

# **ONLINE FEE MANAGEMENT SYSTEM**

A mini project report submitted  
for partial fulfilment for the award of the degree

**B-Tech in computer science & engineering**

**SUBMITTED BY:**

|                            |                      |
|----------------------------|----------------------|
| <b>Yashwanth bharadwaj</b> | <b>-319129510028</b> |
| B.Priyanka                 | -319129510009        |
| N.Rani                     | -319129510030        |
| S.Swapna                   | -319129510040        |

*Under the Esteemed Guidance of*

**Mrs. CHOKKAKULLA DEVI ,M.Tech,(Ph.D)**

**Asst Prof. Department of CSE , WISTM.**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING WELLFARE  
INSTITUTE OF SCIENCE TECHNOLOGY AND MANAGEMENT**

**Approved AICTE ,DELHI&affiliated to  
Andhra University, Pendurthi(M),Visakhapatnam-531173**

# **WELLFARE INSTITUTE OF SCIENCE TECHNOLOGY AND MANAGEMENT**

**Approved AICTE ,DELHI&affiliated to  
Andhra University ,Pendurthi(M),Visakhapatnam-531173**



## **CERTIFICATE**

This is to certify that the mini project report entitled  
**"Online Fee Management System"**

### **SUBMITTED BY:**

|                            |                      |
|----------------------------|----------------------|
| <b>Yashwanth bharadwaj</b> | <b>-319129510028</b> |
| B.Priyanka                 | -319129510009        |
| N.Rani                     | -319129510030        |
| S.Swapna                   | -319129510040        |

For partial fulfillments for the award of the degree Bachelor of Technology in computer science and Engineering is a bonafide project work carried out by him .This work is not submitted any University for the award of any degree.

### **INTERNAL GUIDE**

Mrs. CH DEVI,  
**Assistant Professor,**  
Department of CSE ,  
WISTM , ENGG.COLLEGE

### **HEAD OF THE DEPARTMENT**

Mrs.K.V.LAKSHMI,M.TECH  
**Assistant Professor,**  
Department of CSE ,  
WISTM , ENGG.COLLEGE

### **EXTERNAL EXAMINER**

## **ACKNOWLEDGEMENT**

The satisfaction that accompanies the successful completion of any task would be incomplete without the mention of people who made it possible and whose constant guidance and encouragement crown all the effort with success , Technology and Management , for this kind cooperation that gives me boundless pleasure to avail this opportunity to express my deep sense of gratitude and while hearted thanks to Head of Department **Mrs.K.V.LAKSHMI, Assistant Professor, Mrs. CHOKKAKULLA DEVI Asst Prof. Department of CSE , Assistant Professor** ,department of CSE and all the faculty of computer science department for their valuable suggestions and co-operation during the development of this project . I am really grateful to my parents and all my friends for their constructive suggestions to do this project in effective manner .

### **SUBMITTED BY:**

|                            |                      |
|----------------------------|----------------------|
| <b>Yashwanth bharadwaj</b> | <b>-319129510028</b> |
| B.Priyanka                 | -319129510009        |
| N.Rani                     | -319129510030        |
| S.Swapna                   | -319129510040        |

## **DECLARATION**

I here declare that the project entitled “Online fee management system” has been done under the guidance of **Mrs. CHOKKAKULLA DEVI,M.Tech,(Ph.D) Assistant Professor, Department of CSE** and is dissertation of my own work except where specifically ask to the contrary and is submitted to the department of computer science and Engineering, welfare Insistute of science technology and management for the partial fulfilment of the requirement for the award of B.Tech degree.

### **SUBMITTED BY:**

|                            |                      |
|----------------------------|----------------------|
| <b>Yashwanth bharadwaj</b> | <b>-319129510028</b> |
| B.Priyanka                 | -319129510009        |
| N.Rani                     | -319129510030        |
| S.Swapna                   | -319129510040        |

## **TABLE OF CONTENTS**

|                               |           |
|-------------------------------|-----------|
| <b>1. ABSTRACT</b>            | <b>6</b>  |
| <b>2. INTRODUCTION</b>        | <b>7</b>  |
| <b>3. SYSTEM ANALYSIS</b>     | <b>8</b>  |
| 3.1 system definition         |           |
| 3.2 proposed system           |           |
| <b>4. SYSTEM REQUIREMENTS</b> | <b>9</b>  |
| <b>5. TECHNOLOGIES STACK</b>  | <b>11</b> |
| 6.1 HTML and CSS              |           |
| 6.2 Javascript                |           |
| 6.3 Bootstrap                 |           |
| 6.4 Python                    |           |
| 6.5 Django                    |           |
| <b>7. CODING</b>              | <b>38</b> |
| <b>08. SCREENS</b>            | <b>87</b> |
| <b>09. CONCLUSION</b>         | <b>90</b> |

## **INTRODUCTION / ABSTRACT**

This project “Fee Management System” is a desktop system enables efficient storage of student records to properly manage the fee records of the students. And it also generate messages for due balances of students fee. The system is designed for fee management of a college administration department. It makes searching records easier and faster

### **OBJECTIVE:**

- To reduce paperwork
- To make storage of information more efficient and secure.
- To have a friendly interface.
- To operate it easily and with minimum experience.
- To save time and energy of the admin.

### **SCOPE OF THE SYSTEM:**

- Specifically designed for an individual college.
- It is based on desktop application.
- Including fees other than academic like bus fees and etc.
- There are only limited number of modules for fees management.

### **EXISTING SYSTEM AND DRAWBACK:**

- In the existing system, colleges have to manually maintain information regarding Fees deposited by the students.

- College management system are complex and time consuming to maintain fees of students by that very difficult.
- It is not properly capable to manage the student records with their fee details at a single place.
- Managing collection of student fees, issuing fee receipts and fee register updation is a laborious manual process, leading to data inaccuracy and / or reconciliation.
- Preparing receipts manually everyday needs additional clerical staff.
- To generate due fees report is required a complete manual procedure, which involves a lot of time and clerical staff manpower.
- Re-entry of fees receipts in accounting software separately leads to double manpower cost and time

## **PROPOSED SYSTEM & ADVANTAGES:**

- The fees management System is a desktop system aimed to maintaining students records and their fees details.
- It also generate records like I.e. Feed Paid, dues, and etc.
- The system requires small amount of time to generate reports needed to manage the fees of the student
- Managing collection of school fees, issuing fee receipts and fee register updation is done with the help of software resulting in highly accurate data.
- Software provides facility to print receipts automising office work.
- This Software provides facility to generate due fees report easily and at any point of time.

## **SOFTWARE DESCRIPTION PROJECT DESIGN:**

Software design is an interactive process through which requirements are translated into a ‘Blue Print’ for constructing the software. The design is represented at high level of abstraction, a level that can be directly translated to specific data, functional and behavioural requirements.. Preliminary design is concerned with the transformation of requirements into data and software architecture. Detailed design focuses on refinements to the architectural representation that lead to detailed data structure and algorithmic representation for software.

## **SYSTEM SPECIFICATION:**

A System Requirements Specification (SRS) (also known as a Software Requirements Specification) is a document or set of documentation that describes the features and behaviour of a system or software application. It includes a variety of elements (see below) that attempts to define the intended functionality required by the customer to satisfy their different users. Whenever you purchase software or hardware for your computer, you should first make sure your computer supports the system requirements. These are the necessary specifications your computer must have in order to use the software or hardware.

## **TECHNOLOGY USED**

### **SYSTEM SPECIFICATIONS:**

#### **HARDWARE SPECIFICATION-**

- Processor: Dual core
- RAM: 1 GB
- ROM: 2 GB

#### **SOFTWARE SPECIFICATION-**

- Technologis: HTML, CSS, java script
- Database: MySql
- Language: PYTHON ,Django

### **BRIEF OVERVIEW OF THE TECHNOLOGY:**

#### **Front end: HTML, CSS, JavaScript**

**HTML:** HTML is used to create and save web document. E.g. Notepad/Notepad++

**CSS :** (Cascading Style Sheets) Create attractive Layout

**Bootstrap :** responsive design mobile friendly site

**JavaScript:** it is a programming language, commonly use with web browsers.

**Back end:** PYTHON, MySQL,DJANGO

## HYPertext Markup Language (HTML)

HTML is an application of the Standard Generalised Markup Language (SGML), which was approved as an international in the year 1986. SGML provides a way to encode hyper documents so they can be interchanged. SGML is also a Meta language for formally describing document markup system. In fact HTML uses SGML to define a language that describes a WWW hyper document's structure and inter connectivity. Following the rigors of SGML, TBL bore HTML to the world in 1990. It does not show any compilation errors and also it will be highly executed through the browser. It is the set of markup symbols or codes inserted in a file intended for display on World Wide Web browser page.

**<!doctype html>**: The <!doctype> is not an HTML tag. It is an instruction to the web browser about what version of html the pages is written in. This page is written in HTML5 as opposed to say HTML 4.01.

**<html>**: The <html> element is the root element of an HTML page.

**<head>**: The <head> element is a container for meta information about the document and it is placed between the <html> and <body> tags.

**<title>**: The <title> tag is required in all HTML documents and it specifies a title for the document.

**<body>**: The <body> element contains the visible page content.

**<h1>**: The <h1> element defines a large heading.

**<h6>**: The <h6> element defines a small heading.

**<p>**: The <p> element defines a paragraph.

**<a>**: The <a> element defines the HTML links. The link's destination is specified in the href attribute.

**<img>**: The <img> element defines the HTML images. The source file (src), alternative text (alt), width, and height are provided as attributes.

**<button>**: The <button> element defines a clickable button, inside a button you can put content.

**<br>**: The <br> element defines the line break. This tag is an empty tag, which means that it has no end tag.

**<!-- -->**: The <!-- --> element defines the comment tag.

**<table>**: The <table> element defines the HTML tables.

**<tr>**: The <tr> element defines the table row.

**<th>**: The <th> element defines the table header. By default, table headings are bold and centered.

**<td>**: The <td> element defines the table data/cell.

**<div>**: The <div> element defines the block level element. A block-level element always starts on a new line and takes up the full width available.

**<form>**: It defines a form that is used to collect user input.

**<label>**: The <label> element represents a caption for an item in a user interface.

**<input>**: The <input> element is the most important form element.

The <input> element can be displayed in several ways, depending on the type attribute.

**<ul>**: The <ul> element to define an unordered list.

**<ol>**: The <ol> element to define an ordered list.

**<li>**: The <li> element to define a list item

## **CSS [Cascading Style Sheet]:**

Cascading Style Sheets, fondly referred to as CSS, is a simple design language intended to simplify the process of making web pages presentable.

CSS handles the look and feel part of a web page. Using CSS, you can control the colour of the text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colours are used, layout designs, variations in display for different devices and screen sizes as well as a variety of other effects.

- **Border Width:** The border-width property specifies the width of the four borders.

The width can be set as a specific size (in px, pt, cm, em, etc) or by using one of the three pre-defined values: thin, medium, or thick.

**Border Color:** The border-color property is used to set the color of the four borders.

**Css Margins:** The CSS margin properties are used to create space around elements, outside of any defined borders.

- **Margin-Individual Sides:** The CSS margin properties are used to create space around elements, outside of any defined borders.
- **Margin-Shorthand Property:** To shorten the code, it is possible to specify all the margin properties in one property.

