

Project Report

# Finance Management System

By

Nilay Yadav      225811382

Under the Guidance of  
Dr. Gopalakrishnan  
(April-2024)

**Department of Information Technology  
Manipal Institute Of Technology  
Yelhanka, Bangalore**

# **1. Introduction**

The Finance Management System is a web-based application developed using Django, HTML, CSS, and JavaScript, aimed at providing users with tools to manage their finances efficiently. It includes features such as expense tracking, transaction history, and data visualization to help users make informed financial decisions.

# **2. Objectives**

- Create a user-friendly interface for efficient financial management.
- Implement features for inputting transaction details and viewing spending history.
- Utilize data visualization techniques to analyze spending patterns.
- Ensure data security and privacy using Django's built-in security features.

### **3. Technologies Used**

- Framework: Django
- Frontend: HTML, CSS, JavaScript
- Database: SQLite3
- Additional Libraries: Chart.js for data visualization

### **4. System Architecture**

The Finance Management System follows a Model-View-Template (MVT) architecture provided by Django. HTML templates render dynamic content served by Django views, which interact with SQLite3 database models to retrieve and store user data.

## **5. Features Implemented**

### **Login Page**

- Provides a form for users to enter their credentials (username and password).
- Authenticates users against the database and grants access upon successful login.
- Implements error handling for incorrect credentials.

### **User Registration Page**

- Allows new users to create an account by providing necessary details (username, name, email, password).
- Validates user input to ensure data integrity and uniqueness.
- Stores user information securely in the SQLite3 database upon registration.

### **Dashboard Page**

- Displays a graph of expenses over the last 14 days.
- Provides toggles to switch between total spends of today, this month, and last month.
- Shows the last 10 transactions in a list format.

### **Transaction Form Page**

- Allows users to input transaction details including "Paid to", "Amount", "Reason", and "Remarks".
- Implements form validation to ensure data integrity.
- Stores transaction details in the SQLite3 database upon submission.

## All Spendings Page

- Lists all transactions in a tabular format.
- Implements pagination for better user experience with large transaction histories.

## Profile Page

- Displays user information including name, username, email, and date joined.
- Allows users to update their profile information.

# 6. Implementation Details

## Frontend

- Utilized HTML templates with Django template language for dynamic content rendering.
- Styled using CSS for a clean and intuitive user interface.
- Integrated JavaScript for client-side interactions and data manipulation.

## Backend

- Developed using Django framework to handle server-side logic and request routing.
- Utilized SQLite3 as the database management system for data storage.

## Database Schema

- Designed database tables using Django's ORM to represent user information and transaction details.

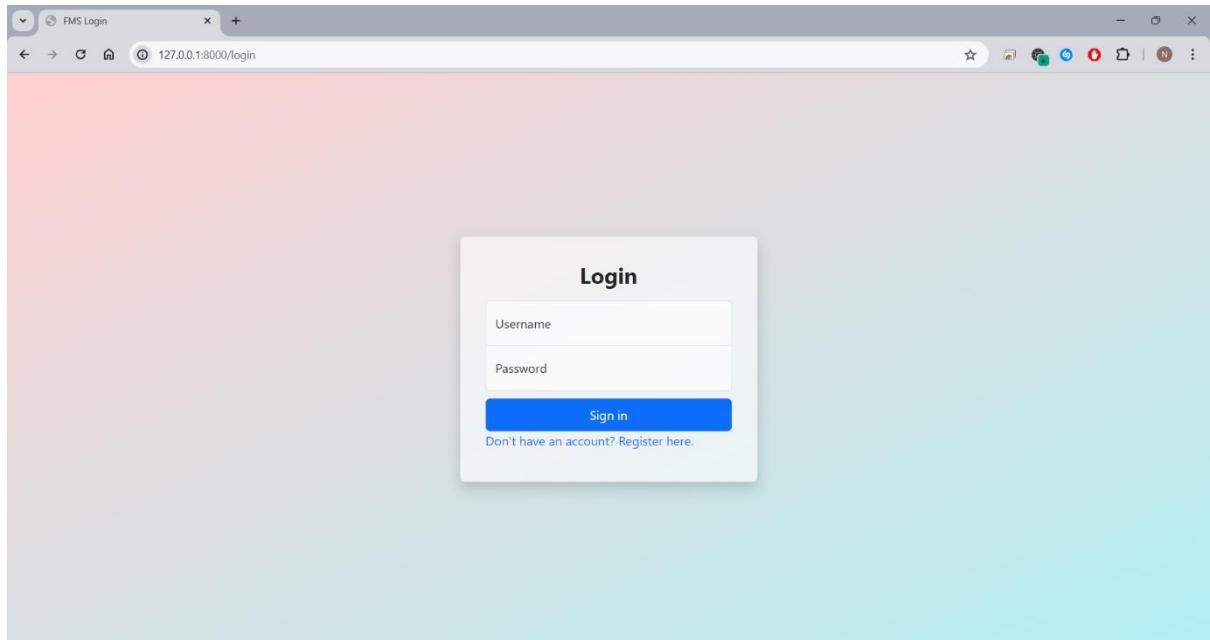
## 7. Conclusion

The Finance Management System, built on Django with HTML, CSS, and JavaScript, offers users a robust platform for managing their finances effectively. With its intuitive interface, comprehensive features, and secure data handling, it aims to empower users to make informed financial decisions.

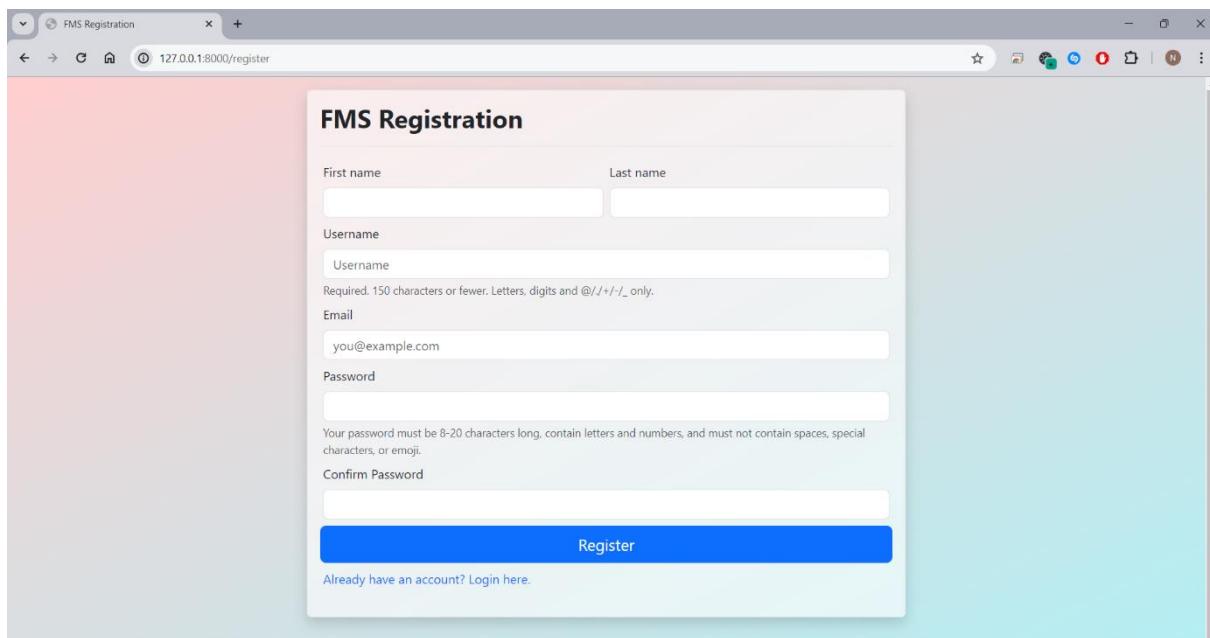
## 8. Future Enhancements

- Add data encryption for sensitive information protection.
- Integrate budgeting and forecasting features to assist users in financial planning.
- Incorporate Normalisation of database

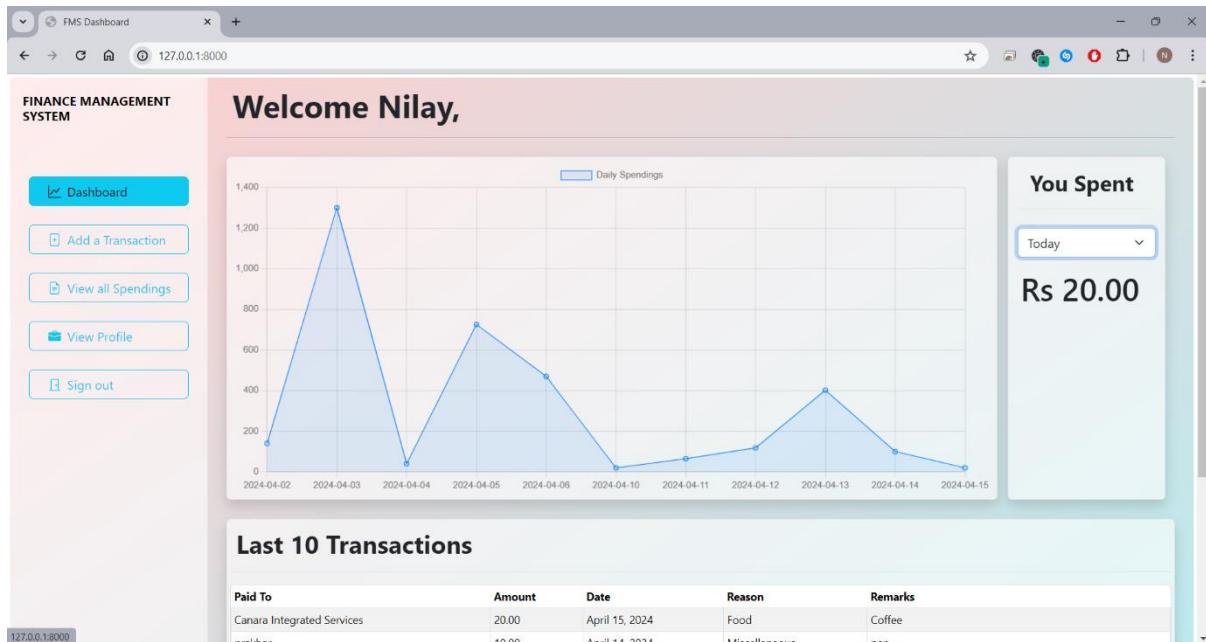
# Final Output:



The screenshot shows a web browser window titled "FMS Login" with the URL "127.0.0.1:8000/login". The page features a light gray background with a central white login form. The form has a title "Login" at the top. It contains two input fields: "Username" and "Password", both with placeholder text. Below the fields is a blue "Sign in" button. At the bottom of the form, there is a link "Don't have an account? Register here."



