

# EXPENSE EASE

*A Mini Project Report Submitted*

*In partial fulfillment of the requirement for the award of the degree of*

***Bachelor of Technology  
In  
Computer Science and Engineering -Artificial Intelligence  
and Machine Learning***

**By**

**P.S.V.S.S. Madhurima      21N31A66D4**

**Puppala Nithin              21N31A66F1**

**Romala Abilash             22N35A6616**

Under the Guidance of

**Mrs. G. Deepthi**

**Assistant Professor**

**Department of Computational Intelligence**

**MRCET**



**DEPARTMENT OF COMPUTATIONAL INTELLIGENCE  
MALLA REDDY COLLEGE OF ENGINEERING AND TECHNOLOGY**

**(Affiliated to JNTU, Hyderabad)**

**ACCREDITED by AICTE-NBA**

**Maisammaguda, Dhulapally post, Secunderabad-500014.**

**2024-2025**

## **DECLARATION**

I hereby declare that the project entitled “**Expense Ease**” submitted to **Malla Reddy College of Engineering and Technology**, affiliated to Jawaharlal Nehru Technological University Hyderabad (JNTUH) for the award of the degree of **Bachelor of Technology in Computer Science and Engineering- Artificial Intelligence and Machine Learning** is a result of original research work done by me.

It is further declared that the project report or any part thereof has not been previously submitted to any University or Institute for the award of degree or diploma.

**P.S.V.S.S. Madhurima** (21N31A66D4)

**Puppala Nithin** (21N31A66F1)

**Romala Abilash** (22N35A6616)



# **MALLA REDDY COLLEGE OF ENGINEERING & TECHNOLOGY**

**(AUTONOMOUS INSTITUTION - UGC, GOVT. OF INDIA)**

**Affiliated to JNTUH; Approved by AICTE, NBA-Tier 1 & NAAC with A-GRADE | ISO 9001:2015**

## **CERTIFICATE**

This is to certify that this is the bonafide record of the project titled **“Expense Ease”** submitted by **P.S.V.S.S. Madhurima(21N31A66D4), Puppala Nithin (21N31A66F1) and Romala Abilash(22N31A6616)** of B. Tech in the partial fulfillment of the requirements for the degree of **Bachelor of Technology in Computer Science and Engineering- Artificial Intelligence and Machine Learning**, Dept. of CI during the year 2024-2025. The results embodied in this project report have not been submitted to any other university or institute for the award of any degree or diploma.

**G. DEEPTHI**

Assistant Professor

**INTERNAL GUIDE**

**Dr. D. SUJATHA**

Professor

**HEAD OF THE DEPARTMENT**

**EXTERNAL EXAMINER**

## ACKNOWLEDGEMENT

We feel honored and privileged to place our warm salutation to our college Malla Reddy College of Engineering and technology (UGC-Autonomous), our Director ***Dr. VSK Reddy*** who gave us the opportunity to have experience in engineering and profound technical knowledge.

We are indebted to our Principal ***Dr. S. Srinivasa Rao*** for providing us with facilities to do our project and his constant encouragement and moral support which motivated us to move forward with the project.

We would like to express our gratitude to our Head of the Department ***Dr. D. Sujatha***, Professor for encouraging us in every aspect of our system development and helping us realize our full potential.

We would like to thank our mini project coordinator ***Mr. D. Chandra Sekhar Reddy***, Associate Professor as well as our internal guide ***Mrs. G. Deepthi***, Assistant Professor for their structured guidance and never-ending encouragement. We are extremely grateful for valuable suggestions and unflinching co-operation throughout mini project work.

We would also like to thank all supporting staff of department of CI and all other departments who have been helpful directly or indirectly in making our mini project a success.

We would like to thank our parents and friends who have helped us with their valuable suggestions and support has been very helpful in various phases of the completion of the mini project.

By

**P.S.V.S.S. Madhurima (21N31A66D4)**

**Puppala Nithin (21N31A66F1)**

**Romala Abilash (22N35A6616)**

## ABSTRACT

**Expense Ease** is a powerful financial management solution that enables users to categorize and track expenses with ease, manage unplanned accidental expenses, and personalize profiles through image uploads. With comprehensive CRUD operations, users can log and filter transactions by custom categories like groceries or entertainment, gaining clear insights into spending patterns and budgeting. The platform includes a dedicated accidental expenses feature, allowing users to set aside an accidental budget for emergencies, log unexpected costs, and view visual comparisons of budgeted versus actual spending, with remaining balance indicators for easy monitoring. Additionally, profile image uploads add a personalized touch and multi-tenancy ensures that users only access their own data while administrators can oversee all activities, providing a well-rounded, user-friendly financial tracking experience.

# TABLE OF CONTENTS

| <b>S. No.</b> | <b>Topic</b>                      | <b>Page No.</b> |
|---------------|-----------------------------------|-----------------|
| CHAPTER 1:    | INTRODUCTION                      |                 |
|               | 1.1: Purpose                      | 1               |
|               | 1.2: Background of project        | 1               |
|               | 1.3: Scope of project             | 2               |
|               | 1.4: Project features             | 2               |
|               | 1.5: Existing System              | 3               |
|               | 1.6: Drawbacks of Existing System | 4               |
|               | 1.7: Proposed System              | 4               |
| CHAPTER 2:    | SYSTEM REQUIREMENTS               |                 |
|               | 2.1: Hardware Requirements        | 5               |
|               | 2.2: Software Requirements        | 5               |
| CHAPTER 3:    | SYSTEM DESIGN                     |                 |
|               | 3.1: System Architecture          | 6               |
|               | 3.2: UML Diagrams                 |                 |
|               | 3.2.1: Use Case Diagram           | 7               |
|               | 3.2.2: Class Diagram              | 8               |
|               | 3.2.3: Sequence Diagram           | 9               |
|               | 3.2.4: Activity Diagram           | 10              |
| CHAPTER 4:    | IMPLEMENTATION                    |                 |
|               | 4.1: Source Code                  | 11              |
|               | 4.2: Output Screens               | 21              |
|               | 4.3: Testing                      | 27              |
| CHAPTER 5:    | CONCLUSION & FUTURE SCOPE         | 28              |
| CHAPTER 6:    | BIBLIOGRAPHY                      | 29              |

# 1. INTRODUCTION

Expense Ease is a comprehensive financial tracking and management solution with CRUD operations for income, expenses, and categories along with currency formatting. Users can manage their financial records, generate detailed monthly reports, and filter data by categories. The system allows users to access only their data while administrators oversee all information. Additionally, it includes robust budget planning tools, advanced categorization features, and receipt uploading, providing a user-friendly platform for effective financial tracking and reporting. It aims to help users accurately record, categorize, and manage their income and expenses, generate detailed financial reports, and gain valuable insights for better financial planning and decision-making.

## 1.1 Purpose:

The purpose of the Expense Ease project is to provide users with an organized, efficient way to manage and track their personal finances. It aims to simplify budgeting by enabling users to categorize expenses, set and monitor budgets—including for unexpected costs—log income and expenses, and generate detailed reports that offer valuable insights into spending patterns. With a user-friendly interface, the project enhances financial awareness, helps users make informed budgeting decisions, and supports responsible financial planning. The platform's multi-tenancy feature ensures data security and privacy, making it ideal for both individual users who seek streamlined financial management and administrators who oversee user activities. Overall, Expense Ease empowers users to maintain control over their finances, reduce overspending, and achieve financial stability.

## 1.2 Background of project:

The background of an Expense Ease project stems from the growing complexity of managing personal and business finances in today's fast-paced world. With more people turning to technology for budgeting and financial planning, the lack of accessible, user-friendly platforms specifically tailored to manage various expenses—such as regular, accidental, and one-off expenditures—became

apparent. Existing solutions were often complex or expensive, with limited flexibility for individual customization. Recognizing this gap, Expense Ease was developed to offer a versatile, comprehensive financial management solution that can cater to a range of users. The platform combines key financial tracking functionalities, like categorization, real-time budget monitoring, and detailed reporting, with a modern user interface and accessibility features, ensuring ease of use. The project addresses the essential need for financial transparency and literacy, making it especially relevant as people seek to improve financial habits, reduce debt, and make informed spending decisions in an increasingly cashless economy.