**Margin Collapse:** Top and bottom margins of elements are sometimes collapsed into a single margin that is equal to the largest of the two margins. This does not happen on left and right margins! Only top and bottom margins.

**Css Padding:** The CSS padding properties are used to generate space around an element's content, inside of any defined borders.

- **Padding-Individual sides:** CSS has properties for specifying the padding for each side of an elements are padding-top, padding-bottom, padding-left, padding-right.

- **Padding - Shorthand Property:** To shorten the code, it is possible to specify all the padding properties in one property.

### CSS Height and Width:

- **Setting max-width:** The max-width property is used to set the maximum width of an element. The max-width can be specified in length values, like px, cm, etc., or in percent (%) of the containing block, or set to none (this is default. Means that there is no maximum width).

**CSS Box Model:** All HTML elements can be considered as boxes. In CSS, the term "box model" is used when talking about design and layout.

The CSS box model is essentially a box that wraps around every HTML element. It consists of margins, borders, padding, and the actual content.

- **Letter Spacing:** The letter-spacing property is used to specify the space between the characters in a text.
- **Line Height:** The line-height property is used to specify the space between lines.
- **Text Direction:** The direction property is used to change the text direction of an element.
- **Word Spacing:** The word-spacing property is used to specify the space between the words in a text.

### CSS Font Families:

In CSS, there are two types of font family names:

- **generic family** - a group of font families with a similar look (like "Serif" or "Monospace")

- **font family** - a specific font family (like "Times New Roman" or "Arial"). The font family of a text is set with the font-family property. The font-family property should hold several font names as a "fallback" system. If the browser does not support the first font, it tries the next font, and so on.
- **Font Style:** The font-style property is mostly used to specify italic text. This property has three values:
  - **normal** - The text is shown normally.
  - **italic** - The text is shown in italics.
  - **oblique** - The text is "leaning" (oblique is very similar to italic, but less supported)
- **Font Size:** The font-size property sets the size of the text. Being able to manage the text size is important in web design. However, you should not use font size adjustments to make paragraphs look like headings, or headings look like paragraphs.

**Css Links:** Links can be styled differently depending on what **state** they are in.

The four links states are:

- a:link - a normal, unvisited link
- a:visited - a link the user has visited
- a:hover - a link when the user mouses over it
- a:active - a link the moment it is clicked

**Css Colors:** Colors are specified using predefined color names, or RGB, HEX, HSL, RGBA, HSLA values.

- **Css Names:** In HTML, a color can be specified by using a color name.
- **Background Color:** You can set the background color for HTML elements.
- **Text Color:** You can set the color of text.

- **Border Color:** You can set the color of borders.
- **Color Values:** In HTML, colors can also be specified using RGB values, HEX values, HSL values, RGBA values, and HSLA values.
- **Css Backgrounds:** The CSS background properties are used to define the background effects for elements.
- **Background color:** The background-color property specifies the background color of an element.
- **Background Image:** The background-image property specifies an image to use as the background of an element. By default, the image is repeated so it covers the entire element.
- **Background Repeat:** The background-repeat property specifies a background image will be repeated.
- **Background Attachment:** The background-attachment property specifies a background image is fixed or scrolls with the rest of the page.
- **Background Position:** The background-position property specifies a starting position of a background image.

### Css Borders:

- **Css Border Properties:** The CSS border properties allow you to specify the style, width, and color of an element's border.
- **Border Style:** The border-style property specifies what kind of border to display.
- **Content** - The content of the box, where text and images appear
- **Padding** - Clears an area around the content. The padding is transparent
- **Border** - A border that goes around the padding and content

- **Margin** - Clears an area outside the border. The margin is transparent .The box model allows us to add a border around elements, and to define space between elements.
- **CSS Outline:** An outline is a line that is drawn around elements, OUTSIDE the borders, to make the element "stand out".
- **Outline Style:** The outline-style property specifies the style of the outline, and can have one of the following values like dotted, dashed, solid, double etc.
- **Outline Color:** The outline-color property is used to set the color of the outline.
- **Outline Width:** The outline-width property specifies the width of the outline, and can have one of the following values:
  - thin (typically 1px)
  - medium (typically 3px)
  - thick (typically 5px)
  - A specific size (in px, pt, cm, em, etc)

**Ø Outline Offset:** The outline-offset property adds space between an outline and the edge/border of an element. The space between an element and its outline is transparent.

### **CSS Text:**

- **Text Color:** The color property is used to set the color of the text.
- **Text Alignment:** The text-align property is used to set the horizontal alignment of a text. A text can be left or right aligned, centered, or justified.
- **Text Decoration:** The text-decoration property is used to set or remove decorations from text. The value text-decoration: none; is often used to remove underlines from link.

- **Text Transformation:** The text-transform property is used to specify uppercase and lowercase letters in a text. It can be used to turn everything into uppercase or lowercase letters, or capitalise the first letter of each word.
- **Text Indentation:** The text-indent property is used to specify the indentation of the first line of a text.

## JAVASCRIPT

JavaScript is a cross-platform, object-oriented scripting language. JavaScript is a small, lightweight language, it is not useful as a standalone language, but is designed for easy embedding in other products and applications, such as web browsers. Inside a host environment, java script can be connected to the objects of its environment to provide programmatic control over them.Core database contains a core set of objects, such as Arrays, Date and Month, and a core set of language elements such as operators, control structures and statements. Core JavaScript can be extended for a variety of purpose by supplementing it with additional objects.Client-side JavaScript extends the core language by supplying objects to control a browser Navigation or another web browser and it's Document Object Model (DOM). For example, client-side extension allow an application to place elements on an HTML form and responds to user events such as mouse clicks, form input, and page navigation.Server-side JavaScript extends the core language by supplying objects relevant to running JavaScript on a server. For example, server-side extensions allow an application to communicate with a relational database, provide community and perform file manipulations on a server.Through JavaScript's live connect functionality, let java and JavaScript code communicates with each other. Since then, many of sites have it to be easy to use but something quite limiting. These limiting factors being addressed but the World Wide Web Consortium at MIT. But HTML had to start somewhere, and its success argues that it did not start out too badly.

- JavaScript is a lightweight, interpreted programming language.
- Designed for creating network-centric applications.
- Complementary to and integrated with Java.
- Complementary to and integrated with HTML.
- Open and cross-platform.

**JavaScript Can Change HTML Content:** One of many JavaScript HTML methods is getElementById().

**<script> Tag:** In HTML, JavaScript code must be inserted between <script> and </script> tags.

**JavaScript in <head> or <body>:** You can place any number of scripts in an HTML document. Scripts can be placed in the <body>, or in the <head> section of an HTML page, or in both.

### **External JavaScript:**

External scripts are practical when the same code is used in many different web pages.

JavaScript files have the file extension .js.

To use an external script, put the name of the script file in the src (source) attribute of a <script> tag:

### **JavaScript Display Possibilities:**

JavaScript can "display" data in different ways:

- Writing into an HTML element, using innerHTML.
- Writing into the HTML output using document.write().
- Writing into an alert box, using window.alert().
- Writing into the browser console, using console.log().

### **Using inner HTML:**

To access an HTML element, JavaScript can use the document.getElementById(id) method.

The id attribute defines the HTML element. The innerHTML property defines the HTML content.

**Using window.alert():** You can use an alert box to display data.

**Using console.log():** For debugging purposes, you can use the console.log() method to display data.

**JavaScript White Space:** JavaScript ignores multiple spaces. You can add white space to your script to make it more readable.

**JavaScript Code Blocks:** JavaScript statements can be grouped together in code blocks, inside curly brackets {...}. The purpose of code blocks is to define statements to be executed together.

**JavaScript Values:** The JavaScript syntax defines two types of values: Fixed values and variable values. Fixed values are called **literals**. Variable values are called **variables**.

**JavaScript Variables:** In a programming language, **variables** are used to **store** data values. JavaScript uses the var keyword to **declare** variables. An **equal sign** is used to **assign values** to variables.

### **JavaScript Identifiers:**

All JavaScript **variables** must be **identified** with **unique names**.

These unique names are called **identifiers**.

Identifiers can be short names (like x and y) or more descriptive names (age, sum, totalVolume).

The general rules for constructing names for variables (unique identifiers) are:

- Names can contain letters, digits, underscores, and dollar signs.
- Names must begin with a letter
- Names can also begin with \$ and \_
- Names are case sensitive (y and Y are different variables)
- Reserved words (like JavaScript keywords) cannot be used as names

## **JavaScript Functions:**

A JavaScript function is a block of code designed to perform a particular task.

A JavaScript function is executed when "something" invokes it.

## **Function Invocation:**

The code inside the function will execute when "something" **invokes** (calls) the function:

- Ø When an event occurs (when a user clicks a button)
- Ø When it is invoked (called) from JavaScript code
- Ø Automatically (self invoked)

## **Function Return:**

When JavaScript reaches a return statement, the function will stop executing.

If the function was invoked from a statement, JavaScript will "return" to execute the code after the invoking statement.

## **JavaScript Events:**

HTML events are "**things**" that happen to HTML elements.

When JavaScript is used in HTML pages, JavaScript can "**react**" on these events.

## **HTML Events:**

An HTML event can be something the browser does, or something a user does.

Here are some examples of HTML events:

- An HTML web page has finished loading
- An HTML input field was changed
- An HTML button was clicked

Often, when events happen, you may want to do something. JavaScript lets you execute code when events are detected.

HTML allows event handler attributes, **with JavaScript code**, to be added to HTML elements.

- **Onchange-** An HTML element has been changed.
- **Onclick-** The user clicks an HTML element.
- **Onmouseover-** The user moves the mouse over an HTML element.
- **Onmouseout-** The user moves the mouse away from an HTML element.

## What is Bootstrap ?

- Bootstrap is the most popular HTML, CSS and JavaScript framework for developing a responsive and mobile friendly website.
- It is absolutely free to download and use.
- It is a front-end framework used for easier and faster web development.
- It includes HTML and CSS based design templates for typography, forms, buttons, tables, navigation, modals, image carousels and many others.
- It can also use JavaScript plug-ins.
- It facilitates you to create responsive designs.

## **SYSTEM CONSTRAINTS**

### **USER INTERFACE CONSTRAINTS:-**

Using this portal is fairly simple and intuitive. A user familiar with basic browser navigation skills should be able to understand all functionality provided by the portal.