The screenshot shows a web browser window titled "FMS Registration" with the URL "127.0.0.1:8000/register". The page has a light gray background with a central white registration form. The form is titled "FMS Registration". It includes fields for "First name" and "Last name" (both with placeholder text), "Username" (placeholder "Username"), and "Email" (placeholder "you@example.com"). Below the email field is a note: "Required. 150 characters or fewer. Letters, digits and @/./-/\_. only." The form also has fields for "Password" and "Confirm Password" (both with placeholder text). A note below the password fields states: "Your password must be 8-20 characters long, contain letters and numbers, and must not contain spaces, special characters, or emoji." At the bottom of the form is a blue "Register" button. There is also a link "Already have an account? Login here."



FMS Add Transaction

127.0.0.1:8000/add-spending

# Add Transaction

FINANCE MANAGEMENT SYSTEM

Dashboard

Add a Transaction

View all Spendings

View Profile

Sign out

Paid to

Amount

Reason

Choose...

Remarks (Optional)

Submit Transaction

# Code:

The screenshot shows a code editor interface with two tabs open, both titled "urls.py".

**Top Tab (urls.py):**

```
1  """
2  URL configuration for FMS project.
3
4  The 'urlpatterns' list routes URLs to views. For more information please see:
5      https://docs.djangoproject.com/en/5.0/topics/http/urls/
6  Examples:
7      Function views
8          1. Add an import: from my_app import views
9             2. Add a URL to urlpatterns: path('', views.home, name='home')
10            Class-based views
11                1. Add an import: from other_app.views import Home
12                2. Add a URL to urlpatterns: path('', Home.as_view(), name='home')
13            Including another URLconf
14                1. Import the include() function: from django.urls import include, path
15                2. Add a URL to urlpatterns: path('blog/', include('blog.urls'))
16
17    from django.contrib import admin
18    from django.urls import path, include
19
20    urlpatterns = [
21        path('admin/', admin.site.urls),
22        path('', include('home.urls')),
23    ]
```

**Bottom Tab (views.py):**

```
1  messages.info(request,"Invalid credentials")
2  return render(request,'login.html')
3
4  def logoutUser(request):
5      Logout(request)
6      return redirect('/login')
7
8  def add_spending(request):
9      if request.user.is_anonymous:
10          return redirect('/login')
11      if request.method=="POST":
12          FMSuser=request.POST.get("FMSuser")
13          Paidto=request.POST.get("Paidto")
14          Amount=request.POST.get("Amount")
15          Reason=request.POST.get("Reason")
16          Remarks=request.POST.get("Remarks")
17          transaction=addSpending(Paidto=Paidto, Amount=Amount, FMSuser=FMSuser, Date=datetime.today(), Reason=Reason, Remarks=Remarks)
18          transaction.save()
19          messages.info(request,"Transaction added succesfully")
20
21  def register(request):
22      if not(request.user.is_anonymous):
23          return redirect("/")
24      form = CreateuserForm()
25      if request.method=="POST":
26          form = CreateuserForm(request.POST)
27          if form.is_valid():
28              form.save()
29              return redirect('/login')
30          else:
31              messages.info(request,"Kindly recheck your details or try a different username")
```

**Screenshot 1: Code Editor - views.py**

```

File Edit Selection View Go Run Terminal Help ← → ⌘ FMS
EXPLORER urls.py FMS urls.py home views.py M models.py forms.py index.html register.html M login.html spendings.html
FMS .vscode _pycache_ __init__.py asgi.py settings.py urls.py wsgi.py
home _pycache_ migrations __init__.py admin.py apps.py forms.py tests.py urls.py views.py
static templates add-spending.html base.html index.html login.html my-profile.html register.html spendings.html
db.sqlite3 manage.py
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
[15/Apr/2024 15:32:40] "GET /login HTTP/1.1" 200 3161
Ln 16, Col 1 Spaces: 4 UTF-8 CRLF {} Python 3.12.2 64-bit

```

**Screenshot 2: Code Editor - models.py**

```

File Edit Selection View Go Run Terminal Help ← → ⌘ FMS
EXPLORER urls.py FMS urls.py home views.py M models.py M forms.py index.html register.html M login.html spendings.html
FMS .vscode _pycache_ __init__.py asgi.py settings.py urls.py wsgi.py
home _pycache_ migrations __init__.py admin.py apps.py forms.py tests.py urls.py views.py
static templates add-spending.html base.html index.html login.html my-profile.html register.html spendings.html
db.sqlite3 manage.py
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
[15/Apr/2024 15:32:40] "GET /login HTTP/1.1" 200 3161
Ln 13, Col 1 Spaces: 4 UTF-8 CRLF {} Python 3.12.2 64-bit

```

The screenshot shows two instances of the Visual Studio Code (VS Code) interface side-by-side, both displaying code for a Django application named 'FMS'.

**Top Window (forms.py):**

- Explorer:** Shows the project structure with files like urls.py, forms.py, models.py, views.py, and various template files (index.html, register.html, login.html, spending.html).
- Editor:** Displays the contents of forms.py:

```
home > forms.py > ...
1  from django.contrib.auth.forms import UserCreationForm
2  from django.contrib.auth.models import User
3
4
5  class CreateUserForm(UserCreationForm):
6      class Meta:
7          model = User
8          fields = ['first_name', 'last_name', 'username', 'email', 'password1', 'password2']
```
- Terminal:** Shows the command: [15/Apr/2024 15:32:40] "GET /login HTTP/1.1" 200 3161
- Status Bar:** Python 3.12.2 64-bit

**Bottom Window (index.html):**

- Explorer:** Shows the project structure with files like urls.py, forms.py, models.py, views.py, and various template files (index.html, register.html, login.html, spending.html).
- Editor:** Displays the contents of index.html:

```
<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="utf-8" />
        <meta name="viewport" content="width=device-width, initial-scale=1" />
        <title>FMS Dashboard</title>
        <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-QWTKZYjpPEjISvSWaRU9OFe" type="text/css"/>
        <script src="https://cdn.jsdelivr.net/npm/chart.js"></script>
    <style>
        .backg {
            background: rgba(255, 255, 255, 0.6);
        }
        #grad {
            background-image: linear-gradient(to bottom right, rgba(255, 207, 207, 1), rgba(179, 239, 244, 1));
        }
        .responsive-img {
            max-width: 100%;
            height: auto;
        }
        .bi {
            display: inline-block;
            width: 1rem;
            height: 1rem;
        }
        @media (min-width: 768px) {
            .sidebar .offcanvas-lg {
                position: -webkit-sticky;
                position: sticky;
                top: 48px;
            }
        }
        .navbar-search {
```
- Terminal:** Shows the command: [15/Apr/2024 15:32:40] "GET /login HTTP/1.1" 200 3161
- Status Bar:** Python 3.12.2 64-bit

The image shows two instances of the Visual Studio Code (VS Code) interface, each displaying a different section of the same file, `index.html`. Both instances have a dark theme.

**Top Instance:**

```
33     .navbar-search {  
34         display: block;  
35     }  
36  
37     .sidebar .nav-link {  
38         font-size: 0.875rem;  
39         font-weight: 500;  
40     }  
41  
42     .sidebar .nav-link.active {  
43         color: #2470dc;  
44     }  
45  
46     .sidebar-heading {  
47         font-size: 0.75rem;  
48     }  
49  
50     .navbar-brand {  
51         padding-top: 0.75rem;  
52         padding-bottom: 0.75rem;  
53     }  
54  
55     .navbar .form-control {  
56         padding: 0.75rem 1rem;  
57     }  
58  
59     .bd-placeholder-img {  
60         font-size: 1.125rem;  
61         text-anchor: middle;  
62         -webkit-user-select: none;  
63         -moz-user-select: none;  
64         user-select: none;  
65     }
```

**Bottom Instance:**

```
67     @media (min-width: 768px) {  
68         .bd-placeholder-img-lg {  
69             font-size: 3.5rem;  
70         }  
71     }  
72  
73     .b-example-divider {  
74         width: 100%;  
75         height: 3rem;  
76         background-color: rgba(0, 0, 0, 0.1);  
77         border: solid rgba(0, 0, 0, 0.15);  
78         border-width: 1px 0;  
79         box-shadow: inset 0 0.5em 1.5em rgba(0, 0, 0, 0.1), inset 0 0.125em 0.5em rgba(0, 0, 0, 0.15);  
80     }  
81  
82     .b-example-vr {  
83         flex-shrink: 0;  
84         width: 1.5rem;  
85         height: 100vh;  
86     }  
87  
88     .bi {  
89         vertical-align: -0.125em;  
90         fill: currentColor;  
91     }  
92  
93     .nav-scroller {  
94         position: relative;  
95         z-index: 2;  
96         height: 2.75rem;  
97         overflow-y: hidden;  
98     }  
99 }
```

Both instances show the same code content, with minor differences in line numbers and styling. The bottom instance includes additional CSS rules for media queries and scrollable navigation components.