### **1.3 Scope of project:**

The scope of the Expense Ease project encompasses the development, implementation, and optimization of a comprehensive expense management platform that caters to both individuals and businesses. This project will include the creation of a user-friendly application with features such as expense tracking, real-time budget monitoring, income and expense categorization, and detailed reporting tools.

Additionally, the project will provide tools for creating budgets from setting financial goals, and offering notifications and reminders to ensure users stay within their spending limits. Advanced features such as data analytics, AI-based expense forecasting, and multi-user access for businesses are also part of the scope. The project will cover robust security measures to protect sensitive financial information and ensure compliance with data protection regulations.

### **1.4 Project Features:**

- **Categorization:**

Expenses and income can be categorized (e.g., food, transport, salary) to help users see where their money is going, making budgeting and financial planning easier. This categorization provides insights into spending habits and highlights areas for potential savings, giving users a clearer picture of their financial health.



- **Image Upload:**

Users can upload their profile pictures, making the application feel more personal and allowing users to visually identify their profiles. This feature adds a personalized touch, enhancing user engagement, and trust in the platform.

- **Accidental Expenses:**

This feature allows users to log and track unplanned or unforeseen costs separately from their regular, planned budget, providing a clear view of unexpected financial impacts, which can aid in better financial decision-making and future budgeting.

- **Budget Planning:**

Allow users to set monthly or yearly budgets for different categories, helping them track spending against their financial goals and receive alerts when approaching or exceeding budget limits.

- **Income and Expense Reports:**

Generate detailed, customizable reports that provide insights into spending trends, income sources, and budget performance. The reports will be generated as per date-wise, monthly-wise, and yearly-wise, allowing users to analyze their financial activities over different timeframes for better financial planning.

## **1.5 Existing System:**

- **User Management:** Enables multi-tenancy with individual user access to their own data while allowing administrators to oversee and manage all users' information.
- **Income Tracking:** Users can log their sources of income to keep track of their earnings alongside their expenses.
- **Simple Dashboards:** Overview dashboards that provide a snapshot of financial health, showing key metrics like total income, total expenses, and remaining budget.

## 1.6 Drawbacks of existing system:

- **Limited User Isolation:** Many existing systems lack robust multi-tenancy, meaning individual users don't have fully isolated environments. This can lead to security concerns where users may inadvertently access data belonging to others.
- **Limited Income Categorization:** In some systems, users can log income, but there is limited support for categorizing income into different sources (e.g., salary, freelance, investments). This can restrict detailed reporting and analysis of earnings.
- **Information Overload:** Dashboards in many systems can become overly complex, cluttered with excessive graphs, charts, and data points, leading to information overload for users. This makes it difficult for users to quickly assess their financial health at a glance.
- **Expense-Centric Design:** Many systems focus heavily on expense tracking but fail to provide comprehensive income tracking. This can make it difficult for users to get a clear picture of their overall financial health, as they can only track what they spend but not what they earn.
- **Weak Data Security:** Some systems do not provide robust encryption or multi-factor authentication (MFA), making them vulnerable to data breaches or unauthorized access.

## 1.7 Proposed System:

- **User Authentication:** Secure registration and login system for users, ensuring that financial data is protected.
- **Financial Tracking:** Comprehensive tracking of income and expenses with detailed categorization (e.g., food, transport, salary) to provide insights into spending habits.
- **CRUD Operations:** Users can create, read, update, and delete financial records, making it easy to manage their finances.
- **Categorization:** Users can categorize expenses and income, enabling a clearer understanding of their financial activities and helping identify areas for savings.
- **Profile Management:** Users can upload profile pictures, enhancing personalization and allowing for easy identification within the application.
- **Accidental Expenses Logging:** A dedicated feature for logging unplanned or unforeseen costs separately from the regular budget, helping users maintain better control over unexpected expenses.

- **Detailed Reporting:** Generation of customizable reports that offer insights into spending trends, income sources, and budget performance, with options for date-wise, monthly-wise, and yearly-wise analysis.
- **User-Friendly Interface:** Intuitive and easy-to-navigate design that enhances user experience and encourages effective financial management.

## 2. SYSTEM REQUIREMENTS

### 2.1 Hardware Requirements:

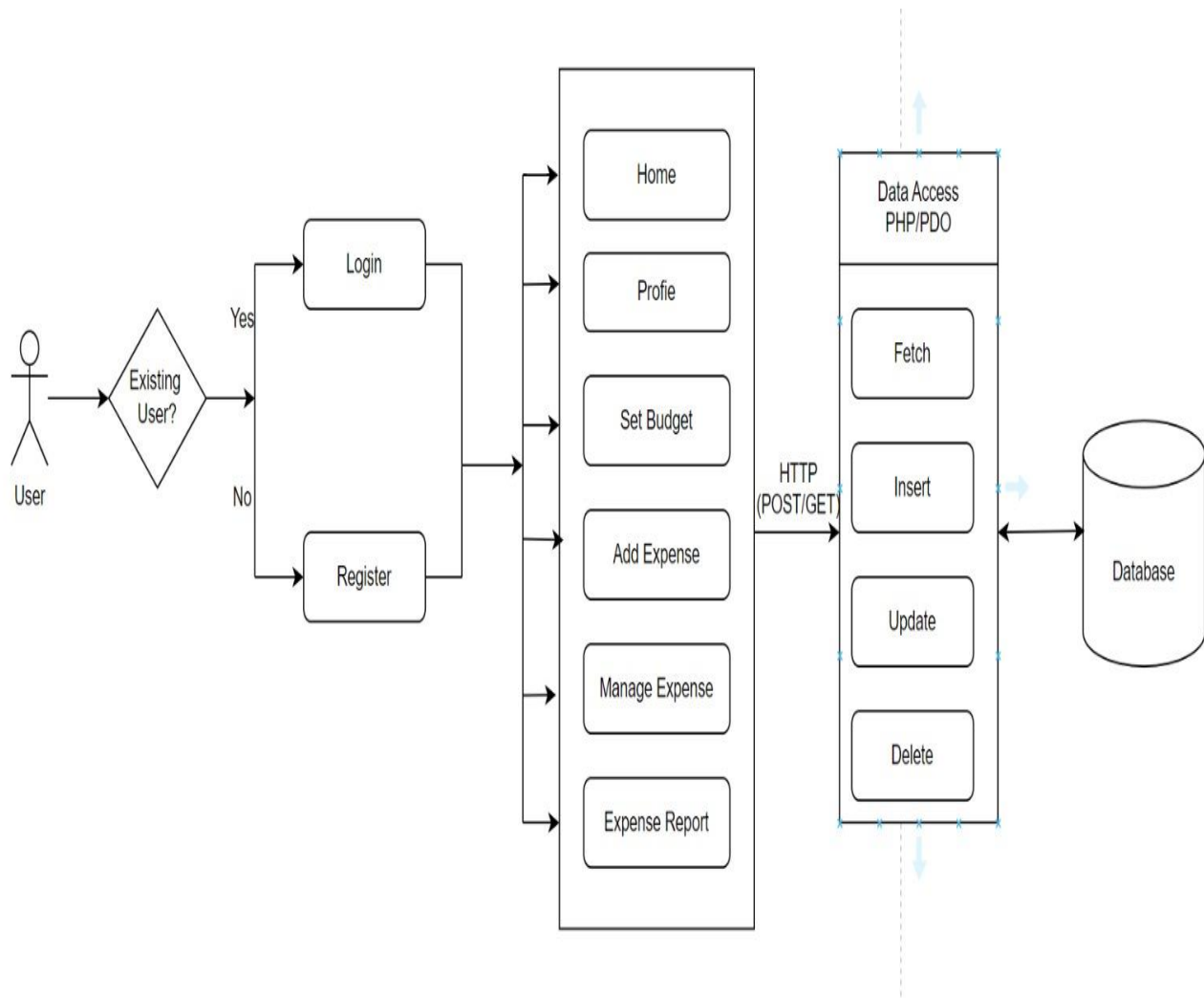
- Processor Type: Intel Core i3 and above
- Speed: 3.40GHZ
- RAM: 4GB RAM and above
- Hard disk: 30GB