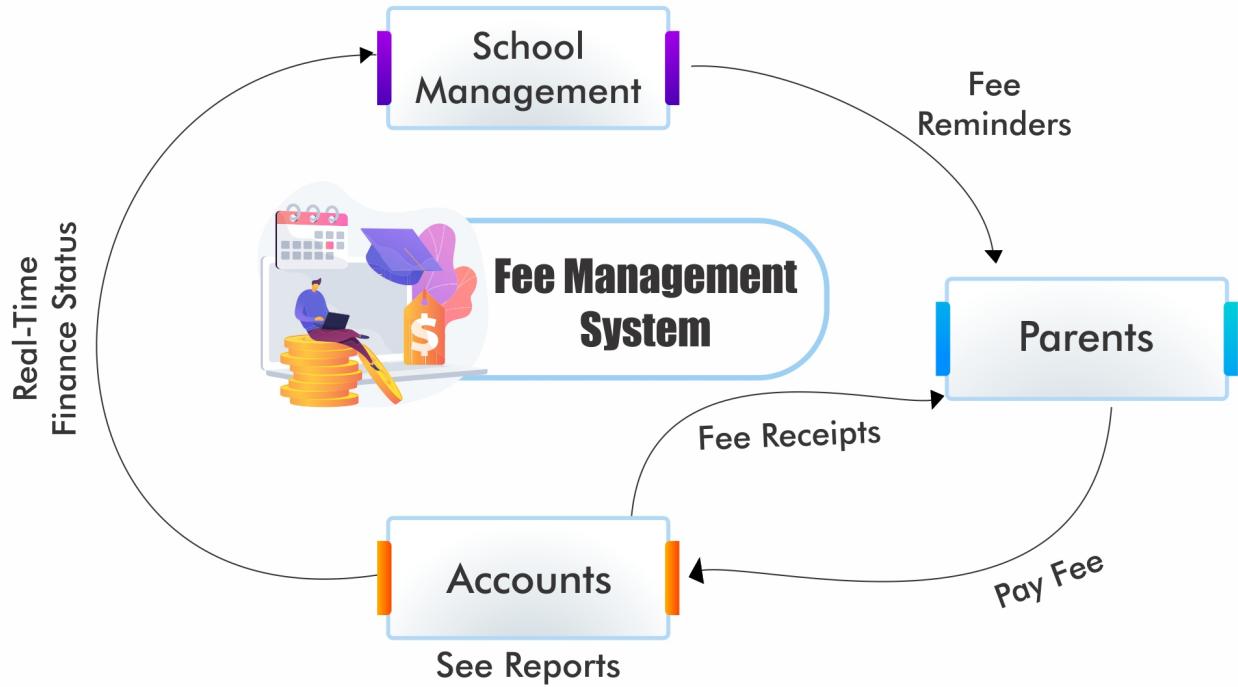
### **HARDWARE CONSTRAINTS:-**

The portal should work on most home desktop and laptop computers.

### **SOFTWARE CONSTRAINTS:-**

The portal is designed to run on google Chrome, Mozilla Firefox and Internet Explorer 10

## SYSTEM ARCHITECTURE



## Screens



The image shows a student details form titled "STUDENT DETAILS". At the top, there are navigation icons for menu, home, session (2021/2022), term (1st Term), user profile, and grid view. Below the title, it says "Add new student". The form consists of several input fields:

|                      |                             |
|----------------------|-----------------------------|
| Current status       | Active                      |
| Registration number  | 319129510028                |
| Surname              | MANTHA                      |
| Firstname            | Yashwanth                   |
| Other name           | bharadwaj                   |
| Gender               | Male                        |
| Date of birth        | 26/03/2022                  |
| Current class        | CSE III/IV                  |
| Date of admission    | 2022-03-26                  |
| Parent mobile number | 08686424320                 |
| Address              | visakhapatnam ,madhilapalem |

## PAYMENT DETAILS AND HISTORY

WISTM FEE MANAGEMENT

Session: 2021/2022 Term: 1st Term

**MANTHA YASHWANTH**  
**BHARADWAJ (319129510028)**

2021/2022  
1st Term  
CSE III/IV  
Status: Active

Expected Balance: 238,005

Invoice Breakdown

| S/N | Description                | Amount  |
|-----|----------------------------|---------|
| 1   | TUITION FEE                | 300000  |
|     | Total Amount this term     | 300,000 |
|     | Balance from previous term | 5       |
|     | Total Amount Payable       | 300,005 |
|     | Total Amount Paid          | 62,000  |

Payment History

| S/N | Amount Paid | Date Paid      | Comment Paid     |
|-----|-------------|----------------|------------------|
| 1   | 60000       | March 25, 2022 | paid sucessfully |
| 2   | 2000        | March 26, 2022 | paid sucessfully |

## PAYMENT UPDATION INTERFACE

WISTM FEE MANAGEMENT

Session: 2021/2022 Term: 1st Term

**Amount paid**

**Date paid**

**Comment**

**Add payment**

**INVOICE**

MANTHA YASHWANTH BHARADWAJ  
(319129510028)

- Session: 2021/2022
- Term: 1st Term
- Class: CSE III/IV
- Total Amount Payable: 300,005
- Total Amount Paid: 60,000
- Balance: 240,005

# STUDENT PAYMENT HISTORY

Session: 2021/2022 Term: 1st Term

 MANTHA  
Yashwanth  
bharadwaj  
319129510030

CSE III/IV  
Male | Active

**Invoice/Payment History**  
There are no payment history for MANTHA Yashwanth bharadwaj (319129510030)

Print   

# BULK REPORTS

Session: 2021/2022 Term: 1st Term

**Select Student(s) and then click on proceed**

**Proceed >** You can also search by name/class by typing in the search box

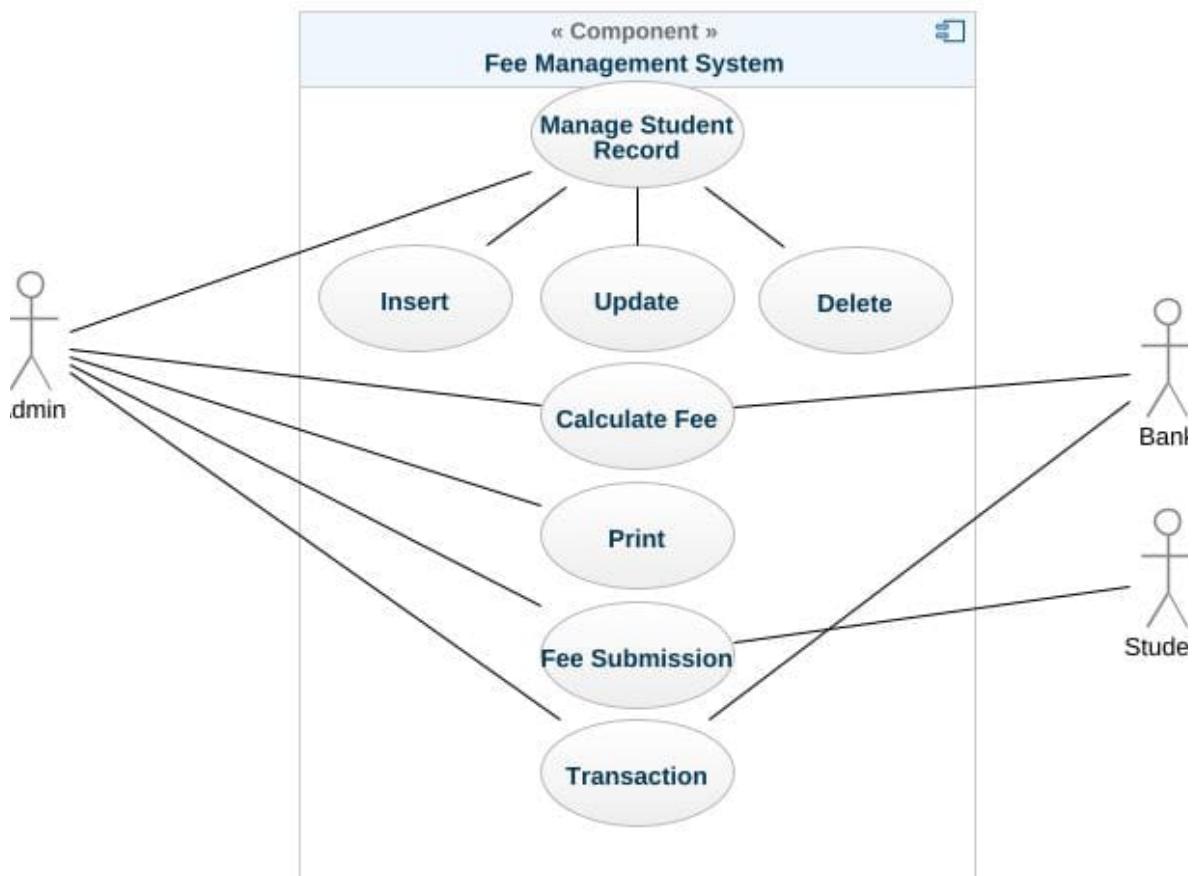
Show 100 entries Search:

| <input type="checkbox"/> | Name                                      | Current Class |
|--------------------------|---|---------------|
| <input type="checkbox"/> | MANTHA YASHWANTH BHARADWAJ (319129510028) | CSE III/IV    |
| <input type="checkbox"/> | MANTHA Yashwanth bharadwaj (319129510030) | CSE III/IV    |
| <input type="checkbox"/> | VERMA3 YUGESH3 (3)                        | CSE II/IV     |
| <input type="checkbox"/> | pradhan gajen (2)                         | CSE 1/IV      |
| <input type="checkbox"/> | sabbavarapu swapna s (3po2r)              | CSE II/IV     |
| <input type="checkbox"/> | verma yugesh (1)                          | CSE II/IV     |

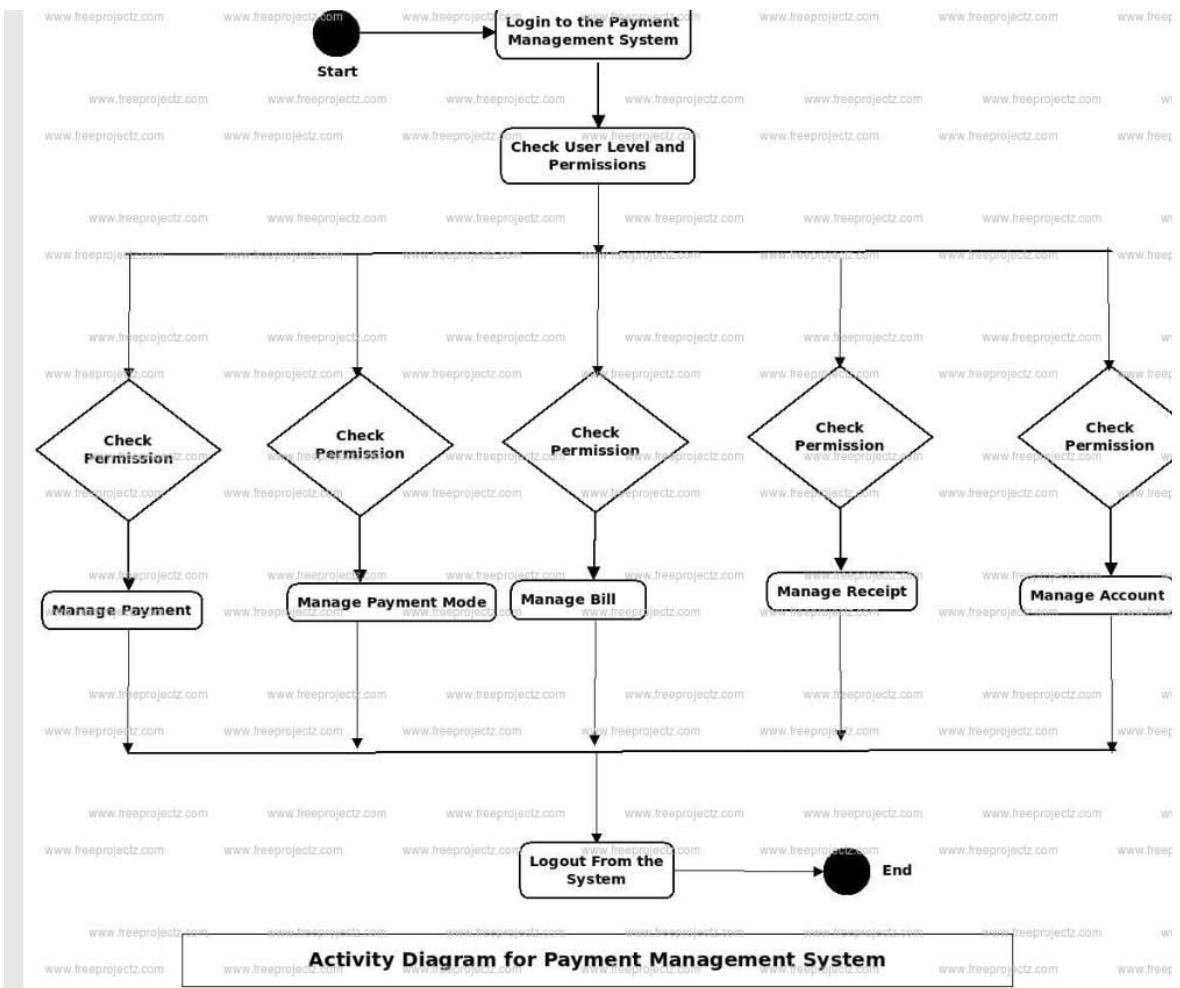
Showing 1 to 6 of 6 entries **Previous** **1** **Next**