The screenshot displays a developer's environment with two main browser windows side-by-side, each showing a different state of a web application's source code.

**Left Browser Tab:**

- Title Bar:** File Edit Selection View Go Run Terminal Help
- Explorer:** Shows project structure under FMS, including files like urls.py, views.py, models.py, forms.py, and index.html.
- Code Editor:** The index.html file is open, showing custom CSS styles for a navigation bar and a dropdown menu.
- Bottom Status:** PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, PORTS. The TERMINAL tab shows the command: [15/Apr/2024 15:32:40] "GET /login HTTP/1.1" 200 3161.
- Bottom Footer:** In 242, Col 1 | Spaces: 2 | UTF-8 | CRLF | Django HTML

**Right Browser Tab:**

- Title Bar:** File Edit Selection View Go Run Terminal Help
- Explorer:** Shows project structure under FMS, including files like urls.py, views.py, models.py, forms.py, and index.html.
- Code Editor:** The index.html file is open, showing a different set of custom CSS styles, including imports for Bootstrap icons and a different structure for the navigation bar.
- Bottom Status:** PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, PORTS. The TERMINAL tab shows the command: [15/Apr/2024 15:32:40] "GET /login HTTP/1.1" 200 3161.
- Bottom Footer:** In 242, Col 1 | Spaces: 2 | UTF-8 | CRLF | Django HTML

**Screenshot 1: VS Code Editor - FMS Project**

The screenshot shows the VS Code interface for a Django project named "FMS". The Explorer sidebar on the left lists files and folders including ".vscode", "FMS", "home", "migrations", "models.py", "spending.py", "static", "templates", and database files "db.sqlite3" and "manage.py". The "templates" folder contains "index.html", "login.html", "my-profile.html", "register.html", and "spendings.html". The "index.html" file is open in the main editor area, showing HTML code for a navigation bar and a main content section.

```

<a class="nav-item">
  <a type="button" class="btn btn-outline-info d-flex align-items-center gap-2 px-4 mb-4" href="/add-spending">
    <svg class="bi">
      <use xlink:href="#file-plus" />
    </svg>Add a Transaction
  </a>
</li>
<li class="nav-item">
  <a type="button" class="btn btn-outline-info d-flex align-items-center gap-2 px-4 mb-4" href="/spendings">
    <svg class="bi">
      <use xlink:href="#file-earmark-text" />
    </svg>View all Spendings
  </a>
</li>
<li class="nav-item">
  <a type="button" class="btn btn-outline-info d-flex align-items-center gap-2 px-4 mb-4" href="/my-profile">
    <svg class="bi">
      <use xlink:href="#briefcase-fill" />
    </svg>View Profile
  </a>
</li>
<li class="nav-item">
  <a type="button" class="btn btn-outline-info d-flex align-items-center gap-2 px-4 mb-4" href="/logout">
    <svg class="bi">
      <use xlink:href="#door-closed" />
    </svg>Sign out
  </a>
</li>
</ul>
</div>

```

**Screenshot 2: VS Code Editor - FMS Project**

This screenshot shows the same VS Code session after some changes have been made to the "index.html" template. The code now includes a canvas element for a chart and a dropdown menu for selecting time periods.

```

<div class="row d-flex my-4 w-100">
  <div class="col-md-10">
    <canvas class="border rounded shadow p-2 responsive-img backg" id="transactionChart" width="900" height="400"></canvas>
  </div>
<div class="container-fluid col-md-2 col-sm-3 border backg rounded shadow align-items-center">
  <h3 class="border-bottom p-3 mb-4">You Spent:</h3>
  <select class="form-select" name="timerange" id="timePeriod" onchange="calculateMoneySpent()">
    <option value="--Choose--">Choose...</option>
    <option value="today">Today</option>
    <option value="this_month">This Month</option>
    <option value="past_month">Past Month</option>
  </select>
  <h1 class="mt-3 px-1">Rs <span id="moneySpent"></span></h1>
</div>
</div>

```

**VS Code Editor Screenshot 1 (Top)**

The screenshot shows the VS Code interface with two tabs open: `index.html` and `register.html`. The `index.html` tab is active, displaying Django template code for a transaction history page. The code includes a table with columns for Paid To, Amount, Date, Reason, and Remarks. A script block at the bottom handles spending calculations based on a selected time period (today, this month, past month, or default) and updates a chart.

```

<div class="d-flex justify-content-between flex-wrap align-items-center p-2 mb-3 border-bottom">
    <h2>Last 10 Transactions</h2>
</div>
<table class="table-responsive small border rounded mb-2">
    <thead>
        <tr>
            <th>Paid To</th>
            <th>Amount</th>
            <th>Date</th>
            <th>Reason</th>
            <th>Remarks</th>
        </tr>
    </thead>
    <tbody>
        {% for instance in data %}
            <tr>
                <td>{{ instance.fields.Paidto }}</td>
                <td>{{ instance.fields.Amount }}</td>
                <td>{{ instance.fields.Date }}</td>
                <td>{{ instance.fields.Reason }}</td>
                <td>{{ instance.fields.Remarks }}</td>
            </tr>
        {% endfor %}
    </tbody>
</table>
</div>
</div>
</div>

```

**VS Code Editor Screenshot 2 (Bottom)**

This screenshot shows the same VS Code interface with the `index.html` tab still active. The code has been partially modified, showing only the beginning of the script block and the chart configuration. The rest of the template code is visible below the fold.

```

function calculateMoneySpent() {
    var selectBox = document.getElementById('timePeriod');
    var selectedValue = selectBox.options[selectBox.selectedIndex].value
    var moneySpent = 0

    switch (selectedValue) {
        case 'today':
            moneySpent = {{tsum}}
            break
        case 'this_month':
            moneySpent = {{msum}}
            break
        case 'past_month':
            moneySpent = {{lsum}}
            break
        default:
            moneySpent = 0
    }

    document.getElementById('moneySpent').innerHTML = moneySpent.toFixed(2)
}

var labels = JSON.parse('{{ labels|safe }}')
var amounts = JSON.parse('{{ amounts|safe }}')

var ctx = document.getElementById('transactionChart').getContext('2d')
var transactionChart = new Chart(ctx, {
    type: 'line',
    data: {
        labels: labels,

```

The screenshot displays two instances of the Visual Studio Code (VS Code) interface, both showing the same project structure and file content.

**Project Structure:**

- File Explorer:** Shows the project structure under the `FMS` folder, including files like `urls.py`, `views.py`, `models.py`, `forms.py`, and various `index.html`, `register.html`, `login.html`, and `spendings.html` templates.
- Terminal:** Shows the command `[15/Apr/2024 15:32:40] "GET /login HTTP/1.1" 200 3161`.
- Status Bar:** Shows the current file is `index.html`, and other tabs include `register.html`, `login.html`, `spendings.html`, and `add-spe`.

**Content of index.html (Top Tab):**

```
var labels = JSON.parse('{{ labels|safe }}');
var amounts = JSON.parse('{{ amounts|safe }}');

var ctx = document.getElementById('transactionchart').getContext('2d');
var transactionChart = new Chart(ctx, {
    type: 'line',
    data: {
        labels: labels,
        datasets: [
            {
                label: 'Daily Spendings',
                data: amounts,
                fill: true,
                borderColor: 'rgba(0, 123, 255, 1)',
                backgroundColor: 'rgba(0, 123, 255, 0.1)',
                borderWidth: 1
            }
        ],
        options: {
            scales: {
                xAxes: [
                    {
                        type: 'time',
                        time: {
                            unit: 'day'
                        },
                        ticks: {
                            maxTicksLimit: 10
                        }
                    }
                ]
            }
        }
    }
});
```

**Content of index.html (Bottom Tab):**

```
<script>
    var labels = JSON.parse('{{ labels|safe }}');
    var amounts = JSON.parse('{{ amounts|safe }}');

    var ctx = document.getElementById('transactionchart').getContext('2d');
    var transactionChart = new Chart(ctx, {
        type: 'line',
        data: {
            labels: labels,
            datasets: [
                {
                    label: 'Daily Spendings',
                    data: amounts,
                    fill: true,
                    borderColor: 'rgba(0, 123, 255, 1)',
                    backgroundColor: 'rgba(0, 123, 255, 0.1)',
                    borderWidth: 1
                }
            ],
            options: {
                scales: {
                    xAxes: [
                        {
                            type: 'time',
                            time: {
                                unit: 'day'
                            },
                            ticks: {
                                maxTicksLimit: 10
                            }
                        }
                    ]
                }
            }
        }
    });
</script>
</body>
</html>
```