### 2.2 Software requirements:

- Operating System: Windows 8/8.1/10/11 (64-bit)
- Coding language: PHP, HTML, CSS, JavaScript
- Visual Studio Code
- XAMPP Server 8.0.3
- MySQL
- XAMPP Control Panel
- PHPMyAdmin

### 3. SYSTEM DESIGN

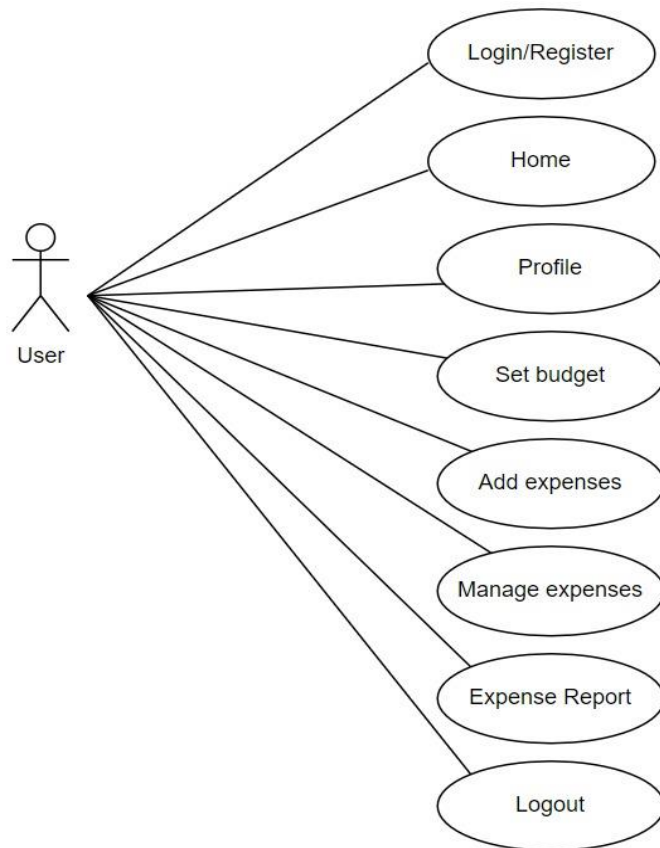
#### 3.1 System Architecture:



**Fig 3.1 Expense Ease - System Architecture**

## 3.2 UML Diagrams:

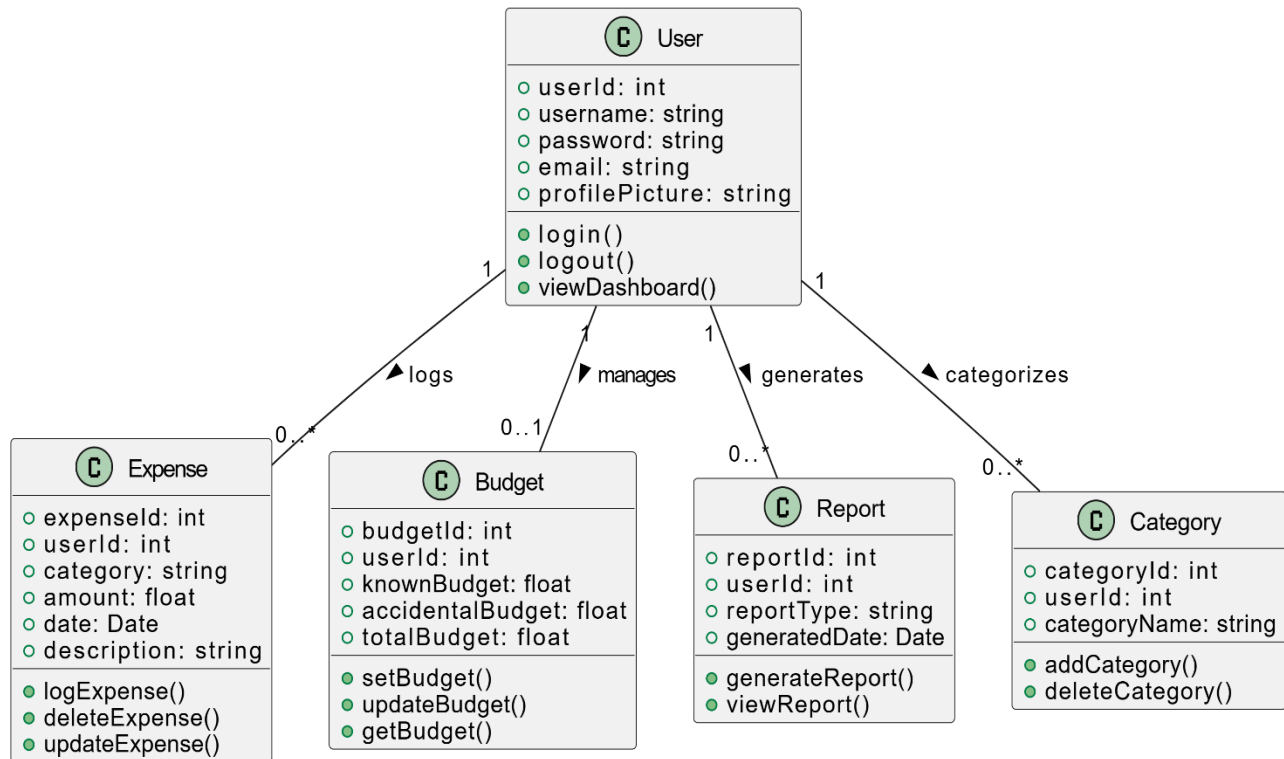
### 3.2.1 Use case diagram



**Fig 3.2.1 Expense Ease – Use Case Diagram**

Use Case during requirement elicitation and analysis to represent the functionality of the system. Use case describes a function by the system that yields a visible result for an actor. The identification of actors and use cases result in the definitions of the boundary of the system i.e., differentiating the tasks accomplished by the system and the tasks accomplished by its environment.