**Proceed >**

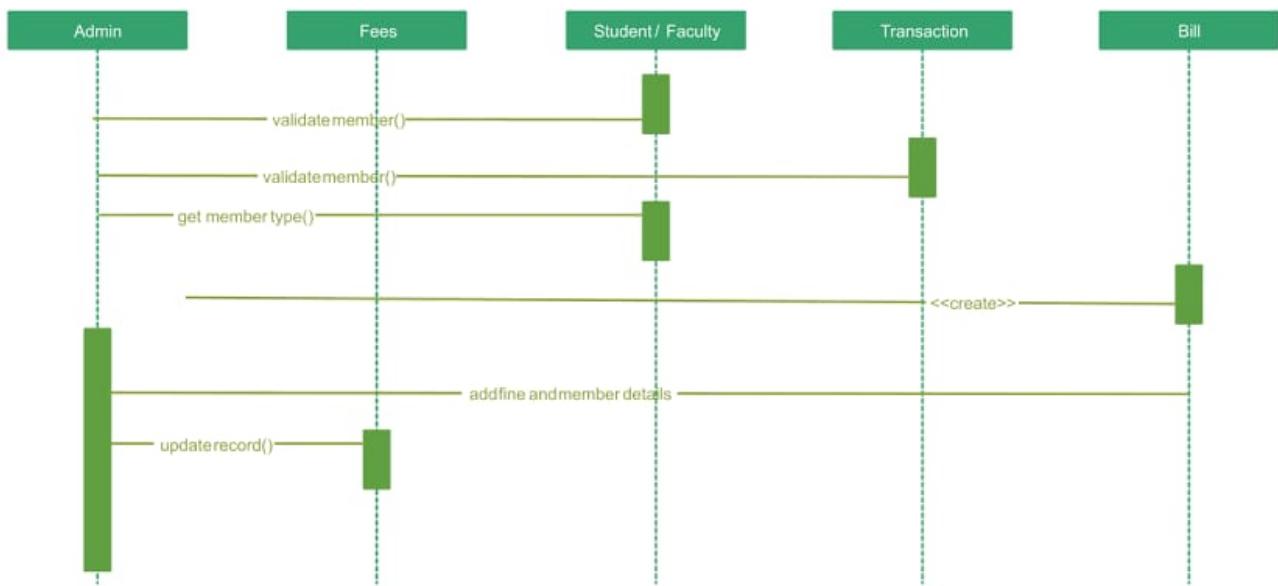
## UML DIAGRAMS



COMPONENT DIAGRAM



## ACTIVITY DIAGRAM



## SEQUENCE DIAGRAM

## CODING

### TEMPLATES

```
{% extends 'base.html' %}
```

```
{% load widget_tweaks %}
```

```
{% block title %}
```

Classes

```
{% endblock title %}
```

```
{% block breadcrumb %}
```

```
<a class="btn btn-primary" href="#" data-toggle="modal" data-target="#modal1">
```

```
 <i class="fa fa-plus"></i>
```

Add new class

```
</a>
```

```
{% endblock breadcrumb %}
```

```
{% block content %}
```

```
<div class="row">
```

```
 <div class="col-sm-12">
```

```
 <table class="table table-bordered">
```

```
   <tbody>
```

```
     {% for object in object_list %}
```

```

<tr>
    <td>{{ object }}</td>
    <td><a href="{% url 'class-update' object.id %}"><i
        class="fa fa-edit"></i></a> <a
        href="{% url 'class-delete' object.id %}"><i
        class="fa fa-trash-alt"></i></a></td>
</tr>
{% endfor %}
</tbody>
</table>

</div>
</div>

<div class="modal fade" id="modal1" tabindex="-1" role="dialog" aria-
labelledby="exampleModalLabel" aria-hidden="true">
    <div class="modal-dialog">
        <div class="modal-content">
            <form action="{% url 'class-create' %}" method="POST">
                <div class="modal-header">
                    <h5 class="modal-title" id="exampleModalLabel">Add new class</h5>
                    <button type="button" class="close" data-dismiss="modal" aria-
label="Close">
                        <span aria-hidden="true">&times;</span>
                    </button>
                </div>
                <div class="modal-body">
                    {% csrf_token %}

```

```
{% for field in form %}  
    <div class="form-group">  
        {{ field.label_tag }}  
        {{ field|add_class:"form-control" }}  
        {{ field.help_text }}  
        {{ field.errors }}  
    </div>  
{% endfor %}  
</div>  
  
<div class="modal-footer">  
    <button type="button" class="btn btn-secondary" data-  
dismiss="modal">Close</button>  
    <button type="submit" class="btn btn-primary">Save</button>  
    </div>  
</form>  
</div>  
</div>  
</div>  
  
{% endblock content %}
```

```
{% extends 'base.html' %}  
  
{% block title %}  
<span class="card-title"><strong>{{ object }}</strong> about to be  
deleted!</span>  
{% endblock title %}
```

```
{% block content %}

<div class="row">
    <div class="col-sm-12">
        <div class="card">
            <div class="card-body">
                <p>Are you sure you want to delete object <strong>{{ object }}</strong>
                    alongside all attached contents?</p>
                <div class="mt-2 red-text">This action cannot be undone</div>

                <form method="POST" class="mt-3">
                    {% csrf_token %}

                    <button type="submit" class="btn btn-primary">
                        Confirm Delete
                        <i class="fa fa-trash-alt"></i>
                    </button>
                </form>
            </div>
        </div>
    </div>
</div>

{% endblock content %}
```

```
{% extends 'base.html' %}
```

```
{% load widget_tweaks %}
```

```
{% block title %}
```

Set Current Session and Term

```
{% endblock title %}
```

```
{% block title-side %}{% endblock title-side %}
```

```
{% block content %}
```

```
<form method="POST">
```

```
{% csrf_token %}
```

```
<div class="mb-3">
```

```
{% for field in form %}
```

```
<div class="form-group row">
```

```
 <div class="col-2">{{ field.label_tag }}</div>
```

```
 <div class="col-6">
```

```
 {{ field | add_class:"form-control" }}
```

```
 <span class="small">{{ field.help_text |safe }}</span>
```

```
 </div>
```

```
</div>
```

```
{% endfor %}
```

```
</div>
```

```
<input type="submit" value="Save" class="w-25 btn btn-primary">

</form>

{% endblock content %}

{% load widget_tweaks %}

{% for field in form %}

<div class="form-group row">

<label class="col-sm-3 col-form-label" for="{{ field.auto_id }}>

{{ field.label }}

{% for error in field.errors %}

<small id="{{ field.auto_id }}" class="text-danger">{{ error }}</small>

{% endfor %}

</label>

<div class="col-sm-9">

{{ field | add_class:"form-control" }}

{% if field.help_text %}

<small>{{ field.help_text|safe }}</small>

{% endif %}

</div>

</div>

{% endfor %}
```

```
{% extends 'base.html' %}  
{% load widget_tweaks %}
```

{% block title %}

## Academic Sessions

{% endblock title %}

{% block breadcrumb %}

```
<a class="btn btn-primary" href="#" data-toggle="modal" data-target="#modal1">
  <i class="fa fa-plus"></i>
  Add new session
</a>
```

{% endblock breadcrumb %}

{% block content %}

```
<div class="row">  
    <div class="col-sm-12">
```

```
<table class="table table-bordered">
```

<tbody>

```
{% for object in object_list %}
```

<tr>

<td>{{ object }}</td>

<td>

{% if object.current == True %}

Current

{% endif %}

```

</td>
<td><a href="{% url 'session-update' object.id %}"><i
    class="fa fa-edit"></i></a> <a
    href="{% url 'session-delete' object.id %}"><i
    class="fa fa-trash-alt"></i></a></td>
</tr>
{% endfor %}
</tbody>
</table>

</div>
</div>

<div class="modal fade" id="modal1" tabindex="-1" role="dialog" aria-
labelledby="exampleModalLabel" aria-hidden="true">
<div class="modal-dialog">
<div class="modal-content">
<form action="{% url 'session-create' %}" method="POST">
<div class="modal-header">
    <h5 class="modal-title" id="exampleModalLabel">Add new session</h5>
    <button type="button" class="close" data-dismiss="modal" aria-
label="Close">
        <span aria-hidden="true">&times;</span>
    </button>
</div>
<div class="modal-body">
    {% csrf_token %}

```

```
{% for field in form %}

<div class="form-group">
    {{ field.label_tag }}
    {{ field|add_class:"form-control" }}
</div>

{% endfor %}

</div>

<div class="modal-footer">
    <button type="button" class="btn btn-secondary" data-
dismiss="modal">Close</button>
    <button type="submit" class="btn btn-primary">Save</button>
</div>
</form>
</div>
</div>

{% endblock content %}

{% extends 'base.html' %}

{% load widget_tweaks %}

{% block title %}
    {{ title }}

```

```
{% endblock title %}

{% block title-side %} {% endblock title-side %}

{% block content %}
<form method="POST">
    {% csrf_token %}
    {{ formset.management_form}}

<div class="mb-3">
    {% for form in formset %}
        <div class="form-group row">
            {{ form.id }}
            <div class="col-2">{{ form.key.value }}</div>
            <div class="col-6">{{ form.value | add_class:"form-control" | attr:"required" }}</div>
        </div>
        {% if form.errors %}
            {% endif %}
        </div>
        {% endfor %}
</div>

<input type="submit" value="Save" class="w-25 btn btn-primary">
```

```
</form>
```

```
{% endblock content %}
```

```
{% extends 'base.html' %}
```

```
{% block title %}
```

```
{% endblock title %}
```

```
{% block content %}
```

```
<form action="" method="POST">
```

```
{% csrf_token %}
```

```
<div class="alert">
```

```
  <p>Are you sure you want to delete "{{ object }}"?</p>
```

```
  <p>All contents attached to this student will be deleted</p>
```

```
</div>
```

```
<input type="submit" class="btn btn-primary" value="Confirm">
```

```
</form>
```

```
{% endblock content %}
```

```

{%- extends 'base.html' %}

{%- load static %}

{%- block content-header %}

<div class="card-header">

<div class="card-tools">

<a href="#" class="btn btn-tool">Print</a>

<a href="{% url 'student-update' object.id %}" class="btn btn-tool">

<i class="fas fa-edit"></i>

</a>

<a href="{% url 'student-delete' object.id %}" class="btn btn-tool">

<i class="fas fa-times"></i>

</a>

</div>

</div>

{%- endblock content-header %}

{%- block content %}

<div class="row mb-4">

<div class="col-sm-4">

<div class="row mb-2">

{%- if object.passport %}



{%- else %}

"  
{%- endif %}  
  
<div class="float-right">  
    <h4>{{ object.surname }}</h4>  
    <h4>{{ object.firstname }}</h4>  
    <h4>{{ object.other_name }}</h4>  
    <h4>{{ object.registration_number }}</h4>  
</div>  
</div>  
<h5>{{ object.current_class }}</h5>  
<h5>{{ object.get_gender_display }} | {{ object.get_current_status_display }}</h5>  
  
</div>  
<div class="col-sm-8">  
    <div>  
        <div class="form-group border-bottom row">  
            <label class="col">  
                Date of Birth  
            </label>  
            <div class="col">  
                {{ object.date_of_birth }}  
            </div>  
        </div>  
    </div>  
  
<div class="form-group border-bottom row">
```

Parent Mobile Number

</label>

<div class="col">

{ { object.parent\_mobile\_number } }

</div>

</div>

<div class="form-group border-bottom row">

<label class="col">

Address

</label>

<div class="col">

{ { object.address } }

</div>

</div>

<div class="form-group border-bottom row">

<label class="col">

Comments

</label>

<div class="col">

{ { object.others } }

</div>

</div>

</div>

</div>

</div>

<h4>Invoice/Payment History</h4>

```

{%- if payments %}

<table class="table table-sm table-bordered table-hover">

  <thead class="thead-light">
    <tr>
      <th>For</th>
      <th>Amount Payable</th>
      <th>Amount Paid</th>
      <th>balance</th>
    </tr>
  </thead>
  <tbody>

    {%- for payment in payments %}

      <tr class='clickable-row' data-href="{% url 'invoice-detail' payment.id %}">
        <td>{{payment.session}} - {{payment.term}} - {{payment.class_for}}</td>
        <td>{{payment.total_amount_payable}}</td>
        <td>{{payment.total_amount_paid}}</td>
        <td>{{payment.balance}}</td>
      </tr>
    {%- endfor %}

  </tbody>
</table>

{%- else %}

  <p>There are no payment history for {{object}}</p>
{%- endif %}

{%- endblock content %}

```

```
{% extends 'base.html' %}
```

```
{% block title %}Students{% endblock title %}
```

```
{% block breadcrumb %}
```

```
<a class="btn btn-primary" href="{% url 'student-create' %}"><i  
    class="fas fa-plus"></i> New student</a>  
<a class="btn btn-primary" href="{% url 'student-upload' %}"><i  
    class="fas fa-upload"></i> Upload students</a>  
{% endblock breadcrumb %}
```

```
{% block content %}
```

```
<div class="table-responsive">  
    <table id="studenttable" class="table table-bordered table-hover" data-page-  
length='100'>  
        <thead class="thead-light">  
            <tr>  
                <th>S/N</th>  
                <th>Fullname</th>  
                <th>Registration Number</th>  
                <th>Current Class</th>  
                <th>Gender</th>  
                <th>Mobile Phone Number</th>  
                <th>Status</th>  
            </tr>
```

```

</thead>
<tbody>

{%- for student in students %}

<tr class='clickable-row' data-href="{% url 'student-detail' student.id %}">
<td>{{ forloop.counter }}</td>
<td>{{ student.surname }} {{ student.firstname }} {{ student.other_name }}</td>
<td>{{ student.registration_number }}</td>
<td>{{ student.current_class }}</td>
<td>{{ student.get_gender_display }}</td>
<td>{{ student.parent_mobile_number }}</td>
<td>{{ student.get_current_status_display }}</td>
</tr>

{%- endfor %}

</tbody>
</table>
</div>

{%- endblock content %}

{%- block morejs %}

<script>
$('#studenttable').DataTable({
});
</script>

{%- endblock morejs %}

```

```
{% extends 'base.html' %}
```

```
{% block title %}
```

Upload Bulk Students from file

```
{% endblock title %}
```

```
{% block content %}
```

```
<div class="col alert alert-info">
```

```
    <h5>Instruction</h5>
```

```
    <p>Please <a href="{% url 'download-csv' %}">download the CSV file attached here</a>. Fill in the sheet and upload the same file back to this page. DO NOT EDIT THE FIRST ROW</p>
```

```
  </div>
```

```
<form method="POST" enctype="multipart/form-data">
```

```
    {% csrf_token %}
```

```
    {% include 'corecode/form_snippet.html' %}
```

```
    <input type="submit" class="btn btn-success" value="Submit">
  </form>
```

```
{% endblock content %}
```

```

{%- load static %}

{%- load widget_tweaks %}

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta http-equiv="x-ua-compatible" content="ie=edge">

    <title>{{ school_name | default:"MySchool" }}</title>

    <!-- Font Awesome Icons -->
    <link rel="stylesheet"
        href="{% static 'plugins/fontawesome-free/css/all.min.css' %}">

    <link rel="stylesheet" href="{% static 'plugins/toastr/toastr.min.css' %}">
    <!-- overlayScrollbars -->
    <link rel="stylesheet"
        href="{% static 'plugins/overlayScrollbars/css/OverlayScrollbars.min.css' %}">
    <!-- Theme style -->
    <link rel="stylesheet" href="{% static 'dist/css/adminlte.min.css' %}">

    <!--Login Page Csss-->
    <link rel="stylesheet" href="{% static 'dist/css/login-page.css' %}">

    <!-- Google Font: Source Sans Pro -->
    <link

```

```
    href="https://fonts.googleapis.com/css?
family=Source+Sans+Pro:300,400,400i,700"
    rel="stylesheet">
</head>
<body>

<div class="limiter">
    <div class="container-login100">
        <div class="wrap-login100">
            <div class="login100-pic js-tilt" style="width: 400px;">
                
            </div>

            <form class="login100-form validate-form" method="post" action="{% url
'login' %}">
                {% csrf_token %}
                <span class="login100-form-title">
                    WISTM FEE MANAGEMENT
                </span>
                <span class="error-msg">
                    {% if form.errors %}
                        <p class="error-msg">Your username and password didn't match. Please
try again.</p>
                    {% endif %}
                <% if next %>
                <% if user.is_authenticated %>
                    <p class="error-msg"> Your account doesn't have access to this page. To
proceed,
                    please login with an account that has access.</p>
                
```

```
{% else %}  
    </span>  
    <p>Please Login to see this page.</p>  
    {% endif %}  
    {% endif %}  
    </span>  
    <div class="wrap-input100 validate-input" data-validate="Valid email is  
required: ex@abc.xyz">  
        <input class="input100" id="id_username" name="username"  
            type="text" placeholder="Username">  
        <span class="focus-input100"></span>  
        <span class="symbol-input100">  
            <i class="fa fa-envelope" aria-hidden="true"></i>  
        </span>  
    </div>  
  
<div class="wrap-input100 validate-input" data-validate="Password is  
required">  
    <input class="input100" type="password" id="id_password"  
        name="password" placeholder="Password">  
    <span class="focus-input100"></span>  
    <span class="symbol-input100">  
        <i class="fa fa-lock" aria-hidden="true"></i>  
    </span>  
    </div>  
  
<div class="container-login100-form-btn">  
    <button class="login100-form-btn" type="submit" value="Login">  
        Login  
    </button>
```

```

<input type="hidden" name="next" value="{{ next }}">

</div>

<div class="text-center p-t-12">
    <span class="txt1">
        Forgot
    </span>
    <a class="txt2" href="#">
        Username / Password?
    </a>
</div>

</form>
</div>
</div>
</div>

<!-- REQUIRED SCRIPTS -->
<!-- jQuery -->
<script src="{% static 'plugins/jquery/jquery.min.js' %}"></script>
<script src="{% static 'dist/js/jquery.formset.js' %}"> </script>
<!-- Bootstrap -->
<script
    src="{% static 'plugins/bootstrap/js/bootstrap.bundle.min.js' %}"></script>
<script src="{% static 'plugins/toastr/toastr.min.js' %}"></script>
<!-- overlayScrollbars -->
<script

```

```

src="<% static 'plugins/overlayScrollbars/js/jquery.overlayScrollbars.min.js'
%>"></script>

<!-- AdminLTE App -->
<script src="<% static 'dist/js/adminlte.js' %>"></script>

<!-- OPTIONAL SCRIPTS -->
<script src="<% static 'dist/js/demo.js' %>"></script>

{%
    if messages %}

    {% for message in messages %}

        <script type="text/javascript">
            toastr.options.progressBar = true;
            toastr.{{ message.tags }}('{{ message }}')
        </script>

    {% endfor %}

    {% endif %}

</body>

```

## 11.BASE .HTML

```

{%
    load static %}

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">

```

```

<meta name="viewport" content="width=device-width, initial-scale=1">
<meta http-equiv="x-ua-compatible" content="ie=edge">

<title>{{ school_name | default:"wellfarecollege" }}</title>

<!-- Font Awesome Icons -->
<link rel="stylesheet" href="{{ static 'plugins/fontawesome-free/css/all.min.css' }}">
<link rel="stylesheet" href="{{ static 'plugins/toastr/toastr.min.css' }}">
<!-- overlayScrollbars -->
<link rel="stylesheet"
      href="{{ static 'plugins/overlayScrollbars/css/OverlayScrollbars.min.css' }}">
<link rel="stylesheet" href="{{ static 'plugins/Datatables/datatables.min.css' }} />
>

<!-- Theme style -->
<link rel="stylesheet" href="{{ static 'dist/css/adminlte.min.css' }}">
<!-- Google Font: Source Sans Pro -->
<link
      href="https://fonts.googleapis.com/css?
family=Source+Sans+Pro:300,400,400i,700"
      rel="stylesheet">
</head>

<body
      class="hold-transition sidebar-mini layout-fixed layout-navbar-fixed layout-footer-fixed">
<div class="wrapper">
    <!-- Navbar -->
    <nav class="main-header navbar navbar-expand navbar-white navbar-light">
        <!-- Left navbar links -->

```

```

<ul class="navbar-nav">
  <li class="nav-item">
    <a class="nav-link" data-widget="pushmenu" href="#" role="button"><i
      class="fas fa-bars"></i></a>
  </li>
  <li class="nav-item d-none d-sm-inline-block">
    <a href="{% url 'home' %}" class="nav-link"><i class="fa fa-home"></i></a>
  </li>
  <li class="nav-item d-none d-sm-inline-block">
    <a href="{% url 'current-session' %}" class="nav-link">Session:
    {{ current_session }}</a>
  </li>
  <li class="nav-item d-none d-sm-inline-block">
    <a href="{% url 'current-session' %}" class="nav-link">Term:
    {{ current_term }}</a>
  </li>
</ul>

```

```

<!-- Right navbar links -->
<ul class="navbar-nav ml-auto">
  <!-- Notifications Dropdown Menu -->
  <li class="nav-item dropdown">
    <a class="nav-link" data-toggle="dropdown" href="#">
      <i class="far fa-user"></i>
    </a>
    <div class="dropdown-menu dropdown-menu-lg dropdown-menu-right">
      <a href="#" class="dropdown-item">
        <i class="fas fa-user mr-2"></i> Profile

```

```

</a>
<div class="dropdown-divider"></div>
<a href="{% url 'logout' %}" class="dropdown-item">
    <i class="fas fa-sign-out"></i> Sign out
</a>
</div>
</li>
<li class="nav-item">
    <a class="nav-link" data-widget="control-sidebar" data-slide="true"
        href="#" role="button"><i class="fas fa-th-large"></i></a>
</li>
</ul>
</nav>
<!-- /.navbar -->

<!-- Main Sidebar Container -->
<aside class="main-sidebar sidebar-dark-primary elevation-4">
    <!-- Brand Logo -->
    <a href="{% url 'home' %}" class="brand-link">WISTM FEE
MANAGEMENT</a>

<!-- Sidebar -->
<div class="sidebar">

    <!-- Sidebar Menu -->
    <nav class="mt-2">
        <ul class="nav nav-pills nav-sidebar flex-column"
            data-widget="treeview" role="menu" data-accordion="false">
            <li class="nav-item">
                <a href="{% url 'student-list' %}" class="nav-link">

```

```
<i class="nav-icon fas fa-graduation-cap"></i>
<p>
    Students
</p>
</a>
</li>

<li class="nav-item">
    <a href="{% url 'staff-list' %}" class="nav-link">
        <i class="nav-icon fas fa-users"></i>
        <p>
            Member of Staff
        </p>
    </a>
</li>

<li class="nav-header">Finance</li>
<li class="nav-item">
    <a href="{% url 'invoice-list' %}" class="nav-link">
        <i class="nav-icon fas fa-money-bill-alt"></i>
        <p>
            Invoices
        </p>
    </a>
</li>

<li class="nav-header">Results</li>
<li class="nav-item">
    <a href="{% url 'create-result' %}" class="nav-link">
        <i class="nav-icon fas fa-money-bill-alt"></i>
        <p>
            Create Results
        </p>
    </a>
</li>
```

```
</p>
</a>
</li>
<li class="nav-item">
  <a href="{% url 'edit-results' %}" class="nav-link">
    <i class="nav-icon fas fa-money-bill-alt"></i>
    <p>
      Update Results
    </p>
  </a>
</li>
<li class="nav-item">
  <a href="{% url 'view-results' %}" class="nav-link">
    <i class="nav-icon fas fa-money-bill-alt"></i>
    <p>
      View Results
    </p>
  </a>
</li>

<li class="nav-header">Others</li>
<li class="nav-item has-treeview">
  <a href="#" class="nav-link">
    <i class="nav-icon fas fa-copy"></i>
    <p>
      Management
      <i class="fas fa-angle-left right"></i>
    </p>
  </a>
  <ul class="nav nav-treeview">
```

```
<li class="nav-item">
  <a href="{% url 'sessions' %}" class="nav-link">
    <i class="nav-icon fas fa-bars"></i>
    <p>
      Sessions
    </p>
  </a>
</li>

<li class="nav-item">
  <a href="{% url 'terms' %}" class="nav-link">
    <i class="nav-icon fas fa-bars"></i>
    <p>
      Terms
      <i class="right"></i>
    </p>
  </a>
</li>

<li class="nav-item">
  <a href="{% url 'subjects' %}" class="nav-link">
    <i class="nav-icon fas fa-bars"></i>
    <p>
      Subjects
    </p>
  </a>
</li>

<li class="nav-item">
  <a href="{% url 'classes' %}" class="nav-link">
    <i class="nav-icon fas fa-bars"></i>
    <p>
      Classes
    </p>
  </a>
</li>
```

```

        </p>
        </a>
    </li>
</ul>
</li>

<li class="nav-item">
    <a href="{% url 'configs' %}" class="nav-link">
        <i class="nav-icon fas fa-cog"></i>
        <p>
            Settings
        </p>
    </a>
</li>

</ul>
</nav>
<!-- ./sidebar-menu -->
</div>
<!-- ./sidebar -->
</aside>

<!-- Content Wrapper. Contains page content -->
<div class="content-wrapper">
    <section class="content-header">
        <div class="container-fluid">
            <div class="row mb-1">
                <div class="col-sm-8">
                    <h1>{% block title %} {% endblock title %}</h1>
                </div>
            </div>
        </div>
    </section>
    <div class="content-body">
        <div class="row">
            <div class="col-12">
                <div class="card">
                    <div class="card-body">
                        <h3>Recent Configs</h3>
                        <table>
                            <thead>
                                <tr>
                                    <th>Config ID</th>
                                    <th>Last Modified</th>
                                    <th>Actions</th>
                                </tr>
                            </thead>
                            <tbody>
                                <tr>
                                    <td>config-1</td>
                                    <td>2023-10-01</td>
                                    <td><a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a></td>
                                </tr>
                                <tr>
                                    <td>config-2</td>
                                    <td>2023-10-02</td>
                                    <td><a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a></td>
                                </tr>
                                <tr>
                                    <td>config-3</td>
                                    <td>2023-10-03</td>
                                    <td><a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a></td>
                                </tr>
                            </tbody>
                        </table>
                    </div>
                </div>
            </div>
        </div>
    </div>
</div>

```

```
<div class="col-sm-4">
  <ol class="breadcrumb float-sm-right">
    <li class="breadcrumb-item">{% block breadcrumb %} {% endblock
      breadcrumb %}</li>
  </ol>
</div>
</div>
</div><!-- /.container-fluid -->
</section>
<!-- Main content -->
<section class="content">
  <div class="container-fluid">
    <div class="row">
      <div class="col-md-12">
        {% block fullcard %}
        <div class="card">
          {% block content-header %} {% endblock content-header %}
          <div class="card-body">
            <div class="row">
              <div class="col">
                {% block content %}
                {% endblock content %}
              </div>
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>
</section>
```

```

    {% endblock fullcard %}

    <!-- /.card -->
</div>
<!-- /.col -->
</div>
<!-- /.row -->
</div>

<!--/. container-fluid -->
</section>
<!-- /.content -->
</div>
<!-- /.content-wrapper -->

<!-- Control Sidebar -->
<aside class="control-sidebar control-sidebar-dark">
    <!-- Control sidebar content goes here -->
</aside>
<!-- /.control-sidebar -->

</div>
<!-- ./wrapper -->

<!-- REQUIRED SCRIPTS -->
<!-- jQuery -->
<script src="{% static 'plugins/jquery/jquery.min.js' %}"></script>
<script src="{% static 'dist/js/jquery.formset.js' %}"></script>
<!-- Bootstrap -->
<script src="{% static 'plugins/bootstrap/js/bootstrap.bundle.min.js' %}"></script>
<script src="{% static 'plugins/toastr/toastr.min.js' %}"></script>

```

```

<!-- overlayScrollbars -->
<script
  src="<% static 'plugins/overlayScrollbars/js/jquery.overlayScrollbars.min.js' %>"></script>

<!-- AdminLTE App -->
<script src="<% static 'dist/js/adminlte.js' %>"></script>

<!-- OPTIONAL SCRIPTS -->
<script src="<% static 'dist/js/demo.js' %>"></script>

{%
  if messages %}

  {% for message in messages %}

    <script type="text/javascript">
      toastr.options.progressBar = true;
      toastr.{{ message.tags }}('{{ message }}')
    </script>

  {% endfor %}

  {% endif %}

<script src="<% static 'plugins/Datatables/datatables.min.js' %>"></script>

<script>
(function ($) {
  $('.clickable-row').css('cursor', 'pointer');
  $(".clickable-row").click(function () {
    window.location = $(this).data("href");
  });
})(jQuery)

```

```
</script>

{% block morejs %}

{% endblock morejs %}

</body>

</html>

{% if page_obj.has_other_pages %}
<div aria-label="pagination">
<ul class="pagination justify-content-center">
    {% if page_obj.has_previous %}
        <li class="page-item">
            <a class="page-link" href="?page={{ page_obj.previous_page_number }}">Previous</a>
        </li>
    {% endif %}
    {% for page in page_obj.paginator.page_range %}
        {% if page_obj.number == page %}
            <li class="page-item active" aria-current="page">
                <span class="page-link">{{ page }}<span class="sr-only">(current)</span>
        
```

```
</span>
</li>
{%- else %}
<li class="page-item"><a class="page-link" href="?page={{ page }}">{{ page }}</a></li>
{%- endif %}

{%- endfor %}

{%- if page_obj.has_next %}
<li class="page-item">
<a class="page-link" href="?page={{ page_obj.next_page_number }}">Next</a>
</li>
{%- endif %}
</ul>
</div>

{%- endif %}
```

*from django.apps import AppConfig*

```
class FinanceConfig(AppConfig):
```

```
    name = "apps.finance"
```

```
    def ready(self):
```

```
import apps.finance.signals

from django.forms import inlineformset_factory, modelformset_factory

from .models import Invoice, InvoiceItem, Receipt

InvoiceItemFormset = inlineformset_factory(
    Invoice, InvoiceItem, fields=["description", "amount"], extra=1, can_delete=True
)

InvoiceReceiptFormSet = inlineformset_factory(
    Invoice,
    Receipt,
    fields=("amount_paid", "date_paid", "comment"),
    extra=0,
    can_delete=True,
)
Invoices = modelformset_factory(Invoice, exclude=(), extra=4)
```

```
from django.db import models
from django.urls import reverse
from django.utils import timezone

from apps.corecode.models import AcademicSession, AcademicTerm, StudentClass
```

```

from apps.students.models import Student

class Invoice(models.Model):
    student = models.ForeignKey(Student, on_delete=models.CASCADE)
    session = models.ForeignKey(AcademicSession, on_delete=models.CASCADE)
    term = models.ForeignKey(AcademicTerm, on_delete=models.CASCADE)
    class_for = models.ForeignKey(StudentClass, on_delete=models.CASCADE)
    balance_from_previous_term = models.IntegerField(default=0)
    status = models.CharField(
        max_length=20,
        choices=[("active", "Active"), ("closed", "Closed")],
        default="active",
    )

    class Meta:
        ordering = ["student", "term"]

    def __str__(self):
        return f"{self.student}"

    def balance(self):
        payable = self.total_amount_payable()
        paid = self.total_amount_paid()
        return payable - paid

    def amount_payable(self):
        items = InvoiceItem.objects.filter(invoice=self)
        total = 0
        for item in items:

```

```

total += item.amount
return total

def total_amount_payable(self):
    return self.balance_from_previous_term + self.amount_payable()

def total_amount_paid(self):
    receipts = Receipt.objects.filter(invoice=self)
    amount = 0
    for receipt in receipts:
        amount += receipt.amount_paid
    return amount

def get_absolute_url(self):
    return reverse("invoice-detail", kwargs={"pk": self.pk})

class InvoiceItem(models.Model):
    invoice = models.ForeignKey(Invoice, on_delete=models.CASCADE)
    description = models.CharField(max_length=200)
    amount = models.IntegerField()

class Receipt(models.Model):
    invoice = models.ForeignKey(Invoice, on_delete=models.CASCADE)
    amount_paid = models.IntegerField()
    date_paid = models.DateField(default=timezone.now)
    comment = models.CharField(max_length=200, blank=True)

    def __str__(self):