File Edit Selection View Go Run Terminal Help ← → ⌘ FMS

EXPLORER FMS

- > .vscode
- ✓ FMS
  - > \_pycache\_
  - \_\_init\_\_.py
  - asgi.py
  - settings.py
  - urls.py
  - wsgi.py
  - home
    - > \_pycache\_
    - migrations
    - \_\_init\_\_.py
    - admin.py
    - apps.py
    - forms.py
    - models.py M
    - tests.py
    - urls.py
    - views.py
  - static
    - > templates
    - add-spending... M
    - base.html
    - index.html
    - login.html
    - my-profile.html
    - register.html M
    - spendings.html
  - db.sqlite3
  - manage.py
- > OUTLINE
- > TIMELINE

templates > register.html

```

1  <!DOCTYPE html>
2  <html lang="en" class="bg-white">
3  <head>
4    <meta charset="utf-8" />
5    <meta name="viewport" content="width=device-width, initial-scale=1" />
6    <meta name="author" content="Mark Otto, Jacob Thornton, and Bootstrap contributors" />
7    <meta name="generator" content="Hugo 0.122.0" />
8    <title>FMS Registration</title>
9
10   <link rel="canonical" href="https://getbootstrap.com/docs/5.3/examples/sign-in/" />
11   <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-QWTKZyjpPEjISVWaRU9FeR" />
12   <meta name="theme-color" content="#712cf9" />
13
14   <style>
15     .backg {
16       background: rgba(255, 255, 255, 0.6);
17     }
18
19     #grad {
20       background-image: linear-gradient(to bottom right, rgba(255, 207, 207, 1), rgba(179, 239, 244, 1));
21     }
22     .bi {
23       vertical-align: -0.125em;
24       fill: currentColor;
25     }
26   </style>
27 </head>
28
29 <body class="d-flex pt-3 pb-3" id="grad">
30   <main class="m-auto w-50 backg border rounded shadow p-3">
31     <div class="d-flex flex-wrap flex-mdnowrap align-items-center pb-2 mb-3 border-bottom">
32       <h1 class="h2">FMS Registration</h1>
33     </div>
34     <form class="p-1 mb-2" method="post" action="">
35       <% csrf token %>
36       <div class="row g-2">
37         <div class="col-sm-6">
38           <label for="firstName" class="form-label">First name</label>
39           <input type="text" class="form-control" id="firstName" name="first_name" placeholder="" value="" required />
40           <div class="invalid-feedback">Valid first name is required.</div>
41         </div>
42
43         <div class="col-sm-6">
44           <label for="lastName" class="form-label">Last name</label>
45           <input type="text" class="form-control" id="lastName" name="last_name" placeholder="" value="" required />
46           <div class="invalid-feedback">Valid last name is required.</div>
47         </div>
48
49         <div class="col-12">
50           <label for="username" class="form-label">Username</label>
51           <div class="input-group has-validation">
52             <% comment %> <span class="input-group-text">&lt;span>% endcomment %</span></span>
53             <input type="text" class="form-control" id="username" name="username" placeholder="Username" required />
54             <div class="invalid-feedback">Your username is required.</div>
55           </div>
56           <div id="passwordHelpBlock" class="form-text">Required. 150 characters or fewer. Letters, digits and @./+/- only.</div>
57         </div>
58
59         <div class="col-12">
60           <label for="email" class="form-label">Email</label>
61           <input type="email" class="form-control" id="email" name="email" placeholder="you@example.com" required />
62         </div>
63       </div>
64     </form>
65   </main>
66 </body>
67
68 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3/dist/js/bootstrap.bundle.min.js" integrity="sha384-QWTKZyjpPEjISVWaRU9FeR" />
69 </html>

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

[15/Apr/2024 15:32:40] "GET /login HTTP/1.1" 200 3161

Ln 31, Col 32 Spaces:2 UTF-8 CRLF Django HTML

File Edit Selection View Go Run Terminal Help ← → ⌘ FMS

EXPLORER FMS

- > .vscode
- ✓ FMS
  - > \_pycache\_
  - \_\_init\_\_.py
  - asgi.py
  - settings.py
  - urls.py
  - wsgi.py
  - home
    - > \_pycache\_
    - migrations
    - \_\_init\_\_.py
    - admin.py
    - apps.py
    - forms.py
    - models.py M
    - tests.py
    - urls.py
    - views.py
  - static
    - > templates
    - add-spending... M
    - base.html
    - index.html
    - login.html
    - my-profile.html
    - register.html M
    - spendings.html
  - db.sqlite3
  - manage.py
- > OUTLINE
- > TIMELINE

templates > register.html

```

31 <body class="d-flex pt-3 pb-3" id="grad">
32   <main class="m-auto w-50 backg border rounded shadow p-3">
33     <div class="d-flex flex-wrap flex-mdnowrap align-items-center pb-2 mb-3 border-bottom">
34       <h1 class="h2">FMS Registration</h1>
35     </div>
36     <form class="p-1 mb-2" method="post" action="">
37       <% csrf token %>
38       <div class="row g-2">
39         <div class="col-sm-6">
40           <label for="firstName" class="form-label">First name</label>
41           <input type="text" class="form-control" id="firstName" name="first_name" placeholder="" value="" required />
42           <div class="invalid-feedback">Valid first name is required.</div>
43         </div>
44
45         <div class="col-sm-6">
46           <label for="lastName" class="form-label">Last name</label>
47           <input type="text" class="form-control" id="lastName" name="last_name" placeholder="" value="" required />
48           <div class="invalid-feedback">Valid last name is required.</div>
49         </div>
50
51         <div class="col-12">
52           <label for="username" class="form-label">Username</label>
53           <div class="input-group has-validation">
54             <% comment %> <span class="input-group-text">&lt;span>% endcomment %</span></span>
55             <input type="text" class="form-control" id="username" name="username" placeholder="Username" required />
56             <div class="invalid-feedback">Your username is required.</div>
57           </div>
58           <div id="passwordHelpBlock" class="form-text">Required. 150 characters or fewer. Letters, digits and @./+/- only.</div>
59         </div>
60
61         <div class="col-12">
62           <label for="email" class="form-label">Email</label>
63           <input type="email" class="form-control" id="email" name="email" placeholder="you@example.com" required />
64         </div>
65       </div>
66     </form>
67   </main>
68 </body>
69
70 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3/dist/js/bootstrap.bundle.min.js" integrity="sha384-QWTKZyjpPEjISVWaRU9FeR" />
71 </html>

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

[15/Apr/2024 15:32:40] "GET /login HTTP/1.1" 200 3161

Ln 31, Col 32 Spaces:2 UTF-8 CRLF Django HTML

The image shows a dual-monitor setup for web development. Both monitors display a code editor interface with a top navigation bar.

**Top Monitor:**

- Left Side (Explorer):** Shows the project structure for a Django application named "FMS". The "templates" directory contains "register.html" and "login.html".
- Right Side (Code Editor):** Displays the content of "register.html". The code uses Bootstrap 5.3 for styling. It includes fields for email and password, validation messages, and a registration button. A script tag at the bottom points to a CDN URL for Bootstrap's CSS.

**Bottom Monitor:**

- Left Side (Explorer):** Shows the same project structure as the top monitor.
- Right Side (Code Editor):** Displays the content of "login.html". The code uses Bootstrap 5.3 for styling. It includes a title, meta tags, and a form with a background gradient. A script tag at the bottom points to a CDN URL for Bootstrap's CSS.

Both monitors have a top bar with standard operating system icons (File, Edit, Selection, View, Go, Run, Terminal, Help) and a tab bar labeled "FMS". The bottom of each monitor shows a terminal window with the command "GET /login HTTP/1.1" and the response code "200 3161".

The image shows two side-by-side instances of the Visual Studio Code (VS Code) code editor. Both instances have the same file structure and are displaying different parts of a Django application named 'FMS'.

**File Structure:**

- File Explorer: Shows files like urls.py, views.py, models.py, forms.py, index.html, register.html, login.html, spending.html, and manage.py.
- Terminal: Shows the command "python" and other terminal logs.
- Output: Shows log messages from the terminal.
- Problems: Shows no errors or warnings.
- Terminal: Shows log messages from the terminal.
- Ports: Shows port information.

**Top Tab (login.html):**

```
html
body {
    height: 100%;
}

.form-signin {
    max-width: 380px;
    padding: 1rem;
}

.form-signin .form-floating:focus-within {
    z-index: 2;
}

.form-signin input[type='text'] {
    margin-bottom: -1px;
    border-top-right-radius: 0;
    border-bottom-left-radius: 0;
}

.form-signin input[type='password'] {
    margin-bottom: 10px;
    border-top-left-radius: 0;
    border-top-right-radius: 0;
}

</style>
</head>

<body class="d-flex p-4 rounded align-items-center" id="grad">
    <main class="form-signin w-100 m-auto backg rounded border shadow">
        <form class="form-signin" method="post" action="/login">
            {% csrf_token %}
            <h1 class="h3 mb-3 fw-normal text-center"><b>Login</b></h1>
            <div class="form-floating">
                <input type="text" name="username" class="form-control backg" id="floatingInput" placeholder="Username" required />
                <label for="floatingInput">Username</label>
            </div>
            <div class="form-floating">
                <input type="password" name="password" class="form-control backg" id="FloatingPassword" placeholder="Password" required />
                <label for="floatingPassword">Password</label>
            </div>
            <button class="btn btn-primary w-100 py-2" type="submit">Sign In</button>
            <a class="link-offset-2 link-offset-3-hover link-underline underline-opacity-0 link-underline-opacity-75-hover" href="/register">Don't have an account?</a>
        </form>
        <div class="d-flex mt-2">
            {% for message in messages %}
                <div class="text-bg-danger p-1"><strong>{{ message }}</strong></div>
            {% endfor %}
        </div>
    </main>
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.1/dist/js/bootstrap.bundle.min.js" integrity="sha384-YvpcrYf0tY31HB60NNkmx5s9FDVZLESaAA" crossorigin="anonymous"></script>
</body>
</html>
```

**Bottom Tab (add-spending.html):**

```
html
body {
    height: 100%;
}

.form-signin {
    max-width: 380px;
    padding: 1rem;
}

.form-signin .form-floating:focus-within {
    z-index: 2;
}

.form-signin input[type='text'] {
    margin-bottom: -1px;
    border-top-right-radius: 0;
    border-bottom-left-radius: 0;
}

.form-signin input[type='password'] {
    margin-bottom: 10px;
    border-top-left-radius: 0;
    border-top-right-radius: 0;
}

</style>
</head>

<body class="d-flex p-4 rounded align-items-center" id="grad">
    <main class="form-signin w-100 m-auto backg rounded border shadow">
        <form class="form-signin" method="post" action="/add-spending">
            {% csrf_token %}
            <h1 class="h3 mb-3 fw-normal text-center"><b>Add Spending</b></h1>
            <div class="form-floating">
                <input type="text" name="category" class="form-control backg" id="floatingInput" placeholder="Category" required />
                <label for="floatingInput">Category</label>
            </div>
            <div class="form-floating">
                <input type="text" name="amount" class="form-control backg" id="floatingAmount" placeholder="Amount" required />
                <label for="floatingAmount">Amount</label>
            </div>
            <div class="form-floating">
                <input type="text" name="date" class="form-control backg" id="floatingDate" placeholder="Date (YYYY-MM-DD)" required />
                <label for="floatingDate">Date</label>
            </div>
            <button class="btn btn-primary w-100 py-2" type="submit">Add Spending</button>
            <a class="link-offset-2 link-offset-3-hover link-underline underline-opacity-0 link-underline-opacity-75-hover" href="/register">Don't have an account?</a>
        </form>
        <div class="d-flex mt-2">
            {% for message in messages %}
                <div class="text-bg-danger p-1"><strong>{{ message }}</strong></div>
            {% endfor %}
        </div>
    </main>
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.1/dist/js/bootstrap.bundle.min.js" integrity="sha384-YvpcrYf0tY31HB60NNkmx5s9FDVZLESaAA" crossorigin="anonymous"></script>
</body>
</html>
```

The screenshot displays two instances of the Visual Studio Code (VS Code) interface, both showing the same project structure and file content.

**Project Structure:**

- File Explorer:** Shows files like `urls.py`, `models.py`, `views.py`, `forms.py`, `index.html`, `register.html`, `login.html`, and `spendings.html`.
- Terminal:** Shows the command `[15/Apr/2024 15:32:40] "GET /login HTTP/1.1" 200 3161`.
- Status Bar:** Shows "Ln 228, Col 1" and "Spaces: 2, UTF-8, CRLF, Django HTML".

**Content Area (Top Tab):**

```
templates > spendings.html
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  <meta charset="utf-8" />
5  <meta name="viewport" content="width=device-width, initial-scale=1" />
6  <title>FMS Spendings</title>
7  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-QWTKZyjpPEjISv5MaRU9Oeh" type="text/css">
8  <style>
9      .background {
10         background: rgba(255, 255, 255, 0.6);
11     }
12     #grad {
13         background-image: linear-gradient(to bottom right, rgba(255, 207, 207, 1), rgba(179, 239, 244, 1));
14     }
15     .bi {
16         display: inline-block;
17         width: 1rem;
18         height: 1rem;
19     }
20     /* *** Sidebar *** */
21     @media (min-width: 768px) {
22         .sidebar .offcanvas-lg {
23             position: -webkit-sticky;
24             position: sticky;
25             top: 48px;
26         }
27         .navbar-search {
28             display: block;
29         }
30     }
31 
```

**Content Area (Bottom Tab):**

```
templates > add-spe
1  .sidebar .nav-link {
2     font-size: 0.875rem;
3     font-weight: 500;
4 }
5     .sidebar .nav-link.active {
6     color: #2470dc;
7 }
8     .sidebar-heading {
9     font-size: 0.75rem;
10 }
11    /*
12     * Navbar
13     */
14     .navbar-brand {
15     padding-top: 0.75rem;
16     padding-bottom: 0.75rem;
17 }
18     .navbar .form-control {
19     padding: 0.75rem 1rem;
20 }
21     .bd-placeholder-img {
22     font-size: 1.125rem;
23     text-anchor: middle;
24     -webkit-user-select: none;
25     -moz-user-select: none;
26     user-select: none;
27 }
```

The image shows two side-by-side views of a Visual Studio Code (VS Code) interface, both displaying the same file: `spendings.html`. The left view shows the first half of the CSS code, and the right view shows the second half. The code is a custom CSS file for a template, utilizing media queries and various CSS properties to style elements like `.bd-placeholder-img`, `.b-example-divider`, and `.nav-scroller`.

```
67     }
68     @media (min-width: 768px) {
69       .bd-placeholder-img-lg {
70         font-size: 3.5rem;
71       }
72     }
73     .b-example-divider {
74       width: 100%;
75       height: 3rem;
76       background-color: rgba(0, 0, 0, 0.15);
77       border: solid 1px #007bff;
78       border-width: 1px 0;
79       box-shadow: inset 0 0.5em 1.5em -0.15em #007bff, inset 0 0 0.5em 0.15em #007bff;
80     }
81     .b-example-vr {
82       flex-shrink: 0;
83       width: 1.5rem;
84       height: 100vh;
85     }
86     .bi {
87       vertical-align: -0.125em;
88       fill: currentColor;
89     }
90     .nav-scroller {
91       position: relative;
92       z-index: 2;
93       height: 2.75rem;
94       overflow-y: hidden;
95     }
96     .nav-scroller .nav {
97       display: flex;
98       flex-wrap: nowrap;
99       padding-bottom: 1rem;
100      margin-top: -1px;
101      overflow-x: auto;
102      text-align: center;
103      white-space: nowrap;
104      -webkit-overflow-scrolling: touch;
105    }
106    .btn-bd-primary {
107      --bd-violet-bg: #732cf9;
108      --bd-violet-rgb: 117.520718, 44.062154, 249.437846;
109      --bs-btn-font-weight: 600;
110      --bs-btn-color: var(--bs-white);
111      --bs-btn-bg: var(--bd-violet-bg);
112      --bs-btn-border-color: var(--bd-violet-bg);
113      --bs-btn-hover-color: var(--bs-white);
114      --bs-btn-hover-bg: #6528e0;
115      --bs-btn-hover-border-color: #6528e0;
116      --bs-btn-focus-shadow-rgb: var(--bd-violet-rgb);
117      --bs-btn-active-color: var(--bs-btn-hover-color);
118      --bs-btn-active-bg: #5a23c8;
119      --bs-btn-active-border-color: #5a23c8;
120    }
121    .bd-mode-toggle {
122      z-index: 1500;
123    }
124  
```

File Edit Selection View Go Run Terminal Help ← → 🔍 FMS

EXPLORER FMS

- > .vscode
- ✓ FMS
  - \_pycache\_
  - \_\_init\_\_.py
  - asgi.py
  - settings.py
  - urls.py
  - wsgi.py
  - home
  - \_pycache\_
  - migrations
  - \_\_init\_\_.py
  - admin.py
  - apps.py
  - forms.py
  - models.py M
  - tests.py
  - urls.py
  - views.py
  - static
  - templates
  - add-spending... M
  - base.html
  - index.html
  - login.html
  - my-profile.html
  - register.html M
  - spendings.html
- db.sqlite3 M
- manage.py

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

[15/Apr/2024 15:32:40] "GET /login HTTP/1.1" 200 3161

In 228, Col 1 Spaces 2 UTF-8 CRLF Django HTML

---

File Edit Selection View Go Run Terminal Help ← → 🔍 FMS

EXPLORER FMS

- > .vscode
- ✓ FMS
  - \_pycache\_
  - \_\_init\_\_.py
  - asgi.py
  - settings.py
  - urls.py
  - wsgi.py
  - home
  - \_pycache\_
  - migrations
  - \_\_init\_\_.py
  - admin.py
  - apps.py
  - forms.py
  - models.py M
  - tests.py
  - urls.py
  - views.py
  - static
  - templates
  - add-spending... M
  - base.html
  - index.html
  - login.html
  - my-profile.html
  - register.html M
  - spendings.html
- db.sqlite3 M
- manage.py

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

[15/Apr/2024 15:32:40] "GET /login HTTP/1.1" 200 3161

In 228, Col 1 Spaces 2 UTF-8 CRLF Django HTML

{# instance.fields.R &lt;td>

**Bottom Instance:**

```
</tr>
        {# endfor #}
      </tbody>
    </table>
  </div>
  </main>
</div>
</div>
<script src="https://cdn.jsdelivr.net/npm/chart.js@4.3.2/dist/chart.umd.js" integrity="sha384-e17P5r3L1XLSH8JdD1I5Yn/njoSxfbrkCTnJrzxt+ENPSMV</script>
<!-- <script src="dashboard.js"></script> -->
</body>
</html>
```

**Status Bar:** Shows "In 228, Col 1" and "python" icon.

The screenshot displays two instances of the Visual Studio Code (VS Code) interface, both showing the FMS (Food Management System) project structure and a specific template file.

**Project Structure:**

- File Explorer:** Shows files like `views.py`, `models.py`, `forms.py`, `index.html`, `register.html`, `login.html`, `spendings.html`, `add-spending.html`, `base.html`, `index.html`, `login.html`, `my-profile.html`, `register.html`, `spendings.html`, `db.sqlite3`, and `manage.py`.
- Terminal:** Shows the command `[15/Apr/2024 15:32:40] "GET /login HTTP/1.1" 200 3161`.
- Status Bar:** Shows "Ln 262, Col 32" and "Spaces: 2, UTF-8, CRLF, Django HTML".

**Code Editor (Top Instance):**

```
templates > add-spending.html
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  <meta charset="utf-8" />
5  <meta name="viewport" content="width=device-width, initial-scale=1" />
6  <title>FMS Add Transaction</title>
7  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-QWTKZyjpPEjJSv5MaRU9OeB" type="text/css">
8  <style>
9      .background {
10         background: rgba(255, 255, 255, 0.6);
11     }
12     #grad {
13         background-image: linear-gradient(to bottom right, rgba(255, 207, 207, 1), rgba(179, 239, 244, 1));
14     }
15     .bi {
16         display: inline-block;
17         width: 1rem;
18         height: 1rem;
19     }
20
21     /* sidebar */
22
23     @media (min-width: 768px) {
24         .sidebar .offcanvas-lg {
25             position: -webkit-sticky;
26             position: sticky;
27             top: 48px;
28         }
29
30         .navbar-search {
31             display: block;
32         }
33     }
34
35     .sidebar .nav-link {
36         font-size: 0.875rem;
37         font-weight: 500;
38     }
39
40     .sidebar .nav-link.active {
41         color: #2470dc;
42     }
43
44     .sidebar-heading {
45         font-size: 0.75rem;
46     }
47
48     /*Navbar*/
49
50     .navbar-brand {
51         padding-top: 0.75rem;
52         padding-bottom: 0.75rem;
53     }
54
55     .navbar .form-control {
56         padding: 0.75rem 1rem;
57     }
58
59     .bd-placeholder-img {
60         font-size: 1.125rem;
61         text-anchor: middle;
62         -webkit-user-select: none;
63         -moz-user-select: none;
64         user-select: none;
65     }
66
```

**Code Editor (Bottom Instance):**

```
templates > add-spending.html
34
35     .sidebar .nav-link {
36         font-size: 0.875rem;
37         font-weight: 500;
38     }
39
40     .sidebar .nav-link.active {
41         color: #2470dc;
42     }
43
44     .sidebar-heading {
45         font-size: 0.75rem;
46     }
47
48     /*Navbar*/
49
50     .navbar-brand {
51         padding-top: 0.75rem;
52         padding-bottom: 0.75rem;
53     }
54
55     .navbar .form-control {
56         padding: 0.75rem 1rem;
57     }
58
59     .bd-placeholder-img {
60         font-size: 1.125rem;
61         text-anchor: middle;
62         -webkit-user-select: none;
63         -moz-user-select: none;
64         user-select: none;
65     }
66
```

The screenshot displays two instances of the Visual Studio Code (VS Code) interface, both showing the FMS project structure and the contents of the `templates/add-spending.html` file.

**Top Instance:**

- File Explorer:** Shows the project structure with files like `views.py`, `models.py`, `forms.py`, `index.html`, `register.html`, `login.html`, `spendings.html`, `add-spending.html`, `base.html`, `index.html`, `my-profile.html`, `register.html`, and `spendings.html`.
- Code Editor:** Displays the `add-spending.html` template file with the following CSS code:

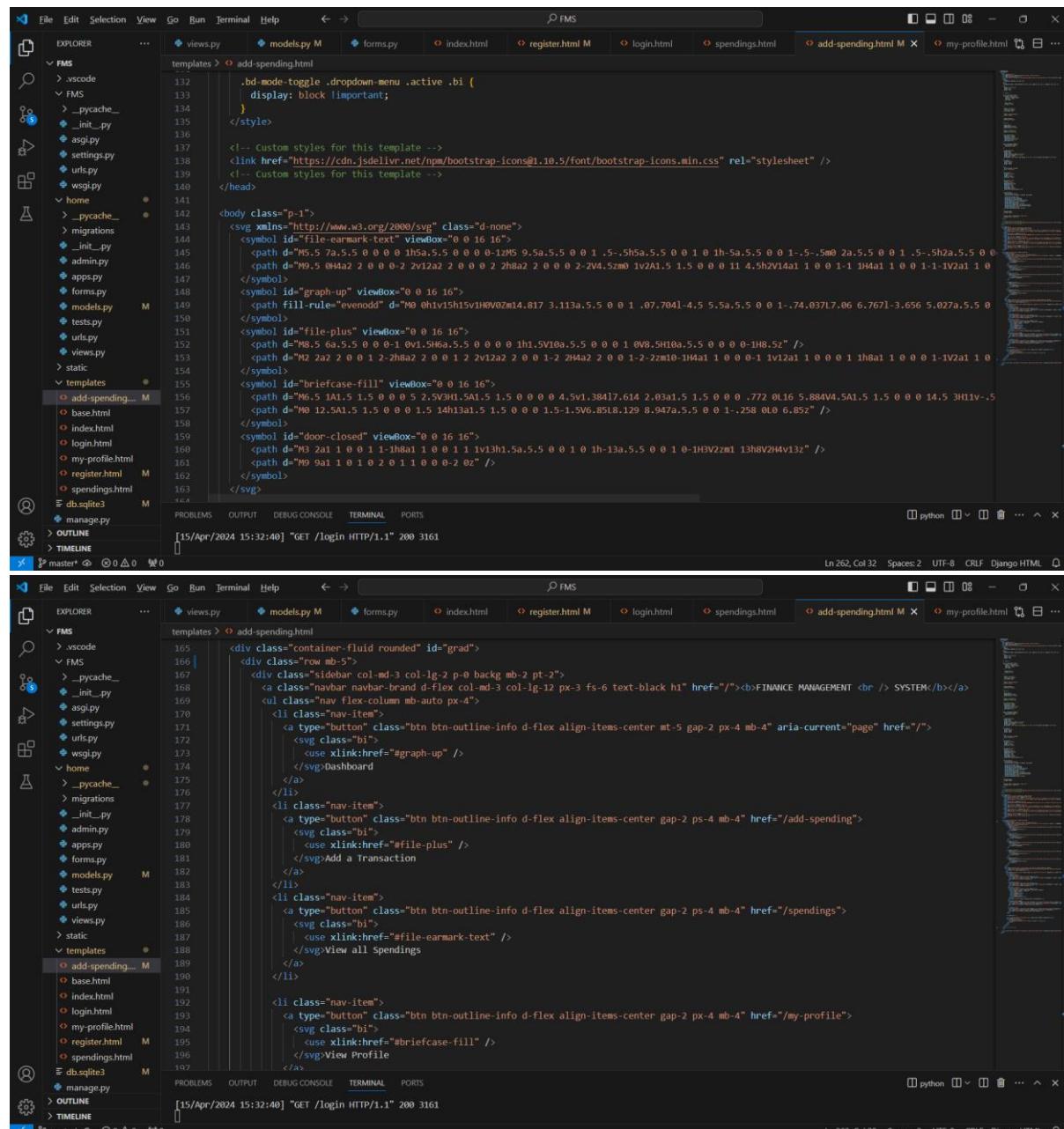
```
@media (min-width: 768px) { .bd-placeholder-img-lg { font-size: 3.5rem; } } .b-example-divider { width: 100%; height: 3rem; background-color: rgba(0, 0, 0, 0.1); border: solid rgba(0, 0, 0, 0.15); border-width: 1px 0; box-shadow: inset 0 0.5em 1.5em rgba(0, 0, 0, 0.1), inset 0 0.125em 0.5em rgba(0, 0, 0, 0.15); } .b-example-vr { flex-shrink: 0; width: 1.5rem; height: 100vh; } .bi { vertical-align: -0.125em; fill: currentColor; } .nav-scroller { position: relative; z-index: 2; height: 2.75rem; overflow-y: hidden; } 
```

**Bottom Instance:**

- File Explorer:** Shows the same project structure.
- Code Editor:** Displays the `add-spending.html` template file with more CSS code:

```
.nav-scroller .nav { display: flex; flex-wrap: nowrap; padding-bottom: 1rem; margin-top: -1px; overflow-x: auto; text-align: center; white-space: nowrap; -webkit-overflow-scrolling: touch; } .btn-bd-primary { --bd-violet-bg: #712cf9; --bd-violet-rgb: 111.520718, 44.062154, 249.437846; --bs-btn-font-weight: 600; --bs-btn-color: var(--bs-white); --bs-btn-bg: var(--bd-violet-bg); --bs-btn-border-color: var(--bd-violet-bg); --bs-btn-hover-color: var(--bs-white); --bs-btn-hover-bg: #6528e0; --bs-btn-hover-border-color: #6528e0; --bs-btn-focus-shadow-rgb: var(--bd-violet-rgb); --bs-btn-active-color: var(--bs-btn-hover-color); --bs-btn-active-bg: #5a23c8; --bs-btn-active-border-color: #5a23c8; } .bd-mode-toggle { z-index: 1500; } 
```

Both instances show the status bar at the bottom with "Ln 262, Col 32" and "Spaces: 2, UTF-8, CRLF, Django HTML".



The screenshot displays two instances of a web browser, likely Microsoft Edge, running Django development servers. Both instances show the same codebase for a 'Finance Management System' (FMS). The top instance shows the 'add-spending.html' template, which includes custom Bootstrap icons and a detailed SVG definition for a briefcase icon. The bottom instance shows the 'my-profile.html' template. Both browsers have their toolbars visible, showing tabs for views.py, models.py, forms.py, index.html, register.html, login.html, spendings.html, add-spending.html, and my-profile.html. The bottom browser also features a 'python' icon in its toolbar, indicating it's running a Python application. The status bar at the bottom of both browser windows shows the URL as '15/APR/2024 15:32:40' and the status as 'GET /login HTTP/1.1 200 3161'. The overall environment suggests a developer is comparing or testing different parts of the application simultaneously.