### 3.2.2 Class Diagram



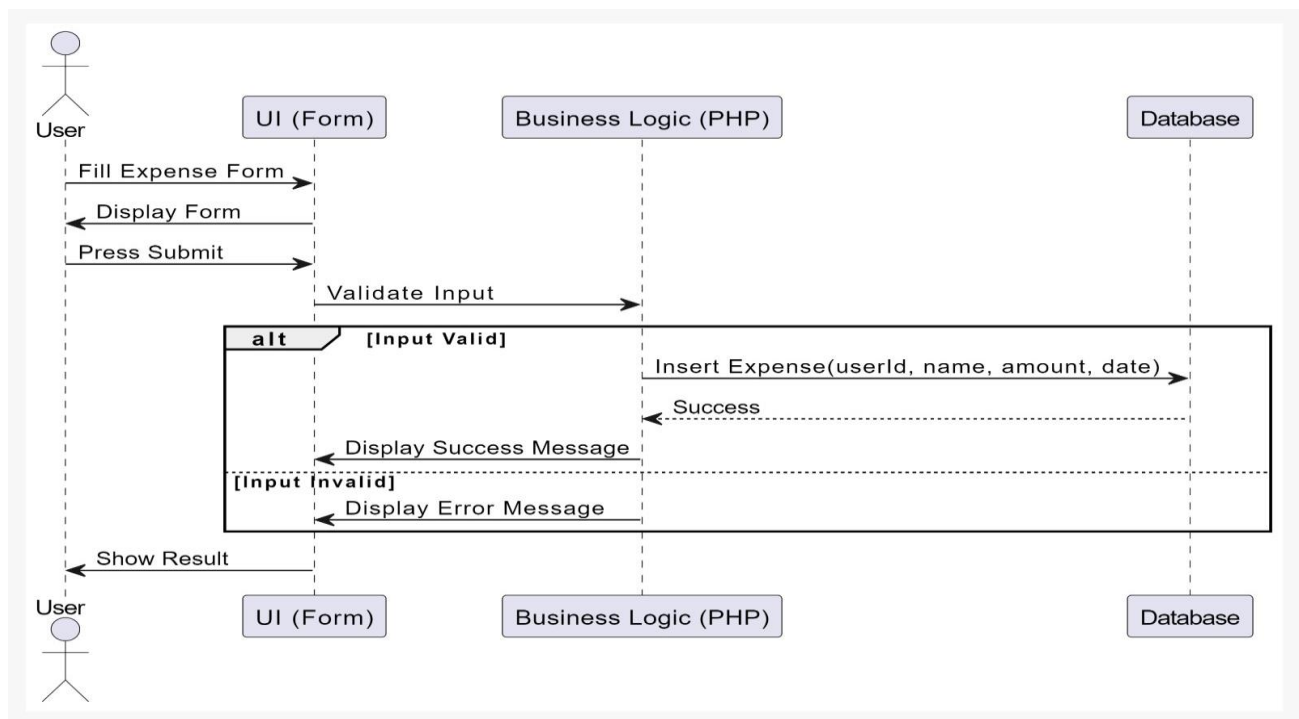
**Fig 3.2.2 Expense Ease – Class Diagram**

Class diagrams model class structure and contents using design elements such as classes, packages and objects. Class diagram describe the different perspective when designing a system-conceptual, specification and implementation. Classes are composed of three things: name, attributes, and operations. Class diagram also display relationships such as containment, inheritance, association etc. The association relationship is most common relationship in a class diagram. The association shows the relationship between instances of classes.

### 3.2.3 Sequence Diagram

Sequence diagram displays the time sequence of the objects participating in the interaction. This consists of the vertical dimension (time) and horizontal dimension (different objects).

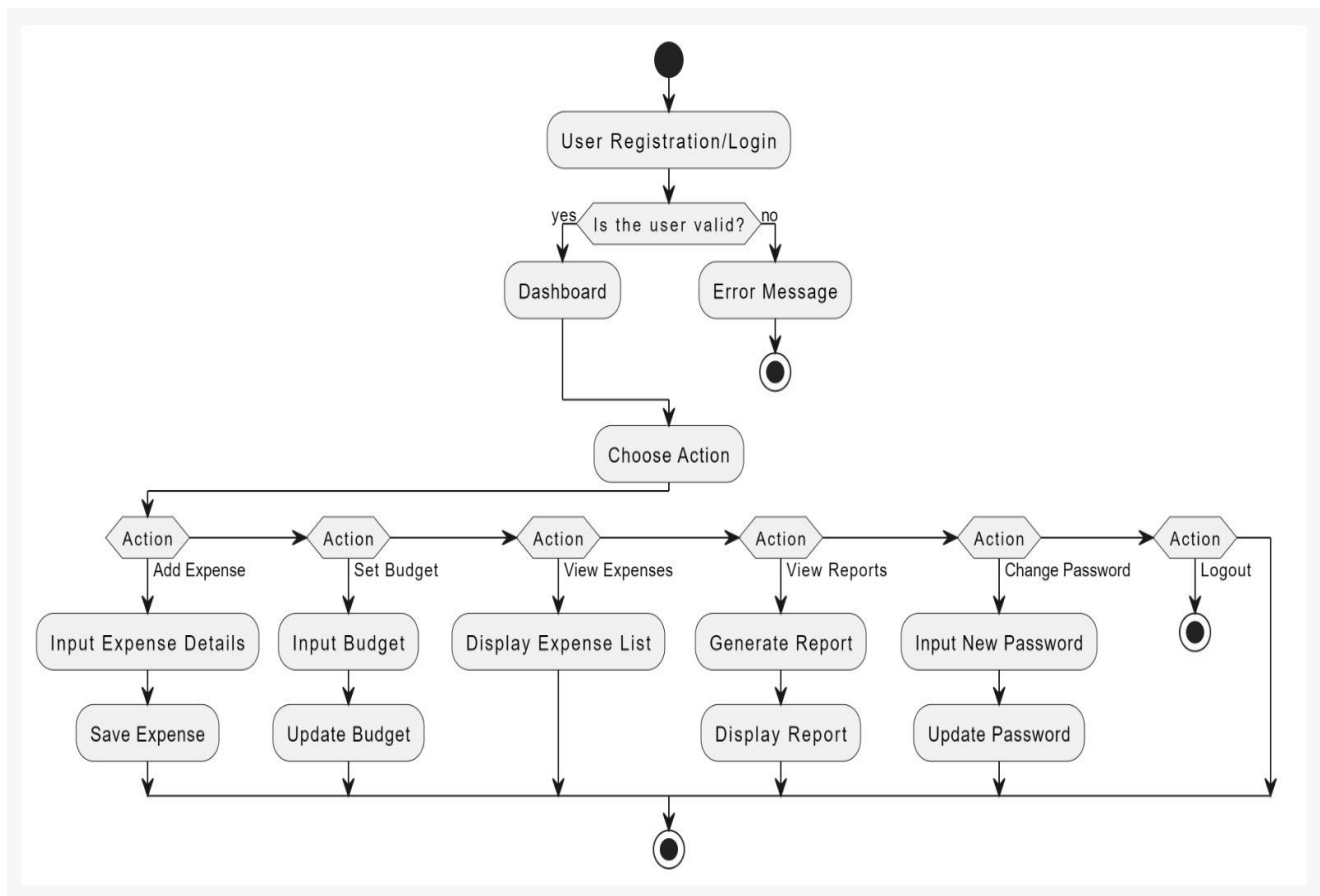
Objects: An object can be thought of as an entity that exists at a specified time and has a definite value, as well as a holder of identity. A sequence diagram depicts item interactions in chronological order. It illustrates the scenario's objects and classes, as well as the sequence of messages sent between them in order to carry out the scenario's functionality. In the Logical View of the system under development, sequence diagrams are often related with use case realizations. Event diagrams and event scenarios are other names for sequence diagrams. A sequence diagram depicts multiple processes or things that exist simultaneously as parallel vertical lines (lifelines), and the messages passed between them as horizontal arrows, in the order in which they occur. This enables for the graphical specification of simple runtime scenarios.



**Fig 3.2.3 Expense Ease – Sequence Diagram**

### 3.2.4 Activity Diagram

The process flows in the system are captured in the activity diagram. Similar to a state diagram, an activity diagram also consists of activities, actions, transitions, initial and final states, and guard conditions.