```

```
return f'Receipt on {self.date_paid}'
```

```
from django.db import models
```

```
from django.urls import reverse
```

```
from django.utils import timezone
```

```
from apps.corecode.models import AcademicSession, AcademicTerm, StudentClass
```

```
from apps.students.models import Student
```

```
class Invoice(models.Model):
```

```
    student = models.ForeignKey(Student, on_delete=models.CASCADE)
```

```
    session = models.ForeignKey(AcademicSession, on_delete=models.CASCADE)
```

```
    term = models.ForeignKey(AcademicTerm, on_delete=models.CASCADE)
```

```
    class_for = models.ForeignKey(StudentClass, on_delete=models.CASCADE)
```

```
    balance_from_previous_term = models.IntegerField(default=0)
```

```
    status = models.CharField(
```

```
        max_length=20,
```

```
        choices=[("active", "Active"), ("closed", "Closed")],
```

```
        default="active",
```

```
)
```

```
class Meta:
```

```
    ordering = ["student", "term"]
```

```
def __str__(self):
```

```
return f'{self.student}'
```

```

def balance(self):
    payable = self.total_amount_payable()
    paid = self.total_amount_paid()
    return payable - paid

def amount_payable(self):
    items = InvoiceItem.objects.filter(invoice=self)
    total = 0
    for item in items:
        total += item.amount
    return total

def total_amount_payable(self):
    return self.balance_from_previous_term + self.amount_payable()

def total_amount_paid(self):
    receipts = Receipt.objects.filter(invoice=self)
    amount = 0
    for receipt in receipts:
        amount += receipt.amount_paid
    return amount

def get_absolute_url(self):
    return reverse("invoice-detail", kwargs={"pk": self.pk})

class InvoiceItem(models.Model):
    invoice = models.ForeignKey(Invoice, on_delete=models.CASCADE)
    description = models.CharField(max_length=200)
    amount = models.IntegerField()

```

```

class Receipt(models.Model):
    invoice = models.ForeignKey(Invoice, on_delete=models.CASCADE)
    amount_paid = models.IntegerField()
    date_paid = models.DateField(default=timezone.now)
    comment = models.CharField(max_length=200, blank=True)

    def __str__(self):
        return f"Receipt on {self.date_paid}"

from django.urls import path

from .views import (
    InvoiceCreateView,
    InvoiceDeleteView,
    InvoiceDetailView,
    InvoiceListView,
    InvoiceUpdateView,
    ReceiptCreateView,
    ReceiptUpdateView,
    bulk_invoice,
)
urlpatterns = [
    path("list/", InvoiceListView.as_view(), name="invoice-list"),
    path("create/", InvoiceCreateView.as_view(), name="invoice-create"),
    path("<int:pk>/detail/", InvoiceDetailView.as_view(), name="invoice-detail"),
    path("<int:pk>/update/", InvoiceUpdateView.as_view(), name="invoice-update"),
]

```

```
path("<int:pk>/delete/", InvoiceDeleteView.as_view(), name="invoice-delete"),
path("receipt/create", ReceiptCreateView.as_view(), name="receipt-create"),
path(
    "receipt/<int:pk>/update/", ReceiptUpdateView.as_view(), name="receipt-
update"
),
path("bulk-invoice/", bulk_invoice, name="bulk-invoice"),
]
```

```
from django.contrib.auth.decorators import login_required
from django.contrib.auth.mixins import LoginRequiredMixin
from django.shortcuts import redirect, render
from django.urls import reverse_lazy
from django.views.generic import DetailView, ListView
from django.views.generic.edit import CreateView, DeleteView, UpdateView
```

```
from apps.students.models import Student
```

```
from .forms import InvoiceItemFormset, InvoiceReceiptFormSet, Invoices
from .models import Invoice, InvoiceItem, Receipt
```

```
class InvoiceListView(LoginRequiredMixin, ListView):
    model = Invoice
```

```
class InvoiceCreateView(LoginRequiredMixin, CreateView):
    model = Invoice
```

```

fields = "__all__"
success_url = "/finance/list"

def get_context_data(self, **kwargs):
    context = super(InvoiceCreateView, self).get_context_data(**kwargs)
    if self.request.POST:
        context["items"] = InvoiceItemFormset(
            self.request.POST, prefix="invoiceitem_set"
        )
    else:
        context["items"] = InvoiceItemFormset(prefix="invoiceitem_set")
    return context

def form_valid(self, form):
    context = self.get_context_data()
    formset = context["items"]
    self.object = form.save()
    if self.object.id != None:
        if form.is_valid() and formset.is_valid():
            formset.instance = self.object
            formset.save()
    return super().form_valid(form)

```

```

class InvoiceDetailView(LoginRequiredMixin, DetailView):
    model = Invoice
    fields = "__all__"

    def get_context_data(self, **kwargs):
        context = super(InvoiceDetailView, self).get_context_data(**kwargs)

```

```

context["receipts"] = Receipt.objects.filter(invoice=self.object)
context["items"] = InvoiceItem.objects.filter(invoice=self.object)
return context

class InvoiceUpdateView(LoginRequiredMixin, UpdateView):
    model = Invoice
    fields = ["student", "session", "term", "class_for", "balance_from_previous_term"]

    def get_context_data(self, **kwargs):
        context = super(InvoiceUpdateView, self).get_context_data(**kwargs)
        if self.request.POST:
            context["receipts"] = InvoiceReceiptFormSet(
                self.request.POST, instance=self.object
            )
            context["items"] = InvoiceItemFormset(
                self.request.POST, instance=self.object
            )
        else:
            context["receipts"] = InvoiceReceiptFormSet(instance=self.object)
            context["items"] = InvoiceItemFormset(instance=self.object)
        return context

    def form_valid(self, form):
        context = self.get_context_data()
        formset = context["receipts"]
        itemsformset = context["items"]
        if form.is_valid() and formset.is_valid() and itemsformset.is_valid():
            form.save()
            formset.save()

```

```
        itemsformset.save()
    return super().form_valid(form)
```

```
class InvoiceDeleteView(LoginRequiredMixin, DeleteView):
    model = Invoice
    success_url = reverse_lazy("invoice-list")
```

```
class ReceiptCreateView(LoginRequiredMixin, CreateView):
    model = Receipt
    fields = ["amount_paid", "date_paid", "comment"]
    success_url = reverse_lazy("invoice-list")
```

```
def form_valid(self, form):
    obj = form.save(commit=False)
    invoice = Invoice.objects.get(pk=self.request.GET["invoice"])
    obj.invoice = invoice
    obj.save()
    return redirect("invoice-list")
```

```
def get_context_data(self, **kwargs):
    context = super(ReceiptCreateView, self).get_context_data(**kwargs)
    invoice = Invoice.objects.get(pk=self.request.GET["invoice"])
    context["invoice"] = invoice
    return context
```

```
class ReceiptUpdateView(LoginRequiredMixin, UpdateView):
    model = Receipt
```

```
fields = ["amount_paid", "date_paid", "comment"]  
success_url = reverse_lazy("invoice-list")  
  
class ReceiptDeleteView(LoginRequiredMixin, DeleteView):  
    model = Receipt  
    success_url = reverse_lazy("invoice-list")
```

```
@login_required  
def bulk_invoice(request):  
    return render(request, "finance/bulk_invoice.html")
```

"""

*WSGI config for school\_app project.*

*It exposes the WSGI callable as a module-level variable named ``application``.*

*For more information on this file, see*

*<https://docs.djangoproject.com/en/3.0/howto/deployment/wsgi/>*

"""

```
import os
```

```
from django.core.wsgi import get_wsgi_application
```

```
os.environ.setdefault("DJANGO_SETTINGS_MODULE", "school_app.settings")
```

```
application = get_wsgi_application()
```

```
"""
```

*Django settings for school\_app project.*

*Generated by 'django-admin startproject' using Django 3.0.5.*

*For more information on this file, see*

<https://docs.djangoproject.com/en/3.0/topics/settings/>

*For the full list of settings and their values, see*

<https://docs.djangoproject.com/en/3.0/ref/settings/>

```
"""
```

```
import os
```

```
# Build paths inside the project like this: os.path.join(BASE_DIR, ...)
```

```
BASE_DIR = os.path.dirname(os.path.dirname(os.path.abspath(__file__)))
```

```
# Quick-start development settings - unsuitable for production
```

```
# See https://docs.djangoproject.com/en/3.0/howto/deployment/checklist/
```

```
# SECURITY WARNING: keep the secret key used in production secret!
```

```
SECRET_KEY = "__$1ud47e&nys05h5o3fwnqu4+hfqcply9h$k*h2s34)hn5@nc"
```

```
# SECURITY WARNING: don't run with debug turned on in production!
```

```
DEBUG = True
```

```
ALLOWED_HOSTS = []
```

*# Application definition*

```
INSTALLED_APPS = [
```

```
    "django.contrib.auth",
    "django.contrib.contenttypes",
    "django.contrib.sessions",
    "django.contrib.messages",
    "django.contrib.staticfiles",
    "django.contrib.humanize",
    "widget_tweaks",
    "apps.corecode",
    "apps.students",
    "apps.staffs",
    "apps.finance",
    "apps.result",
```

```
]
```

```
MIDDLEWARE = [
```

```
    "django.middleware.security.SecurityMiddleware",
    "django.contrib.sessions.middleware.SessionMiddleware",
    "django.middleware.common.CommonMiddleware",
    "django.middleware.csrf.CsrfViewMiddleware",
    "django.contrib.auth.middleware.AuthenticationMiddleware",
    "django.contrib.messages.middleware.MessageMiddleware",
    "django.middleware.clickjacking.XFrameOptionsMiddleware",
```

```
        "apps.corecode.middleware.SiteWideConfigs",
    ]
```

```
ROOT_URLCONF = "school_app.urls"
```

```
TEMPLATES = [
{
    "BACKEND": "django.template.backends.django.DjangoTemplates",
    "DIRS": [
        os.path.join(BASE_DIR, "templates"),
    ],
    "APP_DIRS": True,
    "OPTIONS": {
        "context_processors": [
            "django.template.context_processors.debug",
            "django.template.context_processors.request",
            "django.contrib.auth.context_processors.auth",
            "django.contrib.messages.context_processors.messages",
            "apps.corecode.context_processors.site_defaults",
        ],
    },
},
]
```

```
WSGI_APPLICATION = "school_app.wsgi.application"
```

```
# Database
# https://docs.djangoproject.com/en/3.0/ref/settings/#databases
```

```
DATABASES = {  
    "default": {  
        "ENGINE": "django.db.backends.sqlite3",  
        "NAME": os.path.join(BASE_DIR, "db.sqlite3"),  
    }  
}  
  
# Password validation  
# https://docs.djangoproject.com/en/3.0/ref/settings/#auth-password-validators
```

```
AUTH_PASSWORD_VALIDATORS = [  
    {  
        "NAME":  
            "django.contrib.auth.password_validation.UserAttributeSimilarityValidator",  
        },  
        {  
            "NAME": "django.contrib.auth.password_validation.MinimumLengthValidator",  
        },  
        {  
            "NAME":  
                "django.contrib.auth.password_validation.CommonPasswordValidator",  
            },  
            {  
                "NAME":  
                    "django.contrib.auth.password_validation.NumericPasswordValidator",  
            },  
        ]
```

```
# Internationalization  
# https://docs.djangoproject.com/en/3.0/topics/i18n/
```