File Edit Selection View Go Run Terminal Help ← → 🔍 FMS

EXPLORER FMS

- > .vscode
- > \_pycache\_
- \_\_init\_\_.py
- asgi.py
- settings.py
- urls.py
- wsgi.py
- > home
- > \_pycache\_
- migrations
- \_\_init\_\_.py
- admin.py
- apps.py
- forms.py
- models.py M
- tests.py
- urls.py
- views.py
- > static
- > templates
- add-spending M
- base.html
- index.html
- login.html
- my-profile.html
- register.html M
- spendings.html

templates > add-spending.html

```

197      </div>
198      </li>
199      <li class="nav-item">
200        <a type="button" class="btn btn-outline-info d-flex align-items-center gap-2 px-4 mb-4" href="/logout">
201          <svg class="bi">
202            <use xlink:href="#door-closed" />
203          </svg> Sign out
204        </a>
205      </li>
206    </ul>
207  </div>
208  <main class="col-md-9 m-auto col-lg-6 px-md-4 mt-3 border rounded shadow mb-5 backg">
209    <div class="d-flex justify-content-between flex-wrap flex-mdnowrap align-items-center pt-3 pb-2 mb-3 border-bottom">
210      <h1 class="h2"><b>Add Transaction</b></h1>
211    </div>
212    <form class="form-control mb-3" style="background: rgba(255, 255, 255, 0); method="post" action="/add-spending">
213      [% csrf_token %]
214      <div class="row g-3">
215        <div class="col-sm-6">
216          <label for="PaidTo" class="form-label">Paid to</label>
217          <input type="text" class="form-control" name="Paidto" id="Paidto" placeholder="" value="" required />
218          <div class="invalid-feedback">Invalid</div>
219        </div>
220        <div class="col-sm-6">
221          <label for="amount" class="form-label">Amount</label>
222          <input type="number" class="form-control" name="Amount" id="amount" placeholder="" value="" required />
223          <div class="invalid-feedback">Invalid</div>
224        </div>
225        <div class="col-md-12">
226          <label for="Reason" class="form-label">Reason</label>
227          <select class="form-select" name="Reason" id="Reason" required>
228            <option value="">Choose...</option>
229          </select>
230          <div class="invalid-feedback">Please select a valid Reason.</div>
231        </div>
232        <div class="col-12" hidden>
233          <label for="FMSuser" class="form-label">Username <span class="text-body-secondary">(Fixed)</span></label>
234          <input type="text" readonly class="form-control" name="FMSuser" id="FMSuser" placeholder="" value="{{ user.username }}"/>
235          <div class="invalid-feedback">Please enter valid remarks.</div>
236        </div>
237        <div class="col-12">
238          <label for="Remarks" class="form-label">Remarks <span class="text-body-secondary">(Optional)</span></label>
239          <input type="text" class="form-control" name="Remarks" id="Remarks" placeholder="" />
240          <div class="invalid-feedback">Please enter valid remarks.</div>
241        </div>
242        <div class="my-4" />
243        <button class="w-100 btn btn-primary btn-lg" type="submit">Submit Transaction</button>
244        <div class="d-flex mt-2">
245          [% for message in messages %]
246            <div class="text-bg-success p-1"><strong>{{ message }}</strong></div>
247          [% endfor %]
248        </div>
249      </form>
250    </div>
251  </div>
252  <hr class="my-4" />
253  <div class="d-flex justify-content-between align-items-center">
254    <div>
255      <small>Copyright © 2024. All Rights Reserved. FMS</small>
256    </div>
257    <div>
258      <small>Developed by [REDACTED]</small>
259    </div>
260  </div>
261  <div class="text-center">
262    <small>FMS is a Django application built for managing financial transactions. It allows users to add, view, and edit spending entries. The application uses a clean, modern design with a focus on user-friendly navigation and form validation. The code is well-organized and follows best practices for Django development. The database is managed using SQLite, and the application is designed to be easily deployable to production environments. The user interface is responsive and works well on both desktop and mobile devices. The application includes features such as user authentication, role-based access control, and detailed reporting capabilities. The code is modular and easy to maintain, making it a great starting point for building similar financial management systems. The application is currently in its early stages of development and is still being actively worked on. The team behind the application is committed to providing a high-quality product that meets the needs of its users. If you have any questions or feedback, please don't hesitate to reach out. We're always looking for ways to improve and grow our application. Thank you for using FMS!</small>
263  </div>
264</div>
265</body>
266</html>

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

[15/Apr/2024 15:32:40] "GET /login HTTP/1.1" 200 3161

Ln 262, Col 32 Spaces: 2 UTF-8 CRLF Django HTML

File Edit Selection View Go Run Terminal Help ← → 🔍 FMS

EXPLORER FMS

- > .vscode
- > \_pycache\_
- \_\_init\_\_.py
- asgi.py
- settings.py
- urls.py
- wsgi.py
- > home
- > \_pycache\_
- migrations
- \_\_init\_\_.py
- admin.py
- apps.py
- forms.py
- models.py M
- tests.py
- urls.py
- views.py
- > static
- > templates
- add-spending M
- base.html
- index.html
- login.html
- my-profile.html
- register.html M
- spendings.html

templates > add-spending.html

```

229      <option value="">Choose...</option>
230      <option>Food</option>
231      <option>Clothing</option>
232      <option>Travel</option>
233      <option>Medical</option>
234      <option>Miscellaneous</option>
235    </select>
236    <div class="invalid-feedback">Please select a valid Reason.</div>
237  </div>
238  <div class="col-12" hidden>
239    <label for="FMSuser" class="form-label">Username <span class="text-body-secondary">(Fixed)</span></label>
240    <input type="text" readonly class="form-control" name="FMSuser" id="FMSuser" placeholder="" value="{{ user.username }}"/>
241    <div class="invalid-feedback">Please enter valid remarks.</div>
242  </div>
243  <div class="col-12">
244    <label for="Remarks" class="form-label">Remarks <span class="text-body-secondary">(Optional)</span></label>
245    <input type="text" class="form-control" name="Remarks" id="Remarks" placeholder="" />
246    <div class="invalid-feedback">Please enter valid remarks.</div>
247  </div>
248  <div class="my-4" />
249  <button class="w-100 btn btn-primary btn-lg" type="submit">Submit Transaction</button>
250  <div class="d-flex mt-2">
251    [% for message in messages %]
252      <div class="text-bg-success p-1"><strong>{{ message }}</strong></div>
253    [% endfor %]
254  </div>
255</div>
256</body>
257</html>

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

[15/Apr/2024 15:32:40] "GET /login HTTP/1.1" 200 3161

Ln 262, Col 32 Spaces: 2 UTF-8 CRLF Django HTML

The screenshot displays two tabs in a code editor:

- add-spending.html**: A Django template for a spending form. It includes a message placeholder for invalid remarks, a submit button, and a message block for success messages. A script for a chart is included at the bottom.
- my-profile.html**: A CSS file containing Bootstrap imports and a sidebar style with a linear gradient background.

Both tabs show syntax highlighting for Python and CSS respectively. The status bar at the bottom indicates the current file is my-profile.html.

The image shows two side-by-side instances of the Visual Studio Code (VS Code) code editor. Both instances have the same file structure and code content, indicating they are likely screenshots of the same workspace at different times or from different perspectives.

**File Structure:**

- Project Name: FMS
- Root Folders: .vscode, FMS, home, static, templates
- Files under FMS:
  - \_\_init\_\_.py
  - asgi.py
  - settings.py
  - urls.py
  - wsgi.py
- Files under home:
  - \_\_init\_\_.py
  - migrations
  - models.py
  - tests.py
  - urls.py
  - views.py
- Static files: base.html, index.html, login.html, my-profile.html, register.html, spendings.html
- Database: db.sqlite3
- Management Scripts: manage.py

**Code Editor (Top Tab):**

```
.sidebar .nav-link {  
    font-size: 0.875rem;  
    font-weight: 500;  
}  
  
.sidebar .nav-link.active {  
    color: #2470d2;  
}  
  
.sidebar-heading {  
    font-size: 0.75rem;  
}  
  
/* * Navbar */  
  
.navbar-brand {  
    padding-top: 0.75rem;  
    padding-bottom: 0.75rem;  
}  
  
.navbar .form-control {  
    padding: 0.75rem 1rem;  
}  
  
.bd-placeholder-img {  
    font-size: 1.125rem;  
    text-anchor: middle;  
    -webkit-user-select: none;  
    -moz-user-select: none;  
    user-select: none;  
}
```

**Code Editor (Bottom Tab):**

```
@media (min-width: 768px) {  
    .bd-placeholder-img-lg {  
        font-size: 3.5rem;  
    }  
  
.b-example-divider {  
    width: 100%;  
    height: 3rem;  
    background-color: rgba(0, 0, 0, 0.1);  
    border: solid rgba(0, 0, 0, 0.15);  
    border-width: 1px 0;  
    box-shadow: inset 0 0.5em 1.5em rgba(0, 0, 0, 0.1), inset 0 0.125em 0.5em rgba(0, 0, 0, 0.15);  
}  
  
.b-example-vr {  
    flex-shrink: 0;  
    width: 1.5rem;  
    height: 100vh;  
}  
  
.bi {  
    vertical-align: -0.125em;  
    fill: currentColor;  
}  
  
.nav-scroller {  
    position: relative;  
    z-index: 2;  
    height: 2.75rem;  
    overflow-y: hidden;  
}
```

The image shows two side-by-side screenshots of a code editor, likely Visual Studio Code, displaying Python code for a Django application named 'FMS'.

**Top Screenshot:**

- Explorer View:** Shows the project structure with files like `views.py`, `models.py`, `forms.py`, `index.html`, `register.html`, `login.html`, `spendings.html`, `add-spending.html`, and `my-profile.html`.
- Code Editor:** The file `templates/my-profile.html` is open, containing CSS-like styles for buttons and a mode toggle button.
- Bottom Status Bar:** Shows the command `[15/Apr/2024 15:32:40] "GET /login HTTP/1.1" 200 3161`.