**Fig 3.2.4 Expense Ease – Activity Diagram**



## 4. IMPLEMENTATION

### 4.1 Code:

```
<?php
    include_once "init.php";

    // User login check
    if (isset($_SESSION['UserId'])) {
        header('Location: templates/3-Dashboard.php');
    }

    // Validate credentials and log the user in
    if (isset($_POST['login']) && !empty($_POST)) {
        $password = $_POST['password'];
        $username = $_POST['username'];

        if(!empty($username) || !empty($password)) {
            $username = $getFromU->checkInput($username);
            $password = $getFromU->checkInput($password);
            if($getFromU->login($username, $password) === false) {
                $error = "The username or password is incorrect";
            }
        }
    }
?>

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

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="icon" href="static/images/wallet.png" sizes="16x16" type="image/png">
    <link rel="stylesheet" href="static/css/index.css">
    <link href="https://fonts.googleapis.com/css2?family=Source+Sans+Pro:wght@600&display=swap"
rel="stylesheet">
    <link href="https://fonts.googleapis.com/css2?family=Raleway:wght@300&display=swap"
rel="stylesheet">
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.9.0/css/all.css">
```

```
<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/font-awesome/4.7.0/css/font-awesome.min.css"
```

```
  integrity="sha384-
wvfXppqpZDVQKG6TAh5PVIgOfQNHSoD2xbE+QkPxCAFINEEvoEH3SI0sibVcOQVnN"
  crossorigin="anonymous">
```

```
<title>Expense Management</title>
</head>
```

```
<body>
```

```
  <div class="container">
```

```
    <div class="mob-hidden">
```

```
      <h1>Expense Ease</h1>
```

```
    </div>
```

```
    <div class="top-heading">
```

```
      <h1>Expense Ease</h1>
```

```
    </div>
```

```
    <form action="index.php" method="post" onsubmit = "return validate()" id="form1">
```

```
      <div class="group">
```

```
        <div class="form-controller">
```

```
          <i class="fa fa-user-plus u3" aria-hidden="true"></i>
```

```
          <input type="text" name="username" placeholder="Username" id="user1" required>
```

```
          <br>
```

```
          <small></small>
```

```
        </div>
```

```
        <div class="form-controller">
```

```
          <i class="fa fa-key u4" aria-hidden="true"></i>
```

```
          <input type="password" name="password" placeholder="Password" id="pass1"
autocomplete="on" required>
```

```
          <br>
```

```
          <small></small>
```

```
        </div>
```

```
      </div>
```

```
      <button type="submit" class="sign-in" name="login">Log In</button>
```

```
    <br>
```

```

<?php
    if (isset($error)) {
        $font = "Source Sans Pro";
        echo '<div style="color: red;font-family:'. $font.';">'. $error.'</div>';
    }
?>

    <div class="new-account">
        <span style="color: rgba(0, 0, 0, 0.54); font-weight: bolder; font-family: 'Source Sans
Pro';">Don't have an account?</span>
        <a href="templates/2-sign-up.php" style="text-decoration: none;"><span style="color:
rgba(5, 0, 255, 0.81); font-weight: bolder; font-family: 'Source Sans Pro';">Sign Up Now</span></a>
    </div>

</form>

    <div class="img-container">
        
    </div>
</div>

<script src="static/js/index.js"></script>
</body>

</html>

<?php
    include_once "../init.php";

    //User login check
    if ($getFromU->loggedIn() === false) {
        header('Location: ../index.php');
    }

    include_once 'skeleton.php';

    if (isset($_SESSION['swal']))
    {
        echo $_SESSION['swal'];
        unset($_SESSION['swal']);
    }

    // Budget validity checker

```

```

$budget_validity = $getFromB->budget_validity_checker($_SESSION['UserId']);
if($budget_validity == false)
{
    $getFromB->del_budget_record($_SESSION['UserId']);
}

// Today's Expenses
$today_expense = $getFromE->Expenses($_SESSION['UserId'],0);
if($today_expense == NULL)
{
    $today_expense = "No Expenses Logged Today";
}
else
{
    $today_expense = "₹".$today_expense;
}

// Yesterday's Expenses
$Yesterday_expense = $getFromE->Yesterday_expenses($_SESSION['UserId']);
if($Yesterday_expense == NULL)
{
    $Yesterday_expense = "No Expenses Logged Yesterday";
}
else
{
    $Yesterday_expense = "₹ ".$Yesterday_expense;
}

// Last 7 Days' Expenses
$week_expense = $getFromE->Expenses($_SESSION['UserId'],6);
if($week_expense == NULL)
{
    $week_expense = "No Expenses Logged This Week";
}
else
{
    $week_expense = "₹".$week_expense;
}

// Last 30 Days' Expenses
$monthly_expense = $getFromE->Expenses($_SESSION['UserId'],29);
if($monthly_expense == NULL)
{
    $monthly_expense = "No Expenses This Month";
}

```

```

else
{
    $monthly_expense = "₹ ".$monthly_expense;
}

// Total Expenses
$total_expenses = $getFromE->totalexp($_SESSION['UserId']);
if($total_expenses == NULL)
{
    $total_expenses = "No Expenses Logged Yet";
}
else
{
    $total_expenses = "₹ ".$total_expenses;
}

// Budget Left for the month
$budget_left = $getFromB->checkbudget($_SESSION['UserId']);
if($budget_left == NULL)
{
    $budget_left = "Not Set Yet";
}
else
{
    $currmonexp = $getFromE->Current_month_expenses($_SESSION['UserId']);
    if($currmonexp==NULL)
    {
        $currmonexp = 0;
    }
    $budget_left = $budget_left - $currmonexp;
    $budget_left = "₹ ".$budget_left;
}

?>
<div class="wrapper">
    <div class="row">
        <div class="col-4 col-m-4 col-sm-4">
            <div class="card">
                <div class="counter bg-danger" style="color:white;">
                    <p><i class="fas fa-tasks"></i></p>
                    <h3>
                        Today's Expenses
                    </h3>
                    <p style="font-size: 1.2em;">
                        <?php echo $today_expense ?>

```

```

        </p>
    </div>
</div>
</div>
<div class="col-4 col-m-4 col-sm-4">
    <div class="card">
        <div class="counter bg-primary" style="color:white;">
            <p><i class="fas fa-undo-alt"></i></p>
            <h3>
                Yesterday's Expenses
            </h3>
            <p style="font-size: 1.2em;">
                <?php echo $Yesterday_expense ?>
            </p>
        </div>
    </div>
</div>
<div class="col-4 col-m-4 col-sm-4">
    <div class="card">
        <div class="counter bg-warning" style="color:white;">
            <p><i class="fas fa-calendar-week"></i></p>
            <h3>
                Last 7 day's Expenses
            </h3>
            <p style="font-size: 1.2em;">
                <?php echo $week_expense ?>
            </p>
        </div>
    </div>
</div>
<div class="col-4 col-m-4 col-sm-4">
    <div class="card">
        <div class="counter bg-vio" style="color:white;">
            <p><i class="fas fa-calendar"></i></p>
            <h3>
                Last 30 day's Expenses
            </h3>
            <p style="font-size: 1.2em;">
                <?php echo $monthly_expense ?>
            </p>
        </div>
    </div>
</div>
<div class="col-4 col-m-4 col-sm-4">
    <div class="card">
        <div class="counter bg-success" style="color:white;">

```

```

        <p><i class="fas fa-rupee-sign"></i></p>
        <h3>
            Monthly Budget Left
        </h3>
        <p style="font-size: 1.2em;">
            <?php echo $budget_left ?>
        </p>
    </div>
</div>
<div class="col-4 col-m-4 col-sm-4">
    <div class="card">
        <div class="counter bg-yell" style="color:white;">
            <p><i class="fas fa-file-invoice-dollar" aria-hidden="true"></i></p>
            <h3>
                Total Expenses
            </h3>
            <p style="font-size: 1.2em;">
                <?php echo $total_expenses ?>
            </p>
        </div>
    </div>
</div>
</div>
<div>
    <?php
    class Budget extends Base {
        function __construct($pdo) {
            $this->pdo = $pdo;
        }

        // To check validity of the set budget
        public function budget_validity_checker($UserId){
            $stmt = $this->pdo->prepare("SELECT EXTRACT(MONTH FROM RDATE) AS mon
            FROM budget WHERE UserId = :user");
            $stmt->bindParam(":user", $UserId, PDO::PARAM_INT);
            $stmt->execute();
            $r = $stmt->fetch(PDO::FETCH_OBJ);
            if($r == NULL)
            {
                return true;
            }
            else
            {
                $val1 = $r->mon;
            }
        }
    }

```

```

$stmt2 = $this->pdo->prepare("SELECT EXTRACT(MONTH FROM
CURRENT_TIMESTAMP()) AS current");
$stmt2->execute();
$z = $stmt2->fetch(PDO::FETCH_OBJ);
$val2 = $z->current;

if($val1 === $val2)
{
    return true;
}
else
{
    return false;
}
}

// To set the budget
public function setbudget($UserId, $budget) {
    $stmt = $this->pdo->prepare("INSERT INTO budget(UserId, Budget) VALUES(:user ,
:amount)");
    $stmt->bindParam(":user", $UserId, PDO::PARAM_INT);
    $stmt->bindParam(":amount", $budget, PDO::PARAM_INT);
    $stmt->execute();
}

// To check the current budget
public function checkbudget($UserId) {
    $stmt = $this->pdo->prepare("SELECT Budget AS currentbudget FROM budget WHERE
UserId=:user");
    $stmt->bindParam(":user", $UserId, PDO::PARAM_INT);
    $stmt->execute();
    $rows = $stmt->fetch(PDO::FETCH_OBJ);
    if($rows == NULL)
    {
        return NULL;
    }
    else
    {
        return $rows->currentbudget;
    }
}

// To update current budget
public function updatebudget($UserId, $budget) {
    $stmt = $this->pdo->prepare("UPDATE budget SET Budget = :amount, RDATE =
CURRENT_TIMESTAMP() WHERE UserId = :user");

```



```

$stmt->bindParam(":user", $UserId, PDO::PARAM_INT);
$stmt->bindParam(":amount", $budget, PDO::PARAM_INT);
$stmt->execute();
}

// To delete the monthly budget record (Once the month changes)
public function del_budget_record($UserId){
    $stmt = $this->pdo->prepare("DELETE FROM budget WHERE UserId = :user");
    $stmt->bindParam(":user", $UserId, PDO::PARAM_INT);
    $stmt->execute();
}

public function setAccidentalBudget($UserId, $amount) {
    try {
        $stmt = $this->pdo->prepare("UPDATE budget SET AccidentalBudget = :amount
WHERE UserId = :userId");
        $stmt->bindParam(':amount', $amount);
        $stmt->bindParam(':userId', $UserId);
        return $stmt->execute(); // Returns true on success
    } catch (PDOException $e) {
        return $e->getMessage(); // Return error message on failure
    }
}

public function setAccidentalExpenses($UserId, $accidentalExpense) {
    $stmt = $this->pdo->prepare("INSERT INTO budget (UserId, AccidentalExpenses)
VALUES (:user, :expense)");
    $stmt->bindParam(':user', $UserId, PDO::PARAM_INT);
    $stmt->bindParam(':expense', $accidentalExpense, PDO::PARAM_STR);
    $stmt->execute();
}

// To set expenses
public function setExpenses($UserId, $item, $cost) {
    try {
        $stmt = $this->pdo->prepare("INSERT INTO expense (UserId, Item, Cost, Date,
category) VALUES (:user, :item, :cost, NOW(), :category)");

        // Prepare your variables
        $category = "Accidental Expense"; // Provide a category for the expense

        $stmt->bindParam(':user', $UserId);
        $stmt->bindParam(':item', $item); // This could be something like "Accidental Expense"
        $stmt->bindParam(':cost', $cost); // Use the correct amount for the expense
        $stmt->bindParam(':category', $category);

        $stmt->execute();
    }
}

```

```

        return true;
    } catch (PDOException $e) {
        return $e->getMessage(); // Return the error message for debugging
    }
}

public function getAccidentalBudget($user_id) {
    $stmt = $this->pdo->prepare("SELECT AccidentalBudget FROM budget WHERE UserId = :user_id");
    $stmt->bindParam(':user_id', $user_id, PDO::PARAM_INT);
    $stmt->execute();
    return $stmt->fetchColumn(); // Returns the accidental budget amount
}

public function getAccidentalExpenses($UserId) {
    $stmt = $this->pdo->prepare("SELECT SUM(Cost) FROM expense WHERE UserId = :user
AND category = 'Accidental Expense'");
    $stmt->bindParam(':user', $UserId);
    $stmt->execute();
    return $stmt->fetchColumn(); // Returns the total accidental expenses
}
}
?>
<?php
    session_start();
    include_once 'connection.php';
    include_once 'base.php';
    include_once 'user.php';
    include_once 'expense.php';
    include_once 'budget.php';

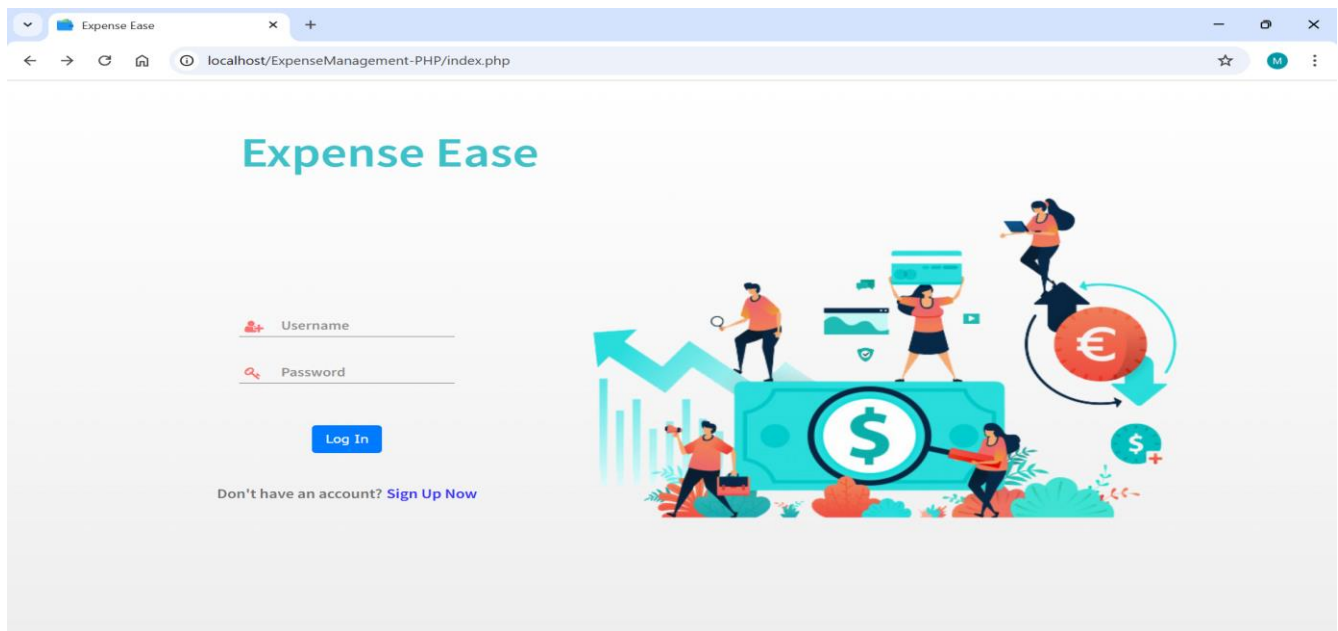
    global $pdo;

    $getFromU = new User($pdo);
    $getFromB = new Budget($pdo);
    $getFromE = new Expense($pdo);

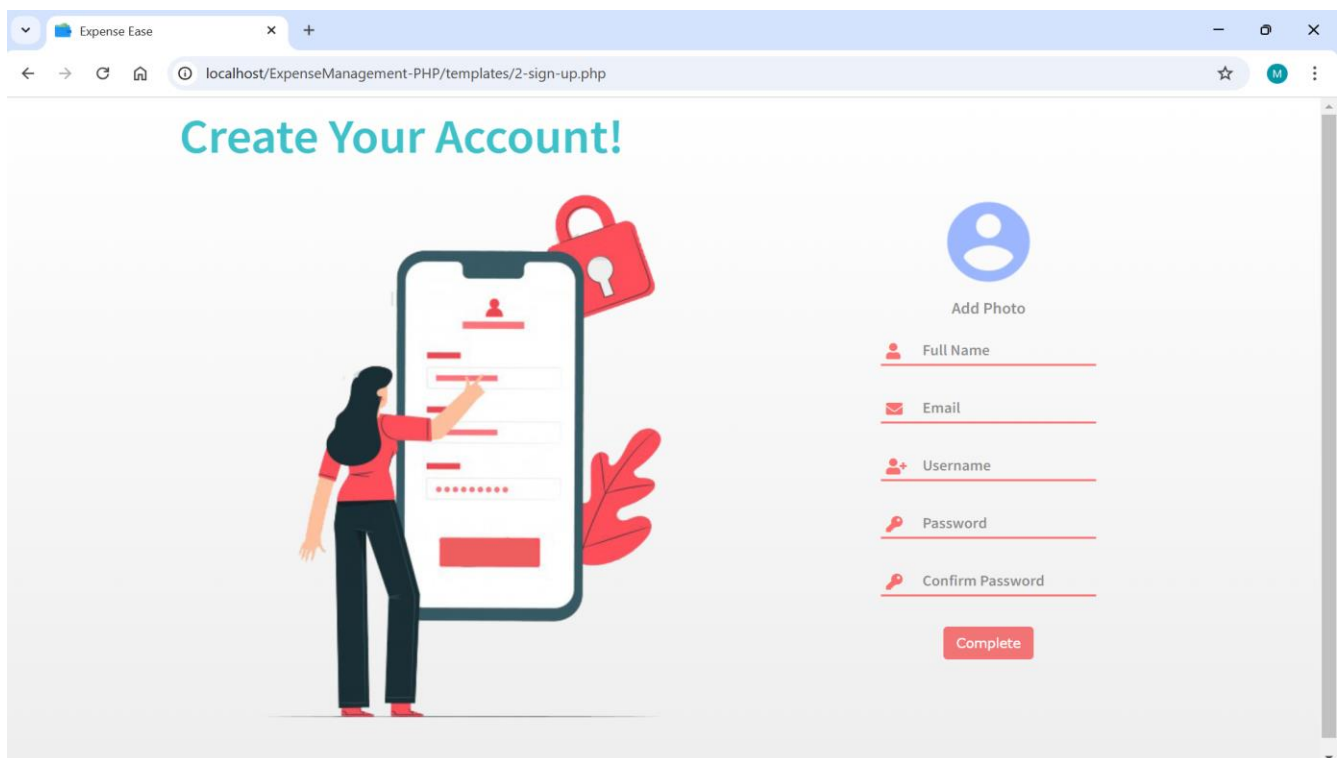
    define("BASE_URL", "http://localhost/ExpenseManagement-PHP/");
?>

```

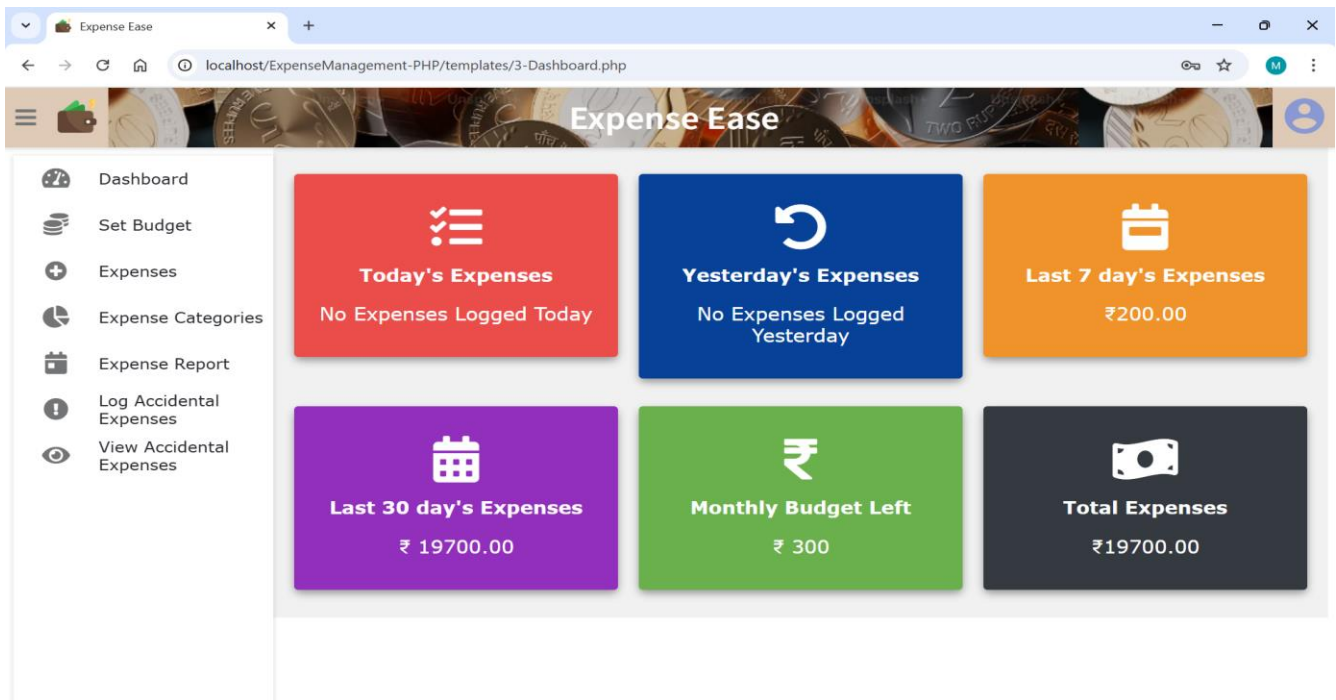
## 4.2 Output Screens:



