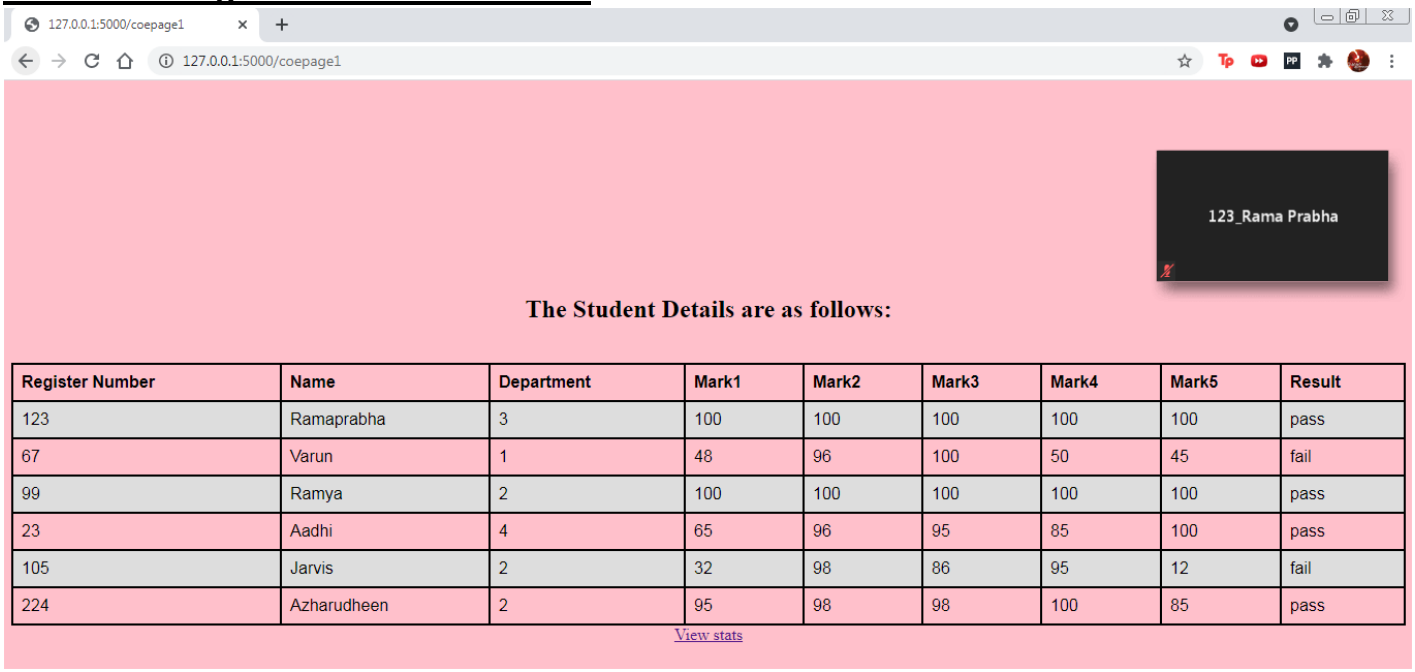


Register Number	Name	Year	Mark1	Mark2	Mark3	Mark4	Mark5	Result
224	Azharudheen	2	95	98	98	100	85	pass

Main page for Admin Login:

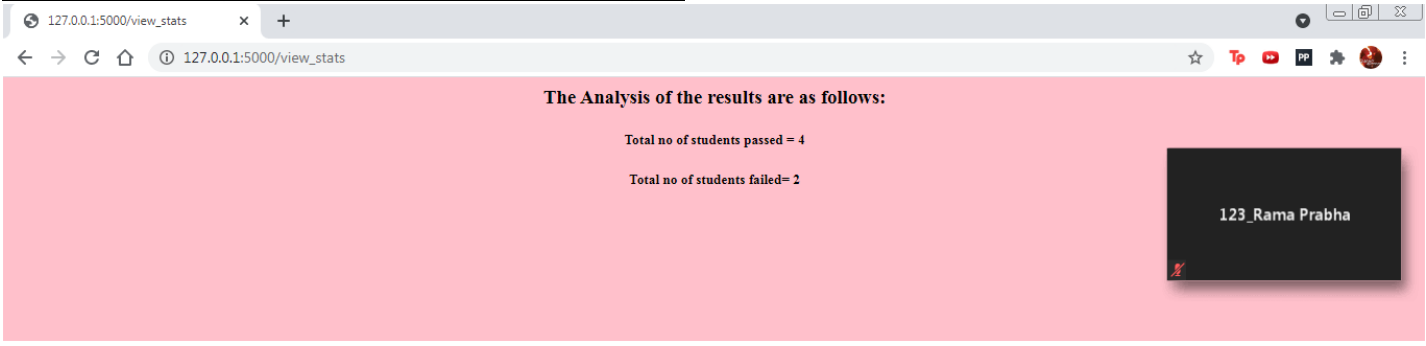


Admin viewing Details of all Students:



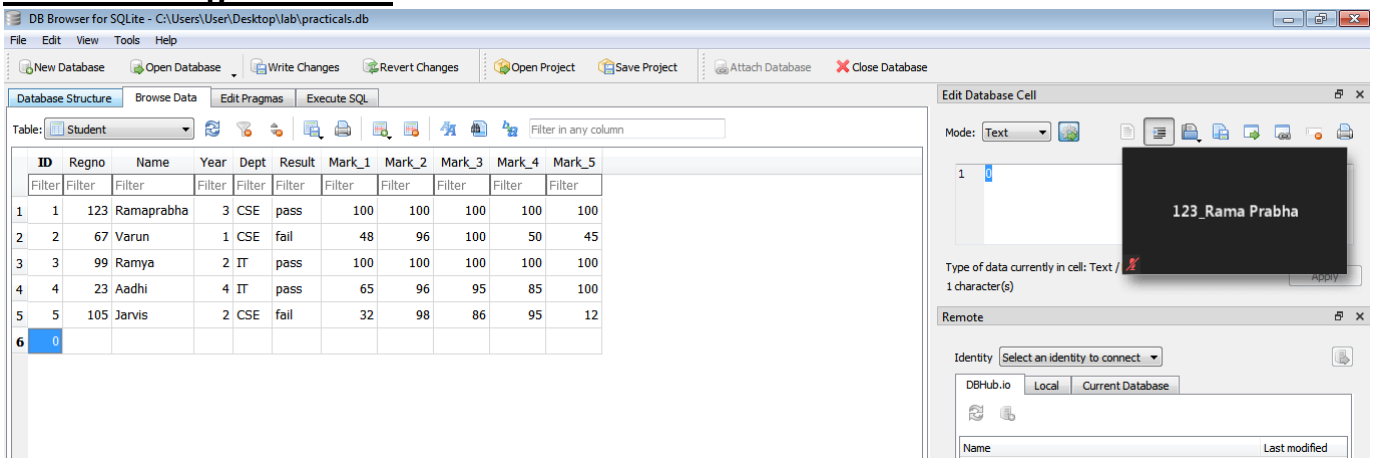
Register Number	Name	Department	Mark1	Mark2	Mark3	Mark4	Mark5	Result
123	Ramaprabha	3	100	100	100	100	100	pass
67	Varun	1	48	96	100	50	45	fail
99	Ramya	2	100	100	100	100	100	pass
23	Aadhi	4	65	96	95	85	100	pass
105	Jarvis	2	32	98	86	95	12	fail
224	Azharudheen	2	95	98	98	100	85	pass

Admin viewing entire pass and fail strength:



Database:

Before Adding a Student:



After Adding a Student: