# Major Project Report

## on

# College Fee Dues Checking System

*A report Submitted in partial fulfilment of the requirements of the degree.*

**Bachelor of Technology**
**IN**

## COMPUTER SCIENCE AND ENGINEERING

**Submitted by**

Shaik Reshma(R170205)

Under the Guidance of

**Mr. A. Mahendra**

**Assistant Professor**

**Department of Computer Science and Engineering**

**Rajiv Gandhi University of Knowledge Technologies RK Valley Kadapa**

**Andhra Pradesh**

## DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

## RAJIV GANDHI UNIVERSITY OF KNOWLEDGE TECHNOLOGIES

IIIT RK VALLEY -516330

# DECLARATION

I hereby declare that the report of the B. Tech Major Project Work entitled with the "**College Fee Dues Checking System**" which is being submitted to Rajiv Gandhi University of Knowledge Technologies, RK Valley, in partial fulfilment of the requirements for the award of Degree of Bachelor of Technology in Computer Science and Engineering, is a bonafide report of the work carried out by me. The material contained in this report has not been submitted to any university or institution for award of any degree.

**SHAIK RESHMA-R170205**

## RAJIV GANDHI UNIVERSITY OF KNOWLEDGE TECHNOLOGIES

## RGUKT

(A.P Government Act 18 of 2008)

RGUKT, RK VALLEY

Department of Computer Science and Engineering

# CERTIFICATE FOR PROJECT COMPLETION

This is to certify that the project entitled “ **College Fee Dues Checking System**” submitted by the **SHAIK RESHMA(R170205)** under our guidance and supervision for the partial fulfilment for the degree Bachelor of Technology in Computer Science and Engineering during the academic year 2022-2023 at RGUKT, RK VALLEY. To the best of my knowledge, the results embodied in this dissertation work have not been submitted to any University or Institute for the award of any degree or diploma.

**Project Internal Guide**

Mr. A. Mahendra

Assistant Professor

RGUKT, RK VALLEY.

**Head of the Department**

Mr. N. Satyanandaram

HOD of CSE

RGUKT, RK VALLEY.

Submitted on ………………………………

# ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of any task would be incomplete without the mention of the people who made it possible and whose constant guidance and encouragement crown all the efforts success.

I am extremely grateful to our respected Director, Prof. K. SANDHYARANI for fostering an excellent academic climate in our institution.

I also express my sincere gratitude to our respected Head of the Department Mr. N. SATYANANDARAM for his encouragement, overall guidance in viewing this project a good asset and effort in bringing out this project.

I would like to convey thanks to our guide at college Mr. A. MAHENDRA for his guidance, encouragement, co-operation and kindness during the entire duration of the course and academics.

My sincere thanks to all the members who helped me directly and indirectly in the completion of project work. I express my profound gratitude to all our friends and family members for their encouragement.

# Index

1.Abstract……………………………………………………………06

2.Introduction…………………………………………………………..06

2.1 Purpose……………………………………………………07

2.2 Intended Audience………………………………………...07

2.3 Product Vision……………………………………………07

2.4 Existing System & Proposed System……………………...07

2.5 Requirement Specification………………………………...08

3. Modules

3.1 Homepage…………………………………………………09

3.2 Admin……………………………………………………...09

3.3 Profile……………………………………………………..09

3.4 Receipts…………………………………………………….10

3.5 Information…………………………………………………10

4. Advantages…………………………………………………………13

5. Technologies Used………………………………………………….14-15

6. Use Case Diagrams…………………………………………………16-17

7. ER-Diagram………………………………………………………...17-18

8. Source Code with Outputs…………………………………….19-45

9. Conclusion………………………………………………………….45

10. Future Enhancement……………………………………………....46

11. References…………………………………………………………47

# ABSTRACT

In the existing system, every student has to collect and store their transactions details at one place or go to sbi collect and check transaction history year by year by filling all the details and go finance section to verify their transactions and know their dues.

College fee dues checking system is an online web-based system in which, student of rkv can login with credentials provided by the college, check their transactions details made through sbi collect and know the fee structure and dues have to be paid by them.

# INTRODUCTION

The main objective of the proposed system is to reduce students’ effort in the college and making them clear about all the information.

College fee dues checking system, this web application will help the users to save their time in going to finance office to check due amounts and also displays previous transaction details made by them such that they need not to maintain all the records or go to sbi collect and fill all the details to get transaction details every time year by year.

This application displays fee’s structure followed in the college so that the students are aware of how much money has to be paid by them and also displays the details of the students.

## Purpose:

The purpose of College fee dues checking system is to help the college students to save their time in writing all the transaction details in paper by going through sbi collect or saved files and going physically to the Finance office to check dues. Students doesn't need to wait at the finance office and all the fee related details will be displayed at one.

## Intended Audience:

1) Admin
2) Students

## Product Vision:

The main vision of this project is to make changes to the old style of college fees, dues management system to be less tiresome and ease to the students by using the web application.

## Existing System:

- In the existing system, if any student wants to know about the due fee, they have to go to the Finance section to know the particulars.

- To verify the amount paid by them, they must check through various websites by which they paid and must maintain the records of their previous transactions to show them to the scholarship section for the verification.

## Proposed System:

- In the proposed system, students can know the details of transactions made by them by just logging in to the website using credentials.
- They can easily get the information of the total fee structure, due amounts and total amount which need to be paid by just browsing through the website.
- Will reduce the effort to maintain records of the previous transactions.

# REQUIREMENT SPECIFICATION:

## Hardware Configuration:

### Client Side:

RAM : 512 MB

Harddisk : 10 GB

Processor : 1.0 GHz

### Server Side:

RAM : 512 MB

Harddisk : 20 GB

Processor : 2.0 GHz

## Software Requirements: -

Front end - HTML, CSS, Bootstrap

Backend - Django

Database Server – SQLite

Web Browser – Chrome, Mozilla Firefox

OS - Ubuntu, Windows

Environment – Visual Studio Code

# MODULES

- **LOGINPAGE**
- **ADMIN**
- **PROFILE**
- **RECEIPTS**
- **INFORMATION**

# MODULES DESCRIPTION

## Login Page:

The Login page is the first page of the website. User has to first login using credintials provided by the college. This page consists of basic UI and includes a login form which takes student id and password as password as input. If any student forgets his/her password they can click on forgot password and the credentials will be sent to their college mail.

## Admin:

Admin is the super user who has access all over the website. He is the one who maintains and updates student’s details and login information, updates transactions details which he got as an excel sheet to the database and changes information about feeing structure in the website etc...

## Profile:

Profile module, it contains details of the student who logged in such as their photo, name, id no, cgpa , branch, category etc...

## Receipts:

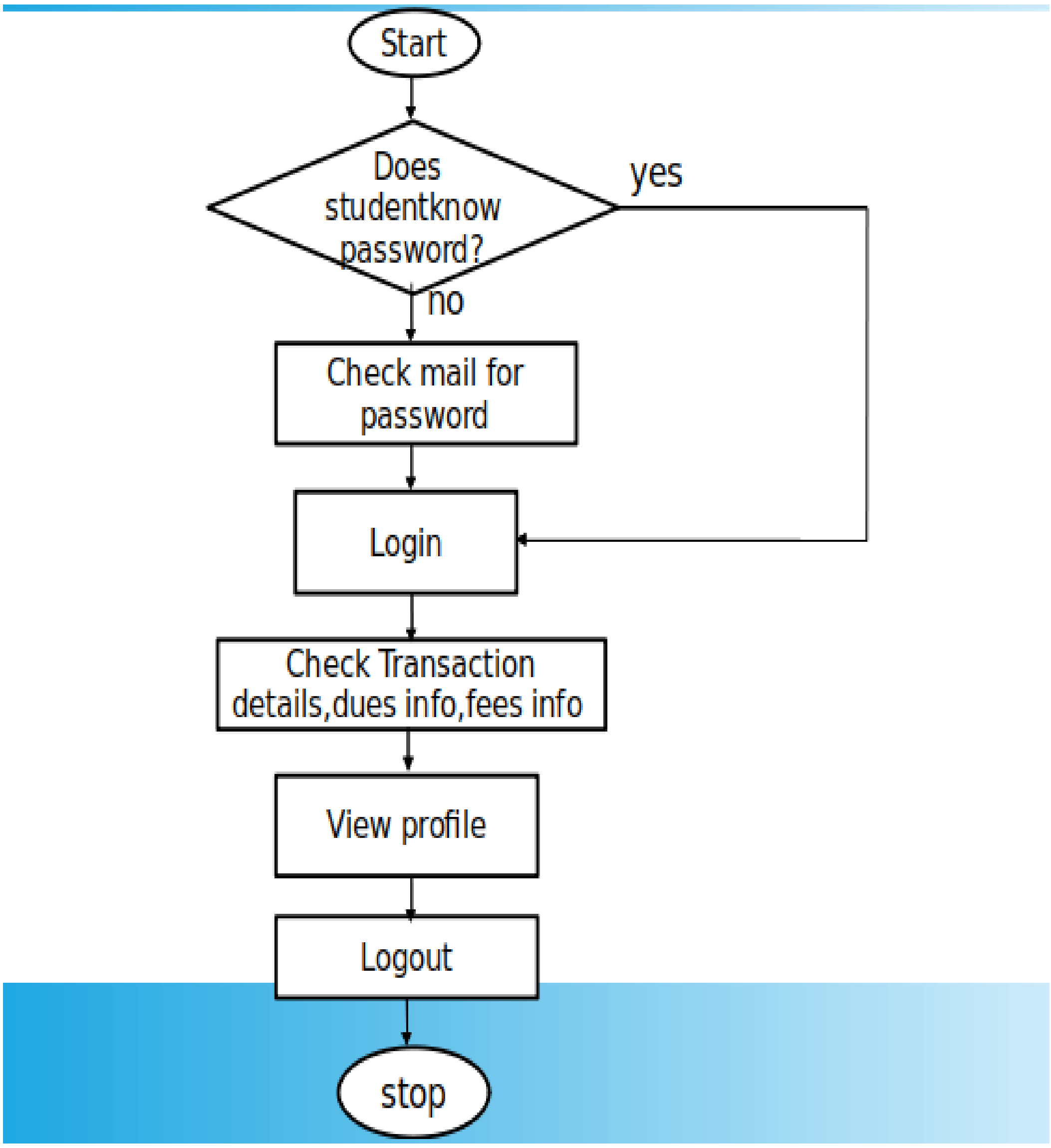
Receipts module contains details of the transactions such as transaction id, amount paid, date of transaction made by the student, total amount paid and the dues have to be paid by the student.

## Information:

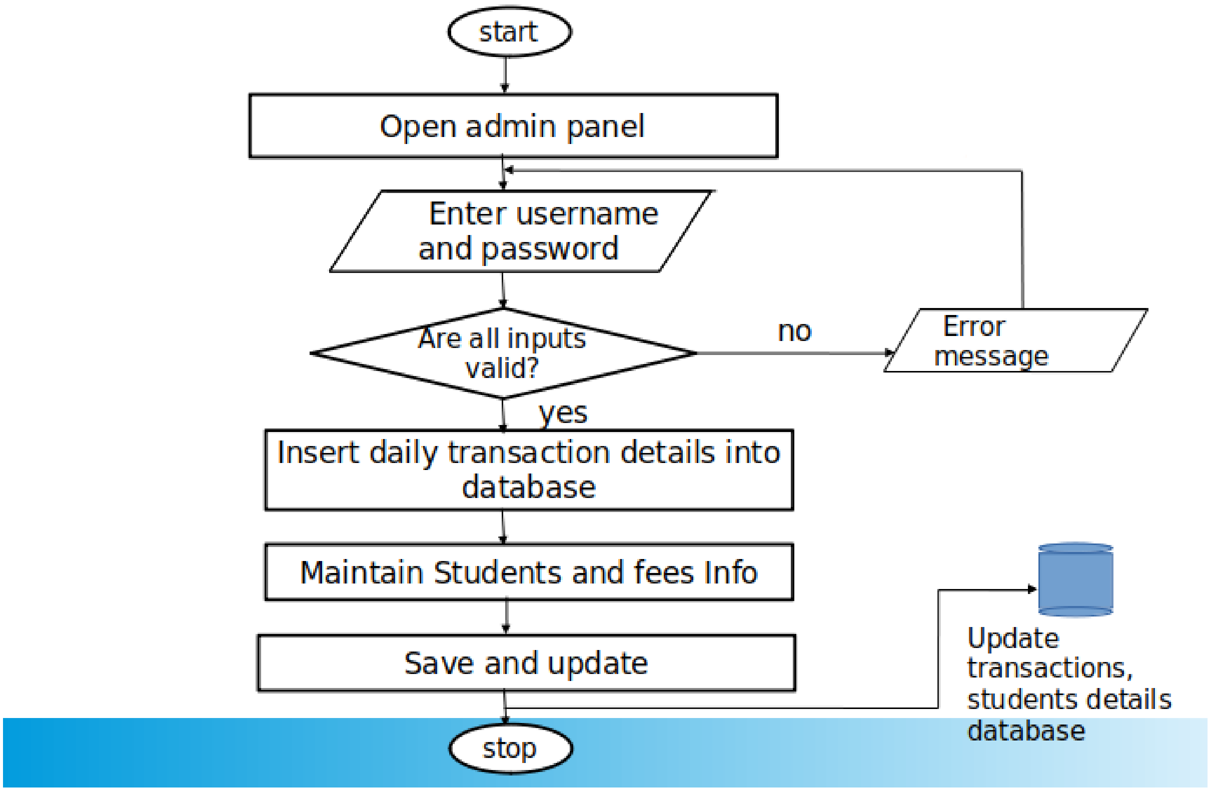
Information module, this includes page which contains information about the feeing structure followed by the structure, that is all the amounts which need to be paid for every year are specified in a tabular form.

# USER FLOW CHART

## User/Student Flow Diagram :


Start
Does studentknow password?
yes
no
Check mail for password
Login
Check Transaction details,dues info,fees info
View profile
Logout
stop

## Admin Flow Diagram:


start
Open admin panel
Enter username and password
Are all inputs valid?
no
Error message
yes
Insert daily transaction details into database
Maintain Students and fees Info
Save and update
Update transactions, students details database
stop

# Advantages of Proposed Technology

## For Organization:

College Fee Dues Checking System, allows organizations for better financial management and provide colleges with real-time data on fee collections, allowing them to better manage their finances and make informed decisions.

As we are hosting a website online, it is much efficient way of storing and retrieving information for fees management.

A fee dues checking system can help organization stay compliant with regulatory requirements by ensuring that fee payments are correctly tracked and accounted for.

Fee dues checking system can accurately calculate fees owed, apply discounts, and process payments. This reduces the risk of errors and ensures that students are charged the correct fees.

## For Students:

Provides students with an easy-to-use online platform for fee dues checking, and viewing fee transactions details system can improve their experience and reduce the stress associated with manual methods.

Fee dues checking systems can accurately calculate fees owed and process payments, reducing the risk of errors and improving data accuracy.

The process of tracking transaction details, and due information the need for manual data entry and paperwork. This can save time and reduce errors, making the fee due checking process more efficient.

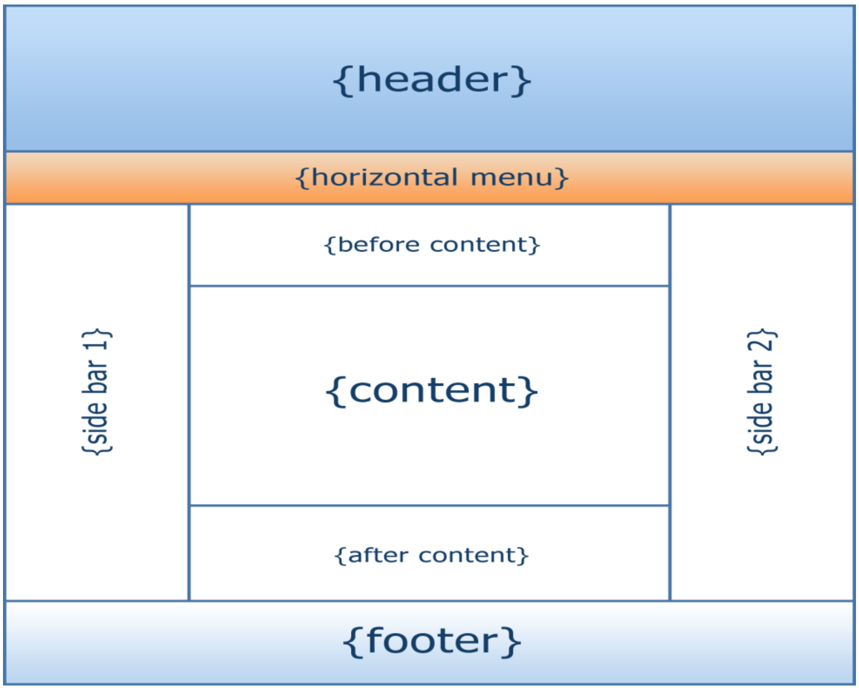
The fee dues checking system provides students with access to information

# Technologies and Tools Used

- FRONTEND : HTML , CSS , BootStrap
- BACKEND : DJANGO
- DATABASE : SQLite
- ENVIRONMENT: Visual Studio Code

## HTML:

means Hypertext Markup Language. This language is used in creating web pages. This language also supports other languages such CSS, PHP, JAVASCRIPT, etc. in creating interactive and responsive pages on the pages. HTML5 is just an updated version of the HTML. It supports new features, new attributes, new HTML elements, full CSS3 support, video and audio, 2D/3D graphics that help users and also help web developers to create new features easily on the website.



## CSS:

CSS is simply referred to as Cascading Style Sheets.CSS is used to define styles for web pages, including the design, layout, and variations in the display for different devices and screen sizes.

The general structure of CSS Basic syntax:

```
selector{property: value}
```

CSS can be used in a separate style sheet or used in the webpage.

## JavaScript:

Javascript is a high-level language which could be used independently or inculcated into the webpage. It can be used to, handle requests and responses and also add dynamic behavior and also store information on a website.

**Syntax:**

```
<script =”javascript” type=” text/javascript” >
        JavaScript code
</script>
```

## Bootstrap:

Bootstrap is the most popular HTML, CSS, and JavaScript framework for developing responsive, mobile-first websites.

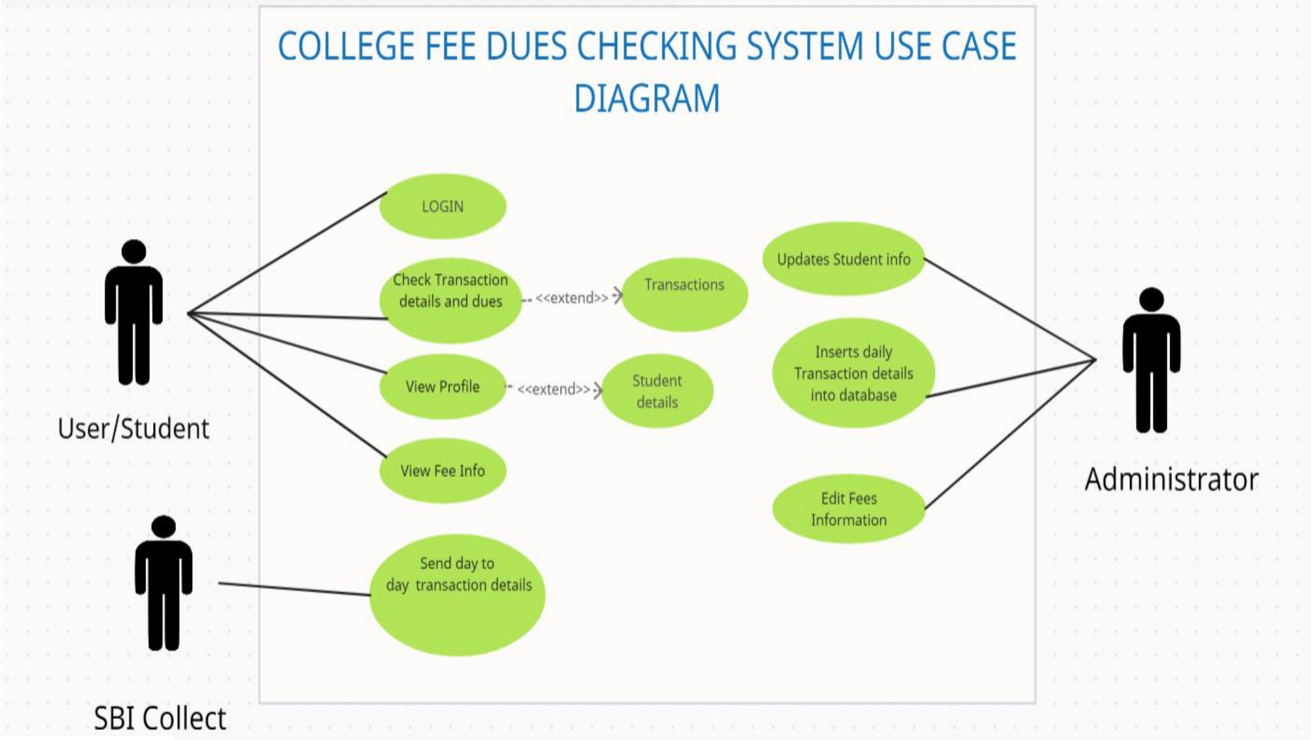
It includes HTML and CSS based design templates for typography, forms, buttons, tables, navigation, modals, image carousels, etc. It also gives you support for JavaScript plugins.

# USE CASE DIAGRAMS

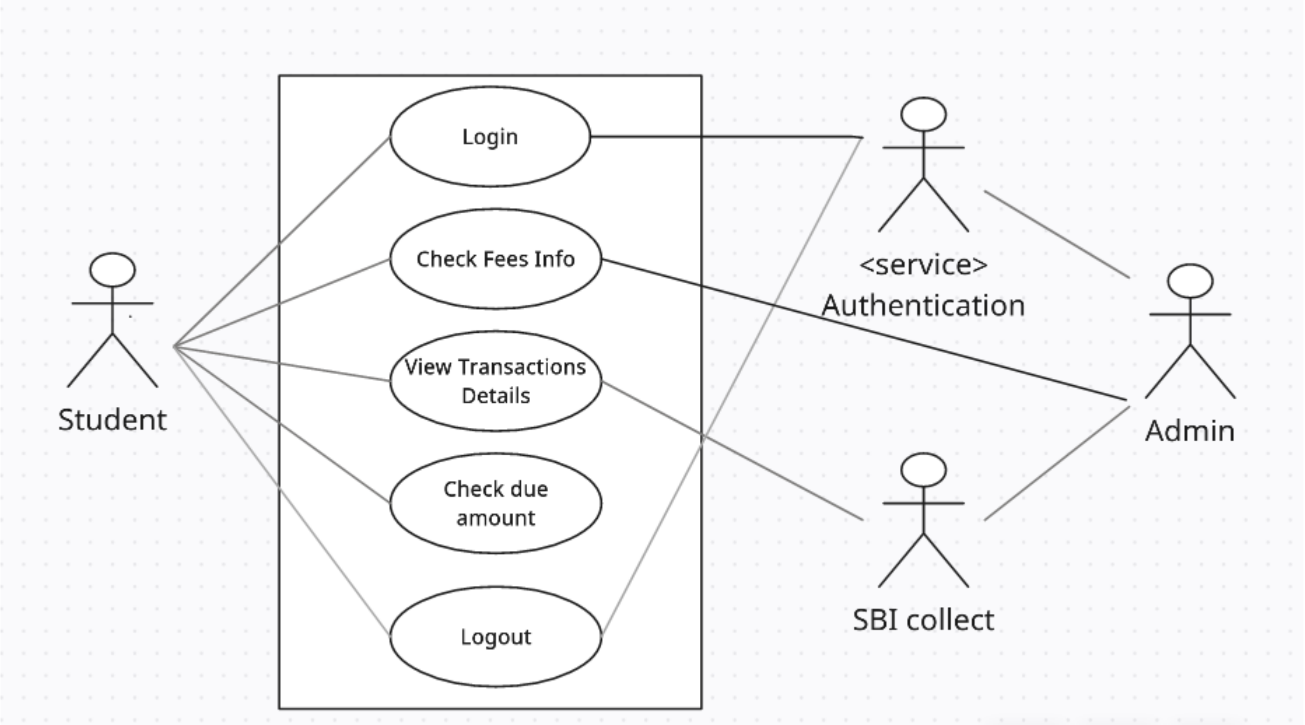
Use case diagrams model behavior within a system and helps the developers understand of what the user require. The stick man represents what's called an actor. Use case diagram can be useful for getting an overall view of the system and clarifying that can do and more importantly what they can't do. Use case diagram consists of use cases and actors and shows the interaction between the use case and actors.

- The purpose is to show the interactions between the use case and actor.
- To represent the system requirements from user's perspective.
- An actor could be the end-user of the system or an external system.

# USE CASE DIAGRAM FOR COLLEGE FEE DUES CHECKING SYSTEM


COLLEGE FEE DUES CHECKING SYSTEM USE CASE DIAGRAM
LOGIN
Check Transaction details and dues
<<extend>>
Transactions
View Profile
<<extend>>
Student details
View Fee Info
Send day to day transaction details
Updates Student info
Inserts daily Transaction details into database
Edit Fees Information
User/Student
SBI Collect
Administrator

# USE CASE DIAGRAM OF COLLEGE FEE DUES CHECKING SYSTEM FOR TOP-LEVEL



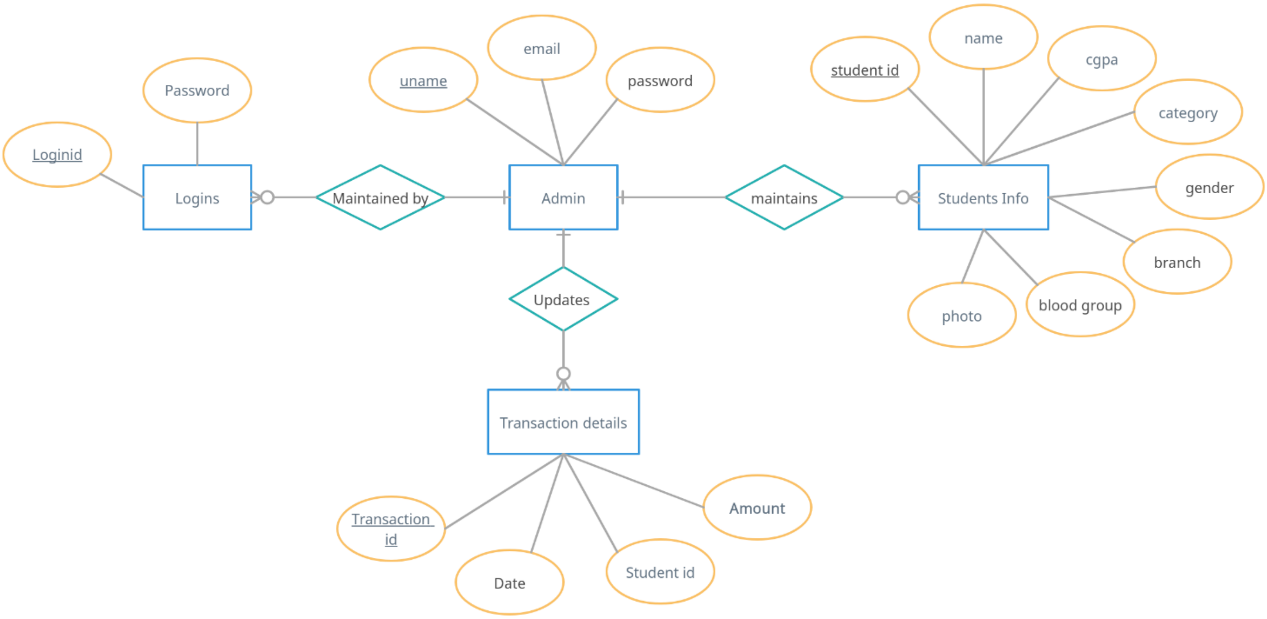
## E-R Diagrams:

The Entity-Relationship (ER) model was originally proposed by Peter in 1976 [Chen76] to unify the network and relational database views. Simply stated the ER model is a conceptual data model that views the real world as entities and relationships. A basic component of the model is the Entity-Relationship diagram which is used to visually represent data objects. Since Chen wrote his paper the model has been extended and today it is commonly used for database design for the database designer, the utility of the ER model is:

- ❖ It maps well to the relational model. The constructs used in the ER model can easily be transformed into relational tables.

- It is simple and easy to understand with a minimum of training. Therefore, the model can be used by the database designer to communicate the design to the end user.
- In addition, the model can be used as a design plan by the database developer to implement a data model in specific database management software.

# E-R DIAGRAM



# SOURCE CODE

## Base.html

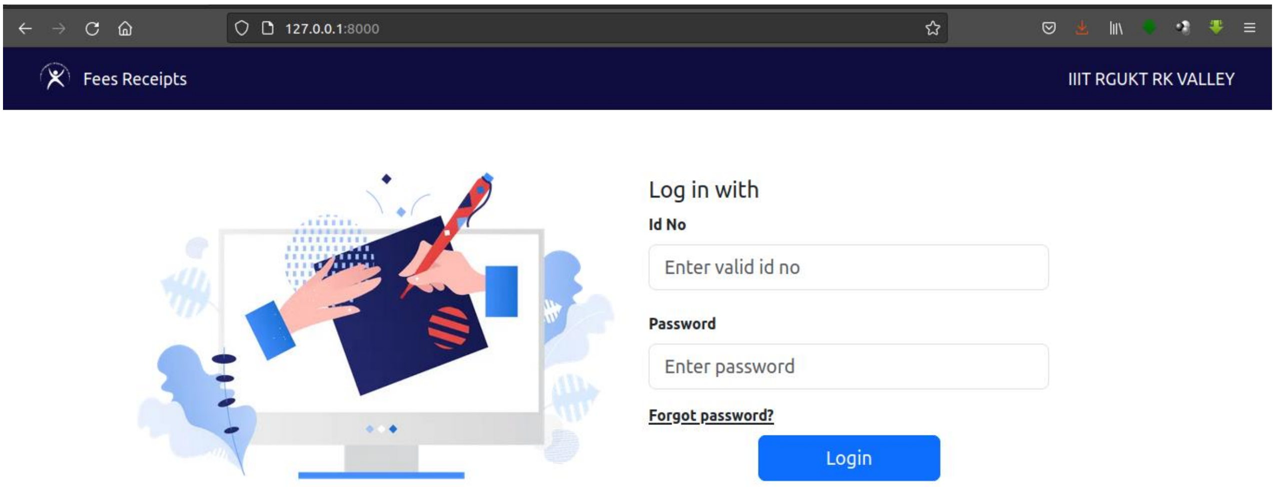
```
<!doctype html>
{%load static%}
<html lang="en">
  <head>
   <meta charset="utf-8">
   <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha3/dist/css/bootstrap.rtl.min.css" integrity="sha384-T5m5WERuXcjgzF8DAb7tRkByEZQGcpraRTinjpywg37AO96WoYN9+hrhDVoM6CaT" crossorigin="anonymous">
   <title> {% block title %}{% endblock title %}</title>
    <style>
      .header{
        width:100%;
        height:65px;
        line-height:65px;
        margin:0;
        background-color:#0F0C40;
        position: fixed;
        padding:0px 40px;
        color: white;
      }
      .header .header-left{
        font-size: large;
      }
      .header .header-right{
        text-align: right;
      }
      .header .header-left img {
        margin-top: 0px;
        margin-right: 0px;
        padding: 0;
        width: 32px;
        height: 32px;
        vertical-align: middle;
        border-radius: 100%;
        margin-top: -0.7%;
        -webkit-filter: brightness(0) invert(1);
        filter: brightness(0) invert(1);
        object-fit: cover;
        margin-right: 10px;
      }
      {% block style %}{% endblock style %}</style>
  </head>
  <body>
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-
alpha3/dist/js/bootstrap.bundle.min.js" integrity="sha384-
ENjdO4Dr2bkBIFxQpeoTz1HIcje39Wm4jDKdf19U8gI4ddQ3GYNS7NTKfA
dVQSZe" crossorigin="anonymous"></script>
    <div class="header">
      <div class="header-left">
        <img src="{% static 'rgukt.png'%}" alt="" />
        Fees Receipts
       <p style="float:right">IIIT RGUKT RK VALLEY</p>
      </div>
      <div class="header-right"></div>
    </div>
    {% block body %}{% endblock body %}
  </body>
</html>
```

## Login.html

```
<!DOCTYPE html>
{% extends 'base.html' %}
{%load static%}
{% block title %}login page{% endblock title %}
{% block style %}
.vh10{
    font-size: 30px;
  }
  .divider:after,
  .divider:before {
    content: "";
    flex: 1;
    height: 1px;
    background: #eee;
  }
  .h-custom {
    height: calc(100% - 73px);
  }
  @media (max-width: 450px) {
```

```
  .h-custom {
    height: 100%;
  }
  }
{% endblock style %}
{% block body %}
<section class="vh-100">
  <div class="container-fluid h-custom fs-">
    <div class="row d-flex justify-content-center align-items-center h-100 fw-bold " >
      <div class="col-md-9 col-lg-6 col-xl-5">
        <img src="https://mdbcdn.b-cdn.net/img/Photos/new-templates/bootstrap-login-form/draw2.webp"
          class="img-fluid" alt="Sample image">
      </div>
      <div class="col-md-8 col-lg-6 col-xl-4 offset-xl-1">
        <form method="post" action="/loginhandle" class="material-form">
          {% csrf_token %}
          <div class="d-flex flex-row align-items-center justify-content-center justify-content-lg-start">
            <p class="lead mb-0 me-0"><b style="font-size:24px !important">Log in with</b></p>
          </div>
          <p class="form-label" style="color:red";>{{msg}}</p>
          <div class="form-outline mb-4">
            <label class="form-label" for="login_uname">Id No</label>
             <input type="text" id="loginid" name="loginid" class="form-control form-control-lg"
              placeholder="Enter valid id no" />
          </div>
          <div class="form-outline mb-3">
            <label class="form-label" for="fpassword">Password</label>
            <input type="password" id="passw" name="passw" class="form-control
form-control-lg"
              placeholder="Enter password" />
          </div>
            <div>
           <a href="/forgot" class="text-body">Forgot password?</a>
          </div>
          <div class="text-center  mt-2 pt-0">
            <button type="submit" class="btn btn-primary btn-lg"
      style="padding-left: 4.5rem; padding-right: 4.5rem;" id="login-submit"
name="login" onClick="check_u_and_p()"><span id="login-
text">Login</span>
          </div>
        </form>
      </div>
    </div>
  </div>
</section>
{% endblock body %}
```

## OUTPUT:



127.0.0.1:8000

Fees Receipts

IIIT RGUKT RK VALLEY

Log in with

Id No

Enter valid id no

Password

Enter password

Forgot password?

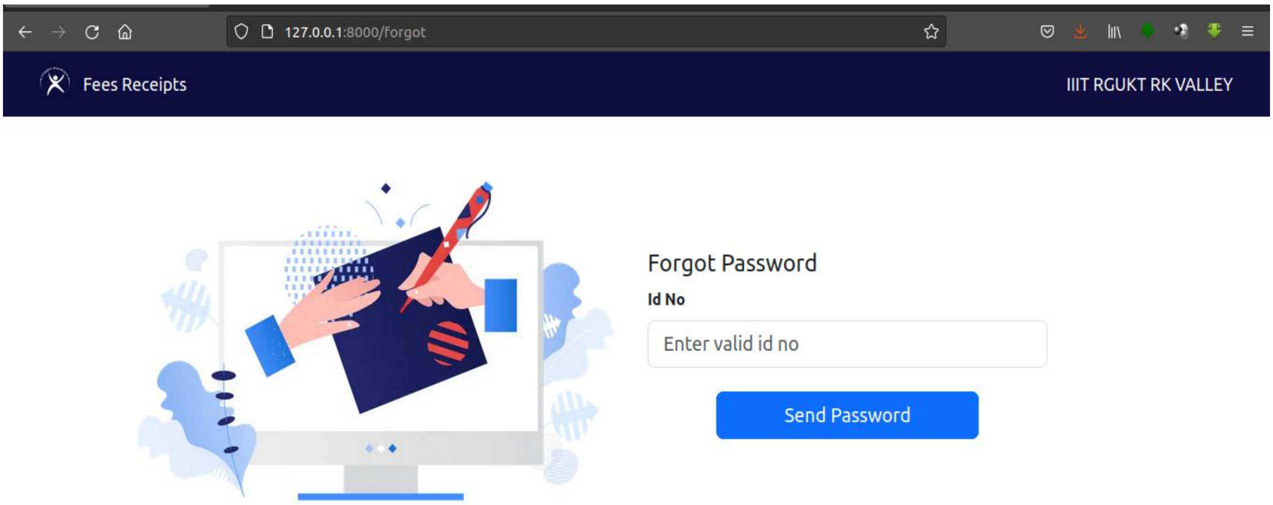
Login

## Forgot.html

```
<!DOCTYPE html>
{% extends 'base.html' %}
{%load static%}
{% block title %}login page{% endblock title %}
{% block style %}
.vh10{
   font-size: 30px;
  }
  .divider:after,
  .divider:before {
    content: "";
    flex: 1;
    height: 1px;
    background: #eee;  }
  .h-custom {
    height: calc(100% - 73px);
  }
  @media (max-width: 450px) {
  .h-custom {
    height: 100%; }
  }
{% endblock style %}
{% block body %}
<section class="vh-100">
  <div class="container-fluid h-custom fs-">
    <div class="row d-flex justify-content-center align-items-center h-100 fw-bold " >
      <div class="col-md-9 col-lg-6 col-xl-5">
        <img src="https://mdbcdn.b-cdn.net/img/Photos/new-templates/bootstrap-login-form/draw2.webp"
          class="img-fluid" alt="Sample image">
      </div>
      <div class="col-md-8 col-lg-6 col-xl-4 offset-xl-1">
         <form method="post" action="/forgothandle" class="material-form">
          {% csrf_token %}


          <div class="d-flex flex-row align-items-center justify-content-center justify-content-lg-start">
            <p class="lead mb-0 me-0"><b style="font-size:24px !important">Forgot Password</b></p>
          </div>
          <p class="form-label" style="color:red";>{{msg}}</p>
          <div class="form-outline mb-4">
```

```
            <label class="form-label" for="login_uname">Id No</label>

            <input type="text" id="loginid" name="loginid" class="form-control
form-control-lg"

              placeholder="Enter valid id no" />

          </div>

          <div class="text-center  mt-2 pt-0">

            <button type="submit" class="btn btn-primary btn-lg"

      style="padding-left: 4.5rem; padding-right: 4.5rem;" id="login-submit"
name="login" onClick="check_u_and_p()"><span id="login-text">Send
Password</span>

            </div>

        </form>

      </div>

    </div>

  </div>

</section>

{% endblock body %}
```

## OUTPUT:



127.0.0.1:8000/forgot

Forgot Password

Id No

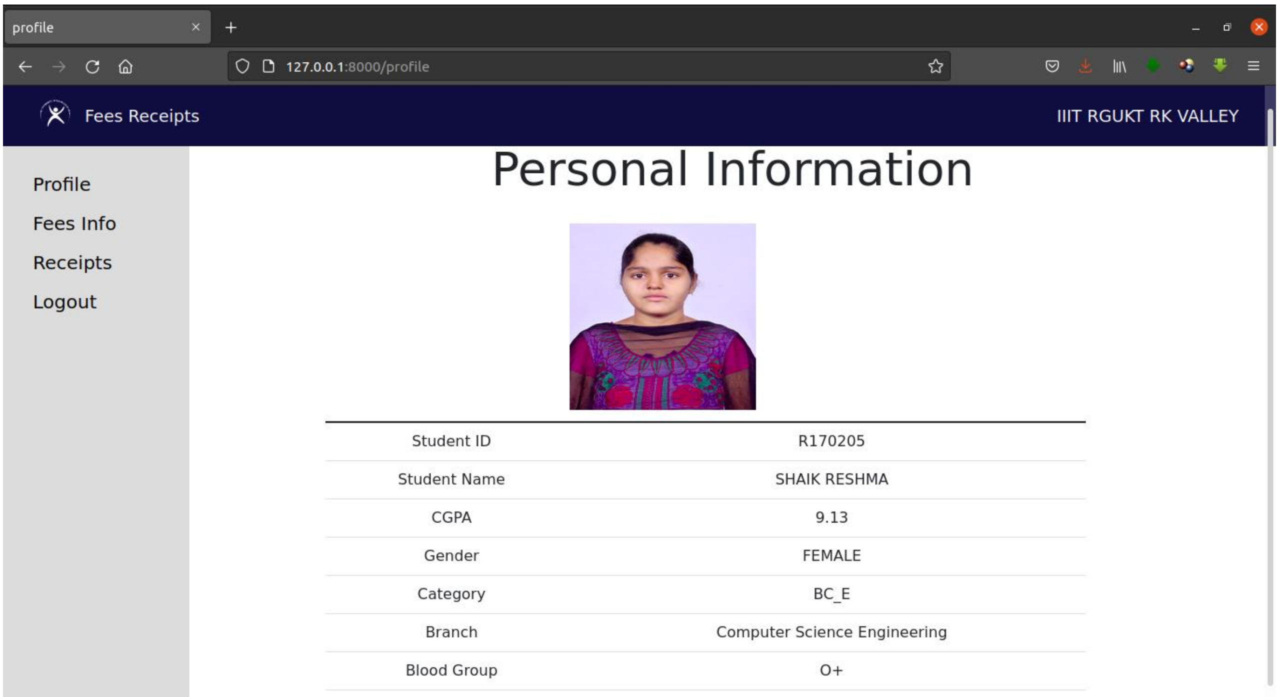
Enter valid id no

## Profile.html

```
<!DOCTYPE html>

{% extends 'navbar.html' %}

{%load static %}

{% static "media" as baseUrl %}

{% block title %}profile{% endblock title %}

{% block style %}

.body-con .sidenav #info{

    backgroundcolor:red;

    color:black;

}

.body-con .pic{

   margin-top:2.5%;

   margin-left:35%;

}

.table{

    margin-top:1%;

    margin-left:12.5%;

    text-align:center;

    width:70%;

    align:center;

}


{% endblock style %}

{% block main %}
```

```
<h1 style="text-align: center; font-size:50px;">Personal Information</h1>
{% for d in dlist %}
<div class="pic" >
   <img src= "{{d.img}}" alt="" width=200px height=200px >
</div>
<div class="tb">
<table class="table" >
      <tbody class="table-group-divider">
         <tr>
            <td>Student ID</td>
            <td>{{d.sid}}</td>
         </tr>
         <tr>
            <td>Student Name</td>
            <td>{{d.name}}</td>
         </tr>
         <tr>
            <td>CGPA</td>
            <td>{{d.cgpa}}</td>
         </tr>
         <tr>
            <td>Gender</td>
            <td>{{d.gen}}</td>
         </tr>
         <tr>
            <td>Category</td>
            <td>{{d.ctg}}</td>
            </tr>
            <tr>
                <td>Branch</td>
                <td>{{d.bran}}</td>
            </tr>
            <tr>
                <td>Blood Group</td>
                <td>{{d.bgrp}}</td>
            </tr>

            {% endfor %}
          </tbody>
</table>
</div>
{% endblock main %}
```

## OUTPUT:



Personal Information

|  |  |
|---|---|
| Student ID | R170205 |
| Student Name | SHAIK RESHMA |
| CGPA | 9.13 |
| Gender | FEMALE |
| Category | BC_E |
| Branch | Computer Science Engineering |
| Blood Group | O+ |

## Navabar.html

```
<!DOCTYPE html>
{% load static %}
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha3/dist/css/bootstrap.rtl.min.css" integrity="sha384-T5m5WERuXcjgzF8DAb7tRkByEZQGcpraRTinjpywg37AO96WoYN9+hrhDVoM6CaT" crossorigin="anonymous" />
    <title>
      {% block title %}
      {% endblock %}
    </title>
   <style>
     * {
       margin: 0px;
       padding: 0px;
       box-sizing: border-box;
       font-family: 'Poppins', sans-serif;
     }
     .header{
       width:100%;
       height:65px;
       line-height:65px;
       margin:0;
       background-color:#0F0C40;
       position: fixed;
       padding:0px 40px;
       color: white;
     }
     .header .header-left{
       font-size: large;
     }
     .header .header-right{
       text-align: right;
     }
     .header .header-left img {
       margin-top: 0px;
       margin-right: 0px;
       padding: 0;
       width: 32px;
       height: 32px;
       vertical-align: middle;
       border-radius: 100%;
       margin-top: -0.7%;
       -webkit-filter: brightness(0) invert(1);
       filter: brightness(0) invert(1);
       object-fit: cover;
       margin-right: 10px;
     }
     .body-con {
       width: 100%;
       height: auto;
       padding: 0;
       top: 80px;
       position: relative;
      box-sizing: border-box;
      overflow:hidden;
     }
     .sidenav {
       height:100%;
       margin-top:65px;
       width: 200px;
       position: fixed;
       z-index: 1;
       top: 0;
       left: 0;
       background-color:#DCDCDC;
       overflow-x: hidden;
       padding-top: 20px;
     }
     .sidenav a {
       padding: 6px 6px 6px 32px;
       text-decoration: none;
       font-size: 20px;
       color: black;
       display: block;
     }

     .sidenav a:hover {
```

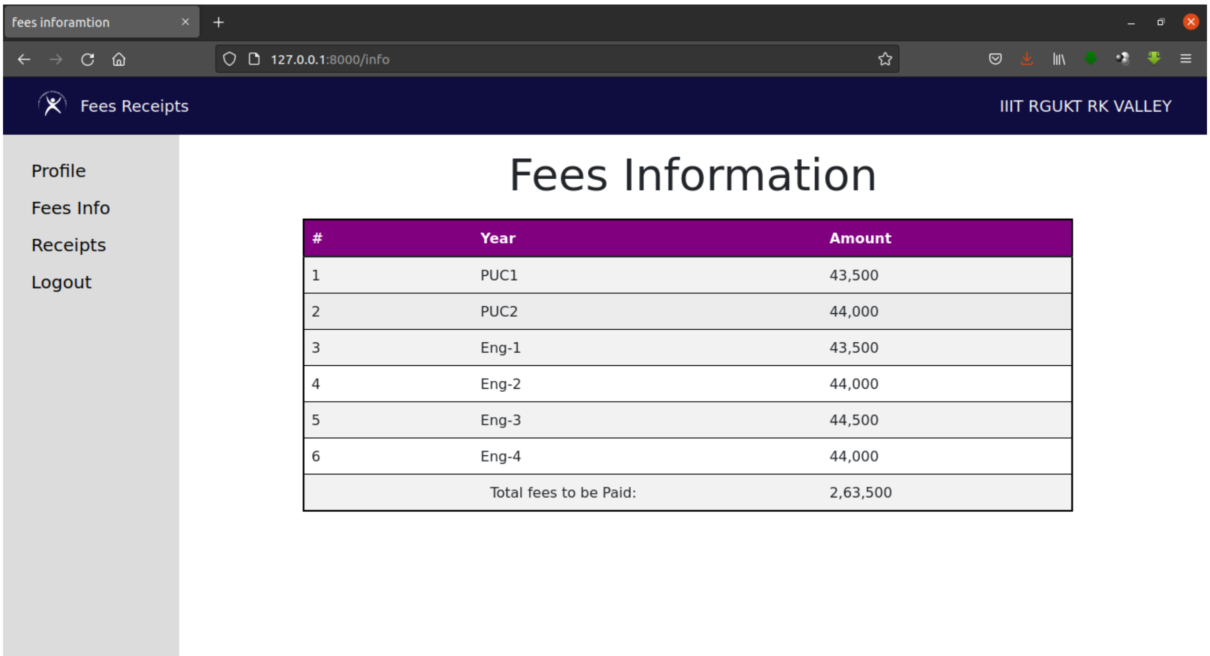
```
          color: #f1f1f1;
        }
        .main {
          margin-left: 200px; /* Same as the width of the sidenav */
        }
        {% block style %}{% endblock style %}
      </style>
    </head>
    <body>
      <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha3/dist/js/bootstrap.bundle.min.js" integrity="sha384-ENjdO4Dr2bkBIFxQpeoTz1HIcje39Wm4jDKdf19U8gI4ddQ3GYNS7NTKfAdVQSZe" crossorigin="anonymous"></script>
      <div class="header">
          <div class="header-left">
            <img src="{% static 'rgukt.png'%}" alt="" />
            Fees Receipts
            <p style="float:right">IIIT RGUKT RK VALLEY</p>
          </div>
          <div class="header-right"></div>
        </div>
      <div class="body-con">
        <div class="sidenav">
          <a class="profile" href="profile">Profile</a>
          <a id="info" href="info">Fees Info</a>
          <a class="dues" href="dues">Receipts</a>
          <a class="login" href="login">Logout</a>
        </div>
        <div class="main">
         {% block main %}{% endblock main %}
        </div>
      </div>
    </body>
  </html>
```

## Info.html

```
<!DOCTYPE html>
{% extends 'navbar.html' %}
{%load static%}
{% block title %}fees inforamtion{% endblock title %}
{% block style %}
.body-con .sidenav #info{
    backgroundcolor:red;
    color:black;
}
.table{
    width:75%;
    align:center;
    margin-left:12%;
    margin-top:20px;
    border:2px solid black;
}
thead{
    color:white;
}
```

```
{% endblock style %}
{% block main %}
<h1 style="text-align: center; font-size:50px;">Fees Information</h1>
<table class="table table-striped table-hover">
    <thead >
        <tr style="background-color:Purple;">
          <th scope="col">#</th>
          <th scope="col">Year</th>
          <th scope="col">Amount </th>
        </tr>
    </thead>
        <tbody class="table-group-divider">
            <tr>
              <td>1</td>
              <td>PUC1</td>
              <td>43,500</td>
            </tr>
            <tr>
                <td>2</td>
                <td>PUC2</td>
                <td>44,000</td>
            </tr>
            <tr>
                <td>3</td>
                <td>Eng-1</td>
                <td>43,500</td>
            </tr>
            <tr>
                <td>4</td>
                <td>Eng-2</td>
                <td>44,000</td>
            </tr>
            <tr>
                <td>5</td>
                <td>Eng-3</td>
                <td>44,500</td>
            </tr>
            <tr>
                <td>6</td>
                <td>Eng-4</td>
                <td>44,000</td>
            </tr>
            <tr>
              <td colspan="2" align="center">Total fees to be Paid:</td>
              <td>2,63,500</td>
            </tr>
          </tbody>
</table>
{% endblock main %}
```

## OUTPUT:



Fees Information

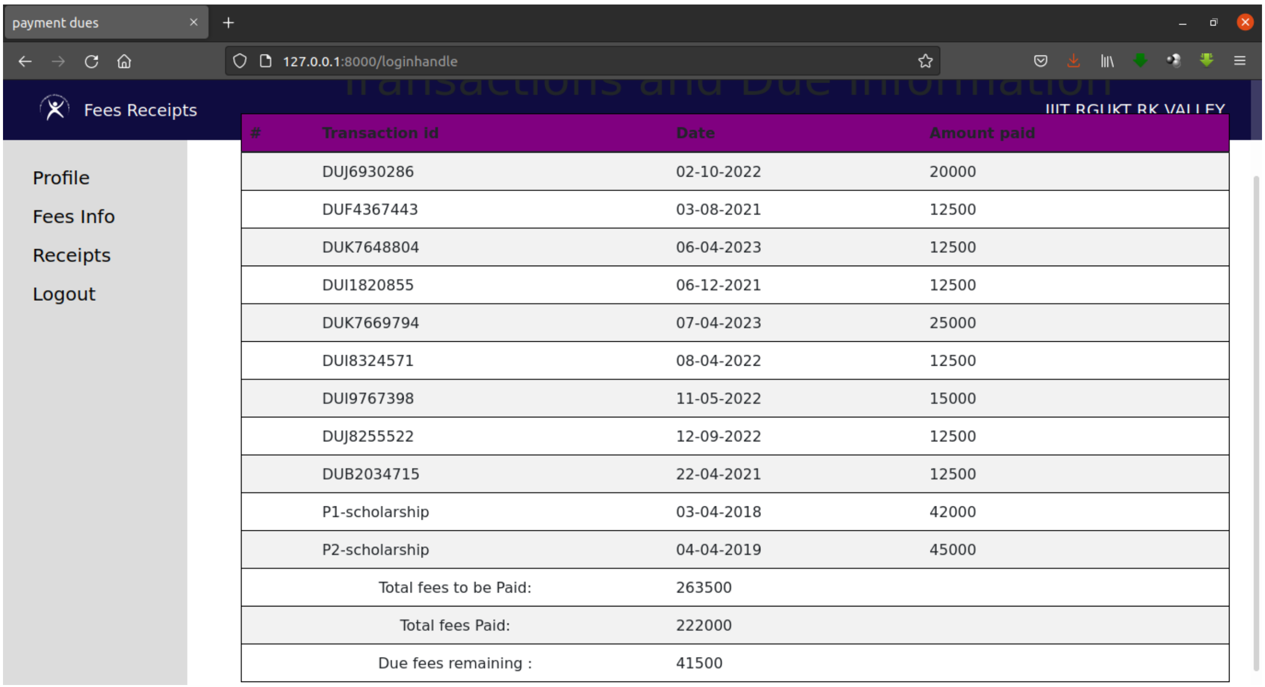
| # | Year | Amount |
|---|---|---|
| 1 | PUC1 | 43,500 |
| 2 | PUC2 | 44,000 |
| 3 | Eng-1 | 43,500 |
| 4 | Eng-2 | 44,000 |
| 5 | Eng-3 | 44,500 |
| 6 | Eng-4 | 44,000 |
| | Total fees to be Paid: | 2,63,500 |

## Dues.html

```
<!DOCTYPE html>
{% extends 'navbar.html' %}
{%load static%}
{% block title %}payment dues{% endblock title %}
{% block style %}
.body-con .sidenav #info{
   backgroundcolor:red;
   color:black;
}
.table{
   width:92%;
   margin-left:5%;
   align:center;
   border: 1px solid black;}
```

```
{% endblock style %}
{% block main %}
<h1 style="text-align: center; font-size:50px;">Transactions and Due
Information</h1>
<table class="table table-striped table-hover">
    <thead >
        <tr style="background-color:Purple;">
          <th scope="col">#</th>
          <th scope="col">Transaction id</th>
          <th scope="col">Date</th>
          <th scope="col">Amount paid</th>
        </tr>
    </thead>
        <tbody class="table-group-divider">
            {%for t in tlist%}
            <tr>
              <td>{{t.tid}}</td>
              <td>{{t.date}}</td>
              <td>{{t.amount}}</td>
            </tr>
            {% endfor %}
            <tr>
              <td colspan="2" align="center">Total fees to be Paid:</td>
              <td colspan="2">263500</td>
            </tr>
            <tr>
                <td colspan="2" align="center">Total fees Paid:</td>
                <td colspan="2">{{paid}}</td>
            </tr>
            <tr>
                <td colspan="2" align="center">Due fees remaining :</td>
                <td colspan="2">{{bal}}</td>
            </tr>
          </tbody>
</table>
{% endblock main %}
```

## OUTPUT:



| # | Transaction id | Date | Amount paid |
|---|---|---|---|
| | DUJ6930286 | 02-10-2022 | 20000 |
| | DUF4367443 | 03-08-2021 | 12500 |
| | DUK7648804 | 06-04-2023 | 12500 |
| | DUI1820855 | 06-12-2021 | 12500 |
| | DUK7669794 | 07-04-2023 | 25000 |
| | DUI8324571 | 08-04-2022 | 12500 |
| | DUI9767398 | 11-05-2022 | 15000 |
| | DUJ8255522 | 12-09-2022 | 12500 |
| | DUB2034715 | 22-04-2021 | 12500 |
| | P1-scholarship | 03-04-2018 | 42000 |
| | P2-scholarship | 04-04-2019 | 45000 |
| Total fees to be Paid: | | 263500 | |
| Total fees Paid: | | 222000 | |
| Due fees remaining : | | 41500 | |

## Project: urls.py

```
from django.contrib import admin
from django.urls import path,include
from django.conf import settings
from django.conf.urls.static import static
urlpatterns = [
    path('admin/', admin.site.urls),
    path('',include("home.urls"))
]+static(settings.MEDIA_URL,document_root=settings.MEDIA_ROOT)
```

## App: urls.py

```
from django.contrib import admin
from django.urls import path
from home import views
urlpatterns = [
    path("",views.index,name="index"),
    path("loginhandle",views.loginhandle,name="loginhandle"),
    path("profile",views.profile,name="profile"),
    path("dues",views.dues,name="dues"),
    path("info",views.info,name="info"),
    path("login",views.logout,name="logout"),
    path("forgot",views.forgot,name="forgot"),
    path("forgothandle",views.forgothandle,name="forgothandle"),
]
```

## Models.py

```
from django.db import models

#Create your models here.
class logins(models.Model):
    loginid=models.CharField(max_length=7)
    passw=models.CharField(max_length=7)

class transactions(models.Model):
    sid=models.CharField(max_length=7)
    tid=models.CharField(max_length=15)
    date=models.CharField(max_length=15)
    amount=models.IntegerField(default=0)

class stu_details(models.Model):
    sid=models.CharField(max_length=7)
    name=models.CharField(max_length=50)
    cgpa=models.CharField(max_length=20)
    ctg=models.CharField(max_length=20)
    gen=models.CharField(max_length=7)
    bran=models.CharField(max_length=50)
    bgrp=models.CharField(max_length=5)
    img=models.ImageField(upload_to="media")
```

## Views.py

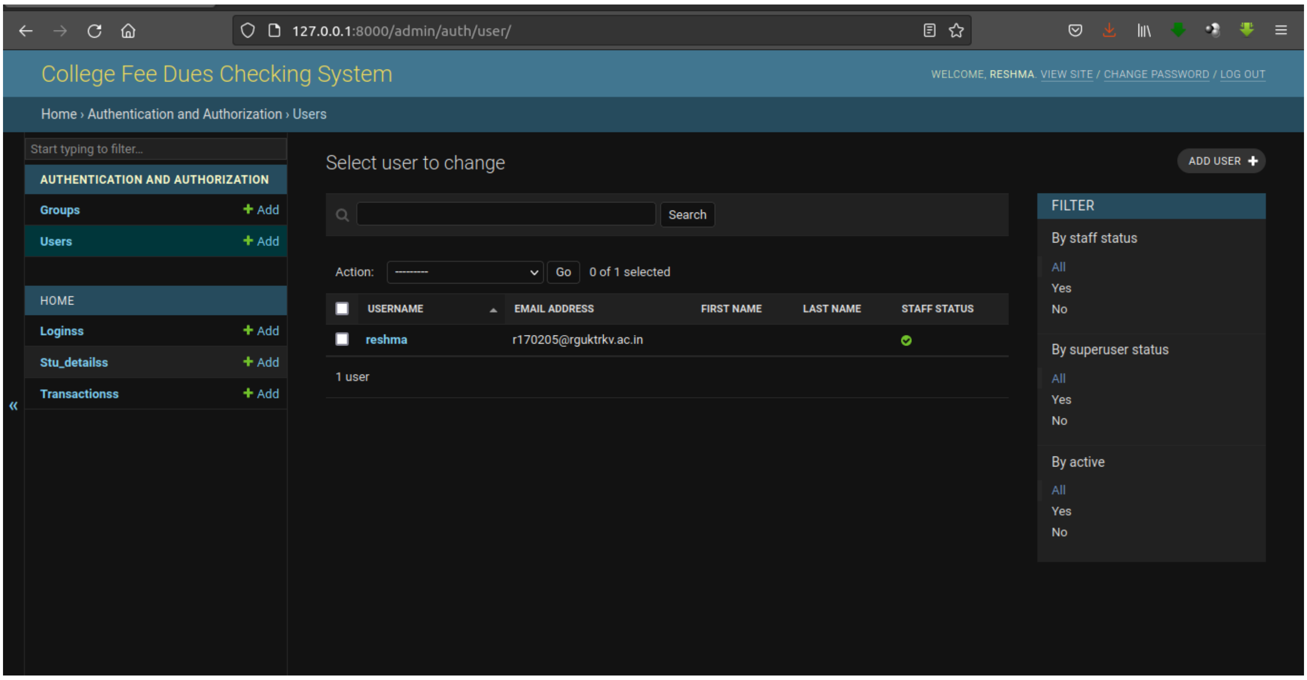
```
from django.shortcuts import render,HttpResponse
from home.models import logins,transactions,stu_details
from django.core.mail import send_mail
# Create your views here.
def index(request):
        return render(request,"login.html")


def loginhandle(request):
    if request.method=='POST':
        lid=request.POST['loginid']
        pas=request.POST['passw']
        data=logins.objects.filter(loginid=lid).values()
        if(data[0]['passw']==pas):
            print("true")
            with open('id.txt','w') as file:
                file.write(lid)
            m=transactions.objects.filter(sid=lid).values()
            x=0
            k=1
            l=[]
            for i in m:
                l.append(k)
                k=k+1
                x=x+i['amount']
            return render(request,"dues.html",{"tlist":m,"paid":x,"bal":263500-x,"c":l})
        else:
                return render(request,"login.html",{"msg":"Invalid credentials! Enter valid details"})


def profile(request):
    with open('id.txt', 'r') as file:
        id= file.read()
        if(len(id)!=0):
            m=stu_details.objects.get(sid=id)
            m.img="media/"+id+".jpg"
            m.save()
            m=stu_details.objects.filter(sid=id).values()
            print(m,m[0]['img'])
            return render(request,"profile.html",{"dlist":m})
def dues(request):
    with open('id.txt', 'r') as file:
        id= file.read()
        print(id)
        if(len(id)!=0):
            from home.models import transactions
            m=transactions.objects.filter(sid=id).values()
            x=0
            for i in m:
                x=x+i['amount']
            print(id,m)
            return render(request,"dues.html",{"tlist":m,"paid":x,"bal":263500-x,},)
def info(request):
    return render(request,"info.html")
def logout(request):
    with open('id.txt','w') as file:
        file.write('')
        return render(request,"login.html")
def forgot(request):
    return render(request,'forgot.html')
def forgothandle(request):
    if request.method=='POST':
        lid=request.POST['loginid']
        usr=logins.objects.get(loginid=lid)
        """send_mail(
            "Login credentials",
            "Id : "+lid+"\nPassword : "+usr.passw,
            "printingpressapp@gmail.com",
            [lid+"@rguktrkv.ac.in"],
        )"""
        return render(request,"login.html")
```

## **xlsave.py(**To save daily transaction details into database )

```
import openpyxl
from models import transactions
wb=openpyxl.load_workbook('02-10-2022.xlsx')
sobj=wb.active
for i in sobj:
    l=[]
```

```
for j in i:
    l.append(j.value)
print(l)
m=transactions(sid=l[0],tid=l[1],date=l[2],amount=l[3])
m.save()
```

## Admin Panel and databases:



127.0.0.1:8000/admin/auth/user/

College Fee Dues Checking System

WELCOME, RESHMA. VIEW SITE / CHANGE PASSWORD / LOG OUT

Home › Authentication and Authorization › Users

Start typing to filter...

AUTHENTICATION AND AUTHORIZATION

| | |
|---|---|
| Groups | + Add |
| Users | + Add |

HOME

| | |
|---|---|
| Loginss | + Add |
| Stu_detailss | + Add |
| Transactionss | + Add |

Select user to change

ADD USER +

Search

Action: --------- Go 0 of 1 selected

| ☐ | USERNAME | EMAIL ADDRESS | FIRST NAME | LAST NAME | STAFF STATUS |
|---|---|---|---|---|---|
| ☐ | reshma | r170205@rguktrkv.ac.in | | | ✓ |

1 user

FILTER

By staff status
- All
- Yes
- No

By superuser status
- All
- Yes
- No

By active
- All
- Yes
- No

# CONCLUSION

After careful observation, it has come to my conclusion that everything is becoming digital and has undeniably become an important part of our society. The world wide web is and will have a large part in our daily lives.

This web application provides students to view fees information, transaction details previously made by them and due details at one place. College fee dues checking system being web-based application needs to be thoroughly tested before implementation to find any security gaps.

# FUTURE ENHANCEMENTS

- To provide students with online payment options such as payment gateways, online banking, or mobile payment options to make it easier for them to pay their fee dues.

- To ensure that the fee dues checking system provides real-time updates on any changes to fee amounts or due dates. This can help students stay informed and avoid any surprises.

- Integration with student accounts i.e., Students can view their fee dues alongside other important information such as their class schedules and grades.

- Automated notifications, setting up an automated system to send notifications to students when their fee dues are approaching or past the due date

- Allowing students to track their payment history and view any outstanding fee dues. This can help them stay on top of their payments and avoid any late fees.

# REFERENCES

## HTML & CSS

- **HTML- W3schools**
- **CSS - W3Schools**

## BOOTSTRAP

- **Bootstrap - W3schools**
- **https://getbootstrap.com/**

## DJANGO

- **Django Documentation**
- **https://www.youtube.com/playlist?list=PLu0W_9lII9ah7DDtYtflgwMwpT3xmjXY9**