**Bottom Screenshot:**

- Explorer View:** Similar to the top one, showing the project structure.
- Code Editor:** The file `templates/my-profile.html` is open, showing more complex CSS and HTML for a dropdown menu and various icons.
- Bottom Status Bar:** Shows the command `[15/Apr/2024 15:32:40] "GET /login HTTP/1.1" 200 3161`.

**Screenshot 1: my-profile.html (Initial State)**

```

<div class="container-fluid rounded" id="grad">
  <div class="row">
    <div class="sidebar col-md-3 col-lg-2 p-0 backg mb-2 pt-2">
      <a class="nav-item nav-link">
        <img alt="FMS Logo" data-bbox="118 118 135 135" style="vertical-align: middle; margin-right: 10px;"/>
        FINANCE MANAGEMENT <br/> SYSTEM
      </a>
    </div>
    <div class="nav flex-column mb-auto px-4">
      <li class="nav-item">
        <a type="button" class="btn btn-outline-info d-flex align-items-center gap-2 px-4 mb-4" href="#" aria-current="page">
          <img alt="Graph Up" data-bbox="118 148 135 165" style="vertical-align: middle; margin-right: 10px;"/>
          Dashboard
        </a>
      </li>
      <li class="nav-item">
        <a type="button" class="btn btn-outline-info d-flex align-items-center gap-2 ps-4 mb-4" href="/add-spending">
          <img alt="File Plus" data-bbox="118 178 135 195" style="vertical-align: middle; margin-right: 10px;"/>
          Add a Transaction
        </a>
      </li>
      <li class="nav-item">
        <a type="button" class="btn btn-outline-info d-flex align-items-center gap-2 ps-4 mb-4" href="/spendings">
          <img alt="File Earmark Text" data-bbox="118 208 135 225" style="vertical-align: middle; margin-right: 10px;"/>
          View all Spendings
        </a>
      </li>
      <li class="nav-item">
        <a type="button" class="btn btn-outline-info d-flex align-items-center gap-2 px-4 mb-4" href="/my-profile">
          <img alt="Briefcase Fill" data-bbox="118 238 135 255" style="vertical-align: middle; margin-right: 10px;"/>
          View Profile
        </a>
      </li>
    </div>
  </div>

```

**Screenshot 2: my-profile.html (After Logout)**

```

<div class="container-fluid rounded" id="grad">
  <div class="row">
    <div class="sidebar col-md-3 col-lg-2 p-0 backg mb-2 pt-2">
      <a class="nav-item nav-link">
        <img alt="FMS Logo" data-bbox="118 118 135 135" style="vertical-align: middle; margin-right: 10px;"/>
        FINANCE MANAGEMENT <br/> SYSTEM
      </a>
    </div>
    <div class="nav flex-column mb-auto px-4">
      <li class="nav-item">
        <a type="button" class="btn btn-outline-info d-flex align-items-center gap-2 px-4 mb-4" href="/logout">
          <img alt="Door Closed" data-bbox="118 148 135 165" style="vertical-align: middle; margin-right: 10px;"/>
          Sign out
        </a>
      </li>
    </div>
  </div>

```



# SQLITE database and tables:

DB Browser for SQLite - D:\django\_project\FMS\db.sqlite3

File Edit View Tools Help

New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database Close Database

Create Table Create Index Modify Table Delete Table Print

Name Type Schema

**Tables (14)**

- auth\_group
- auth\_group\_permissions
- auth\_permission
- auth\_user
- auth\_user\_groups
- auth\_user\_user\_permissions
- django\_admin\_log
- django\_content\_type
- django\_migrations
- django\_session
- home\_addspending
- home\_custommodelname
- home\_registeruser
- sqlite\_sequence

**Indices (16)**

- auth\_group\_permissions\_group\_id\_b120cbf9
- auth\_group\_permissions\_group\_id\_c0c325b0\_uniq
- auth\_group\_permissions\_permission\_id\_0c325b0\_uniq
- auth\_group\_permissions\_permission\_id\_84c5c2e1
- auth\_group\_permissions\_permission\_id\_2f476e4b
- auth\_permission\_content\_type\_id\_97559544
- auth\_user\_groups\_group\_id\_6a12ed8b
- auth\_user\_groups\_user\_id\_group\_id\_943500c0\_uniq
- auth\_user\_permissions\_permission\_id\_1fb657c2
- auth\_user\_permissions\_user\_id\_a95ead1b
- auth\_user\_permissions\_user\_id\_permission\_id\_14a6bb632\_uniq
- django\_admin\_log\_content\_type\_id\_cfcbe8e0
- django\_admin\_log\_user\_id\_564ebaf0
- django\_content\_type\_app\_label\_model\_76bd3d3b\_uniq

```

CREATE TABLE "auth_group" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "name" varchar(150) NOT NULL UNIQUE)
CREATE TABLE "auth_group_permissions" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "group_id" integer NOT NULL REFERENCES "auth_group" ON DELETE CASCADE, "permission_id" integer NOT NULL REFERENCES "auth_permission" ON DELETE CASCADE)
CREATE TABLE "auth_permission" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "content_type_id" integer NOT NULL REFERENCES "django_content_type" ON DELETE CASCADE, "codename" varchar(100) NOT NULL, "name" varchar(50) NOT NULL)
CREATE TABLE "auth_user" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "password" varchar(128) NOT NULL, "last_login" datetime NOT NULL, "is_superuser" integer NOT NULL, "username" varchar(150) NOT NULL, "first_name" varchar(150) NOT NULL, "last_name" varchar(150) NOT NULL, "email" varchar(254) NOT NULL, "is_staff" integer NOT NULL, "is_active" integer NOT NULL, "date_joined" datetime NOT NULL)
CREATE TABLE "auth_user_groups" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "user_id" integer NOT NULL REFERENCES "auth_user" ON DELETE CASCADE, "group_id" integer NOT NULL REFERENCES "auth_group" ON DELETE CASCADE)
CREATE TABLE "auth_user_user_permissions" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "user_id" integer NOT NULL REFERENCES "auth_user" ON DELETE CASCADE, "permission_id" integer NOT NULL REFERENCES "auth_permission" ON DELETE CASCADE)
CREATE TABLE "django_admin_log" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "object_id" text NULL, "object_repr" varchar(200) NOT NULL, "action_flag" smallint NOT NULL, "change_message" longtext NOT NULL, "content_type_id" integer NOT NULL, "user_id" integer NOT NULL, "log_time" datetime NOT NULL)
CREATE TABLE "django_content_type" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "app_label" varchar(100) NOT NULL, "model" varchar(100) NOT NULL)
CREATE TABLE "django_migrations" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "app" varchar(255) NOT NULL, "name" varchar(255) NOT NULL, "applied" datetime NOT NULL)
CREATE TABLE "django_session" ("session_key" varchar(40) NOT NULL PRIMARY KEY, "session_data" text NOT NULL, "expire_date" datetime NOT NULL)
CREATE TABLE "home_addspending" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "PaidTo" varchar(122) NOT NULL, "Amount" decimal NOT NULL)
CREATE TABLE "home_custommodelname" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "user_id" integer NOT NULL REFERENCES "auth_user" ON DELETE CASCADE)
CREATE TABLE "home_registeruser" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "FirstName" varchar(122) NOT NULL, "LastName" varchar(122) NOT NULL, "Email" varchar(150) NOT NULL, "Mobile" varchar(15) NOT NULL, "Address" varchar(255) NOT NULL, "City" varchar(100) NOT NULL, "State" varchar(100) NOT NULL, "Country" varchar(100) NOT NULL, "Pincode" integer NOT NULL, "ProfilePic" varchar(255) NOT NULL, "CreatedDate" datetime NOT NULL, "LastModifiedDate" datetime NOT NULL)
CREATE TABLE "sqlite_sequence" ("name" text, "seq" integer)

```

DB Browser for SQLite - D:\django\_project\FMS\db.sqlite3

File Edit View Tools Help

New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Pragmas Execute SQL

Table: home\_addspending

Filter in any column

	PaidTo	Amount	Date	Reason	Remarks	FMSUser
1	3 CampusMart	40	2024-04-02	Food	2 Itchi juice	nilay
2	4 bluedoveco	100	2024-04-02	Food	noodles	nilay
3	5 nikhil	200	2024-04-03	Medical	disprin	nilay
4	6 mom	1000	2024-04-03	Travel	ticket	nilay
5	7 valbhav	100	2024-04-03	Food	2 Itchi juice	nilay
6	9 bbindstant	40	2024-04-04	Food	coffee	nilay
7	10 swiggy	347	2024-04-05	Food	ordered from swiggy	nilay
8	11 amazon	378	2024-04-05	Clothing	noodles	nilay
9	12 arnav	345	2024-04-06	Miscellaneous	book	nilay
10	13 rameshwara	80	2024-04-06	Travel	auto	nilay
11	14 chef's touch	45	2024-04-06	Food	cold coffee	nilay
12	16 CampusMart	20	2024-04-10	Food	Juice	nilay
13	17 CampusMart	40	2024-04-11	Food	2 chips packet@	nilay
14	18 om	25	2024-04-11	Food	icecream	nilay
15	21 valbhav	78	2024-04-12	Miscellaneous	ball	nilay
16	22 Canara Integrated Services	40	2024-04-12	Food	2 chocolates	nilay
17	23 Swiggy	382	2024-04-13	Food	BurgerKing	nilay
18	24 Netmeds	20	2024-04-13	Medical	disprin	nilay
19	25 CampusMart	70	2024-04-14	Food	Cup noodles and 2 Itlchee juice	nilay
20	26 Canara Integrated Services	20	2024-04-14	Food	snickers	nilay
21	27 prakhar	10	2024-04-14	Miscellaneous	pen	nilay
22	28 Canara Integrated Services	20	2024-04-15	Food	Coffee	nilay

Go to: 1