LANGUAGE\_CODE = "en-us"

TIME\_ZONE = "UTC"

USE\_I18N = True

USE\_L10N = True

USE\_TZ = True

```
# Static files (CSS, JavaScript, Images)  
# https://docs.djangoproject.com/en/3.0/howto/static-files/
```

DATA\_UPLOAD\_MAX\_NUMBER\_FIELDS = 10240

STATIC\_URL = "/static/"

STATICFILES\_DIRS = (os.path.join(BASE\_DIR, "static"),)

STATIC\_ROOT = os.path.join(BASE\_DIR, "staticfiles")

MEDIA\_ROOT = os.path.join(BASE\_DIR, "media")

MEDIA\_URL = "/media/"

LOGIN\_REDIRECT\_URL = "/"

LOGOUT\_REDIRECT\_URL = "/"

SESSION\_SAVE\_EVERY\_REQUEST = True

SESSION\_EXPIRE\_AT\_BROWSER\_CLOSE = True

SESSION\_COOKIE\_AGE = 10800

LOGGING = {

    "version": 1,

    "disable\_existing\_loggers": False,

    "formatters": {

        "verbose": {

            "format": "{levelname} {asctime} {message}",

            "style": "{},"

        },

    },

    "handlers": {

        "file": {

            "level": "INFO",

            "class": "logging.handlers.TimedRotatingFileHandler",

            "when": "W6",

            "interval": 4,

            "backupCount": 3,

            "encoding": "utf8",

            "filename": os.path.join(BASE\_DIR, "debug.log"),

            "formatter": "verbose",

```
    },
},
"loggers": {
    "django": {
        "handlers": ["file"],
        "level": "INFO",
        "propagate": True,
    },
},
}
```

DEFAULT\_AUTO\_FIELD = "django.db.models.AutoField"

# Site Default values

"""\nnewapp URL Configuration

The `urlpatterns` list routes URLs to views. For more information please see:

<https://docs.djangoproject.com/en/3.0/topics/http/urls/>

Examples:

Function views

1. Add an import: `from my_app import views`
2. Add a URL to urlpatterns: `path('/', views.home, name='home')`

Class-based views

1. Add an import: `from other_app.views import Home`
2. Add a URL to urlpatterns: `path('/', Home.as_view(), name='home')`

Including another URLconf

1. Import the `include()` function: `from django.urls import include, path`

2. Add a URL to `urlpatterns`: `path('blog/', include('blog.urls'))`

"""

```
from django.conf import settings
from django.conf.urls.static import static
from django.urls import include, path
```

```
urlpatterns = [
    path("accounts/", include("django.contrib.auth.urls")),
    path("", include("apps.corecode.urls")),
    path("student/", include("apps.students.urls")),
    path("staff/", include("apps.staffs.urls")),
    path("finance/", include("apps.finance.urls")),
    path("result/", include("apps.result.urls")),
] + static(settings.MEDIA_URL, document_root=settings.MEDIA_ROOT)
```

{

  "meta": {

    "hash": {

      "sha256":

      "7e6dca07b964c2888324e576ba6c1bc240d74a27b75619fc88bca2ee3979baf8"

    },

    "pipfile-spec": 6,

    "requires": {

      "python\_version": "3.10"

    },

    "sources": [

      {

```
        "name": "pypi",
        "url": "https://pypi.org/simple",
        "verify_ssl": true
    },
]
},
"default": {
    "asgiref": {
        "hashes": [
            "sha256:2f8abc20f7248433085eda803936d98992f1343ddb022065779f37c5da0181d
0",
            "sha256:88d59c13d634dcffe0510be048210188edd79aec6a6c9028cdad6f31d730a9
",
            ],
        "markers": "python_version >= '3.7'",
        "version": "==3.5.0"
    },
    "django": {
        "hashes": [
            "sha256:110fb58fb12eca59e072ad59fc42d771cd642dd7a2f2416582aa9da7a8ef954a"
,
            "sha256:996495c58bff749232426c88726d8cd38d24c94d7c1d80835aafffa9bc52985a
",
            ],
        "index": "pypi",
        "version": "==4.0.2"
    }
}
```

```
        },
        "sqlparse": {
            "hashes": [
                "sha256:0c00730c74263a94e5a9919ade150dfc3b19c574389985446148402998287da
                e",
                "sha256:48719e356bb8b42991bdbb1e8b83223757b93789c00910a616a071910ca4a6
                4d"
            ],
            "markers": "python_version >= '3.5'",
            "version": "==0.4.2"
        }
    },
    "develop": {}
}
```

appdirs==1.4.4  
asgiref==3.4.1  
black==21.7b0  
click==8.0.1  
colorama==0.4.4  
Django==3.2.5  
django-widget-tweaks==1.4.8  
flake8==3.9.2  
isort==5.9.2  
mccabe==0.6.1  
mypy-extensions==0.4.3

```
pathspec==0.8.1
Pillow==8.3.2
pycodestyle==2.7.0
pyflakes==2.3.1
pytz==2021.1
regex==2021.7.6
sqlparse==0.4.2
tomli==1.0.4
```

```
#!/usr/bin/env python
"""Django's command-line utility for administrative tasks."""
import os
import sys

def main():
    os.environ.setdefault("DJANGO_SETTINGS_MODULE", "school_app.settings")
    try:
        from django.core.management import execute_from_command_line
    except ImportError as exc:
        raise ImportError(
            "Couldn't import Django. Are you sure it's installed and "
            "available on your PYTHONPATH environment variable? Did you "
            "forget to activate a virtual environment?"
        ) from exc
    execute_from_command_line(sys.argv)
```

```
if __name__ == "__main__":
    main()

# Generated by Django 3.0.5 on 2020-05-06 14:19

from django.db import migrations, models

class Migration(migrations.Migration):

    initial = True

    dependencies = []

    operations = [
        migrations.CreateModel(
            name="AcademicSession",
            fields=[
                (
                    "id",
                    models.AutoField(
                        auto_created=True,
                        primary_key=True,
                        serialize=False,
                        verbose_name="ID",
                    ),
                ),
                ("name", models.CharField(max_length=200, unique=True)),
                ("current", models.BooleanField()),
            ],
        )
    ]

```

```
],
options={

    "ordering": ["-name"],

},
),
migrations.CreateModel(
    name="AcademicTerm",
    fields=[

        (
            "id",
            models.AutoField(
                auto_created=True,
                primary_key=True,
                serialize=False,
                verbose_name="ID",
            ),
        ),
        (
            "name", models.CharField(max_length=20, unique=True)),
            ("current", models.BooleanField()),
        ],
options={

    "ordering": ["name"],

},
),
migrations.CreateModel(
    name="SiteConfig",
    fields=[

        (
            "id",
            models.AutoField(

```

```
        auto_created=True,
        primary_key=True,
        serialize=False,
        verbose_name="ID",
    ),
),
("key", models.SlugField()),
("value", models.CharField(max_length=200)),
],
),
migrations.CreateModel(
    name="StudentClass",
    fields=[
        (
            "id",
            models.AutoField(
                auto_created=True,
                primary_key=True,
                serialize=False,
                verbose_name="ID",
            ),
        ),
        ("name", models.CharField(max_length=200, unique=True)),
    ],
options={
    "verbose_name": "Class",
    "verbose_name_plural": "Classes",
    "ordering": ["name"],
},
),
```

```

migrations.CreateModel(
    name="Subject",
    fields=[
        (
            "id",
            models.AutoField(
                auto_created=True,
                primary_key=True,
                serialize=False,
                verbose_name="ID",
            ),
        ),
        (
            "name",
            models.CharField(max_length=200, unique=True),
        ],
        options={
            "ordering": ["name"],
        },
    ),
)

```

## AUTO2022

*# Generated by Django 3.0.5 on 2020-05-06 14:20*

```

from django.db import migrations
from django.contrib.auth.models import User

```

```

def default_site_config(apps, schema_editor):
    """Default site configurations"""

```

```
User.objects.create_superuser("admin", "admin@schoolapp.com", "admin123")
```

```
Config = apps.get_model("corecode", "SiteConfig")
Config.objects.bulk_create(
    [
        Config(key="school_name", value="My School"),
        Config(key="school_slogan", value="A great school"),
        Config(key="school_address", value="Lagos, Nigeria"),
    ]
)
```

```
Session = apps.get_model("corecode", "AcademicSession")
Session.objects.bulk_create(
    [
        Session(name="2019/2020", current=True),
    ]
)
```

```
Term = apps.get_model("corecode", "AcademicTerm")
Term.objects.bulk_create(
    [
        Term(name="1st Term", current=True),
        Term(name="2nd Term", current=False),
        Term(name="3rd Term", current=False),
    ]
)
```

```
Subject = apps.get_model("corecode", "Subject")
Subject.objects.bulk_create()
```

```

[

    Subject(name="Mathematics"),
    Subject(name="English"),
]

)

StudentClass = apps.get_model("corecode", "StudentClass")
StudentClass.objects.bulk_create(
[
    StudentClass(name="JSS 1"),
    StudentClass(name="JSS 2"),
]
)

class Migration(migrations.Migration):

    dependencies = [
        ("corecode", "0001_initial"),
    ]

    operations = [
        migrations.RunPython(default_site_config),
    ]
}

# Generated by Django 3.0.5 on 2020-05-06 14:19

from django.db import migrations, models
import django.db.models.deletion

```

```

class Migration(migrations.Migration):

    initial = True

    dependencies = [
        ("students", "0001_initial"),
        ("corecode", "0001_initial"),
    ]

    operations = [
        migrations.CreateModel(
            name="Result",
            fields=[
                (
                    "id",
                    models.AutoField(
                        auto_created=True,
                        primary_key=True,
                        serialize=False,
                        verbose_name="ID",
                    ),
                ),
                (
                    "test_score",
                    models.IntegerField(default=0),
                ),
                (
                    "exam_score",
                    models.IntegerField(default=0),
                ),
                (
                    "current_class",
                    models.ForeignKey(
                        on_delete=django.db.models.deletion.CASCADE,
                        to="corecode.StudentClass",
                    ),
                ),
            ],
        )
    ]

```

```
        ),  
    ),  
    (  
        "session",  
        models.ForeignKey(  
            on_delete=django.db.models.deletion.CASCADE,  
            to="corecode.AcademicSession",  
        ),  
    ),  
    (  
        "student",  
        models.ForeignKey(  
            on_delete=django.db.models.deletion.CASCADE,  
            to="students.Student",  
        ),  
    ),  
    (  
        "subject",  
        models.ForeignKey(  
            on_delete=django.db.models.deletion.CASCADE,  
            to="corecode.Subject",  
        ),  
    ),  
    (  
        "term",  
        models.ForeignKey(  
            on_delete=django.db.models.deletion.CASCADE,  
            to="corecode.AcademicTerm",  
        ),  
    ),
```

```
],  
options={  
    "ordering": ["subject"],  
},  
,  
]  
]
```

## **CONCLUSION:**

The project of “Fee Management System” is designed in order to reduce the maintaining bulk of records of all student fees details of who study in an Educational Institution. Inserting, retrieving and updating the fees details of a student are easy when it is compared to the manual register and storing. Maintaining the project is also easy which can be easily understandable. Maintaining the database is manageable.