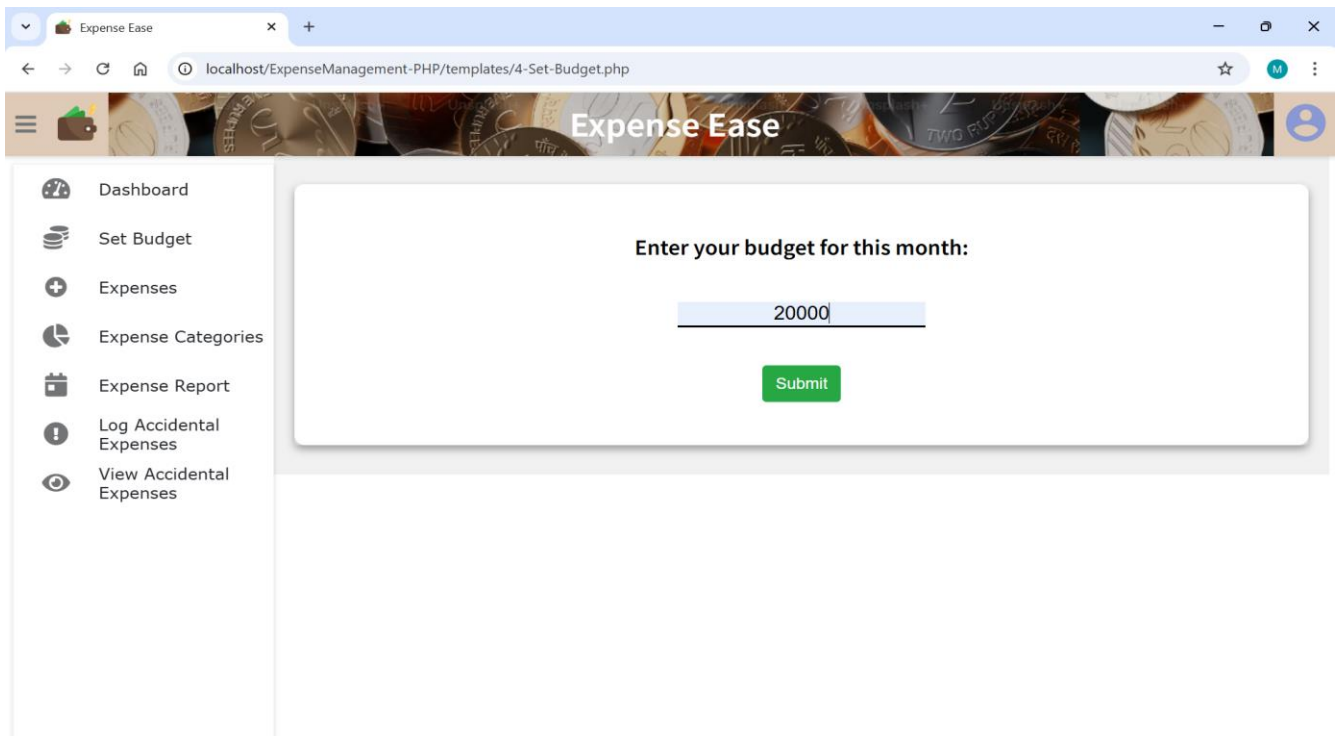
**Fig 4.2.1 Expense Ease Login Page**



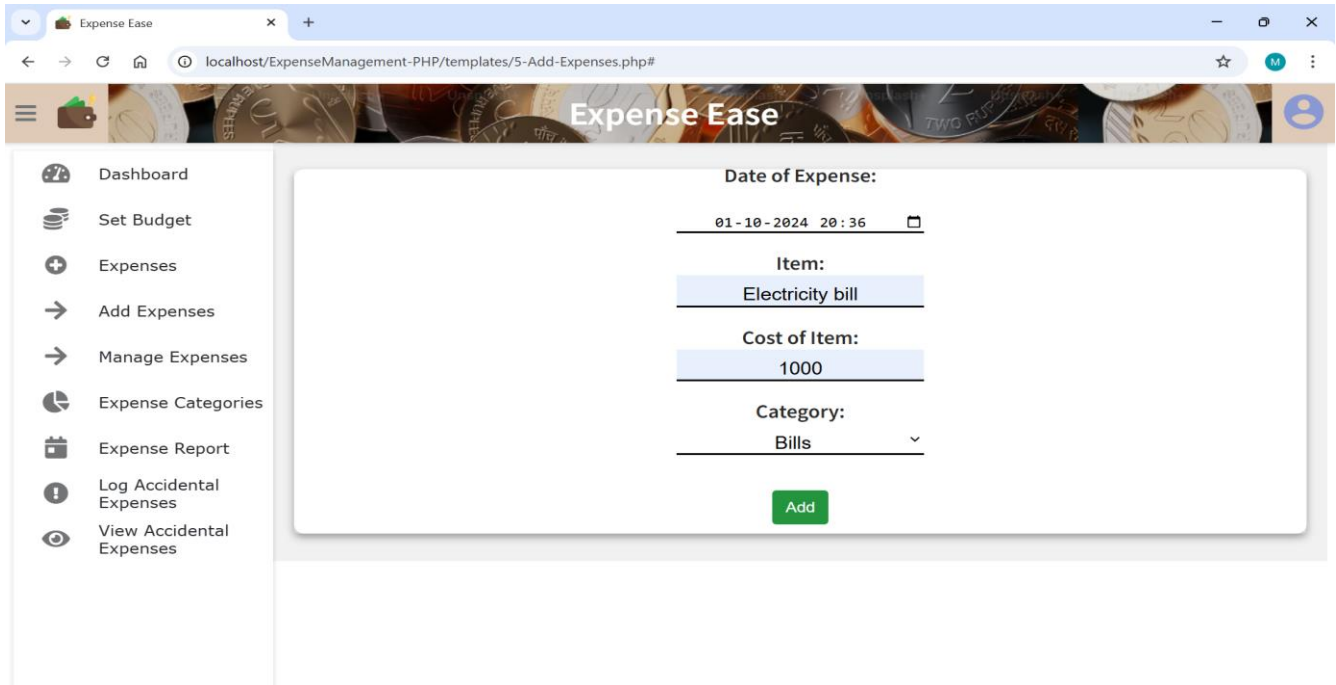
**Fig 4.2.2 Expense Ease Signup Page**



**Fig 4.2.3 Expense Ease Dashboard**

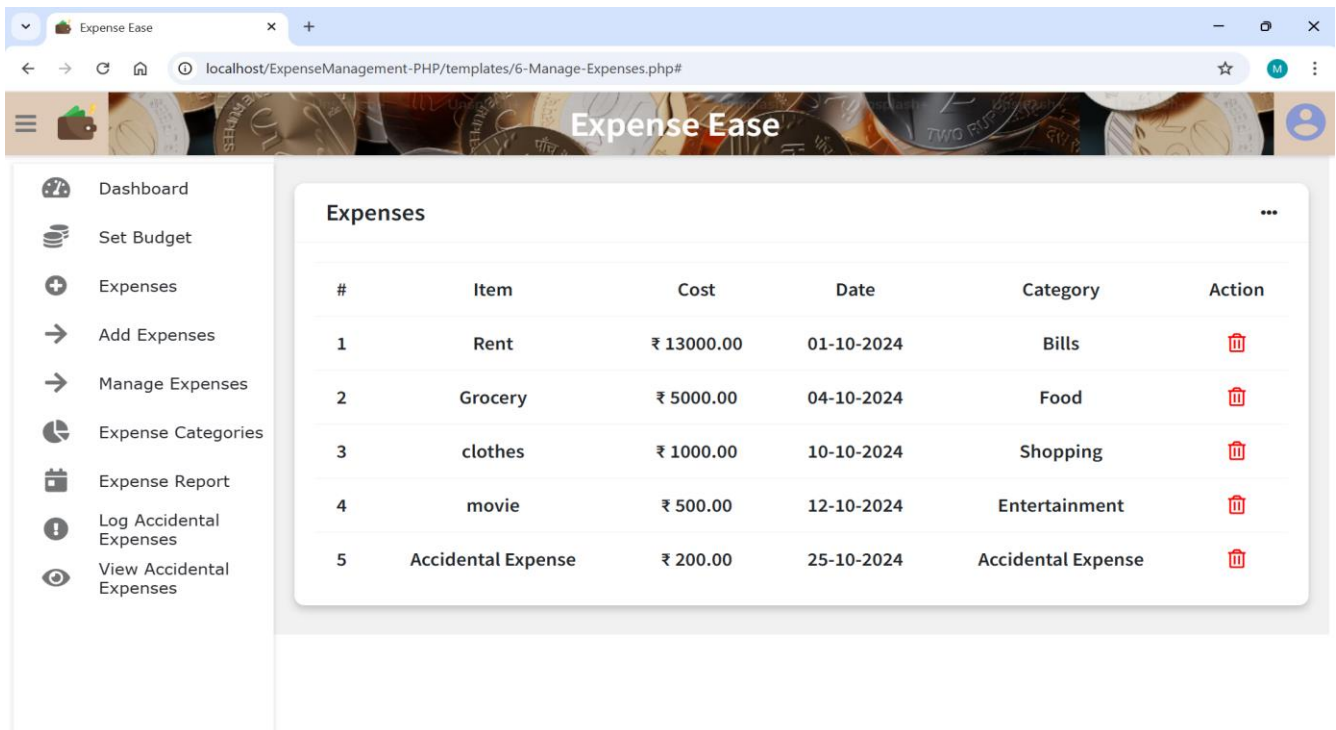


**Fig 4.2.4 Expense Ease Budget Setup**








The screenshot shows the 'Add Expenses' form in the Expense Ease application. The form is located at the URL `localhost/ExpenseManagement-PHP/templates/5-Add-Expenses.php#`. It features a sidebar with navigation options: Dashboard, Set Budget, Expenses, Add Expenses, Manage Expenses, Expense Categories, Expense Report, Log Accidental Expenses, and View Accidental Expenses. The main form area has a header 'Expense Ease' and a date field set to '01-10-2024 20:36'. Below this, there are input fields for 'Item' (Electricity bill), 'Cost of Item' (1000), and 'Category' (Bills). A green 'Add' button is at the bottom of the form.

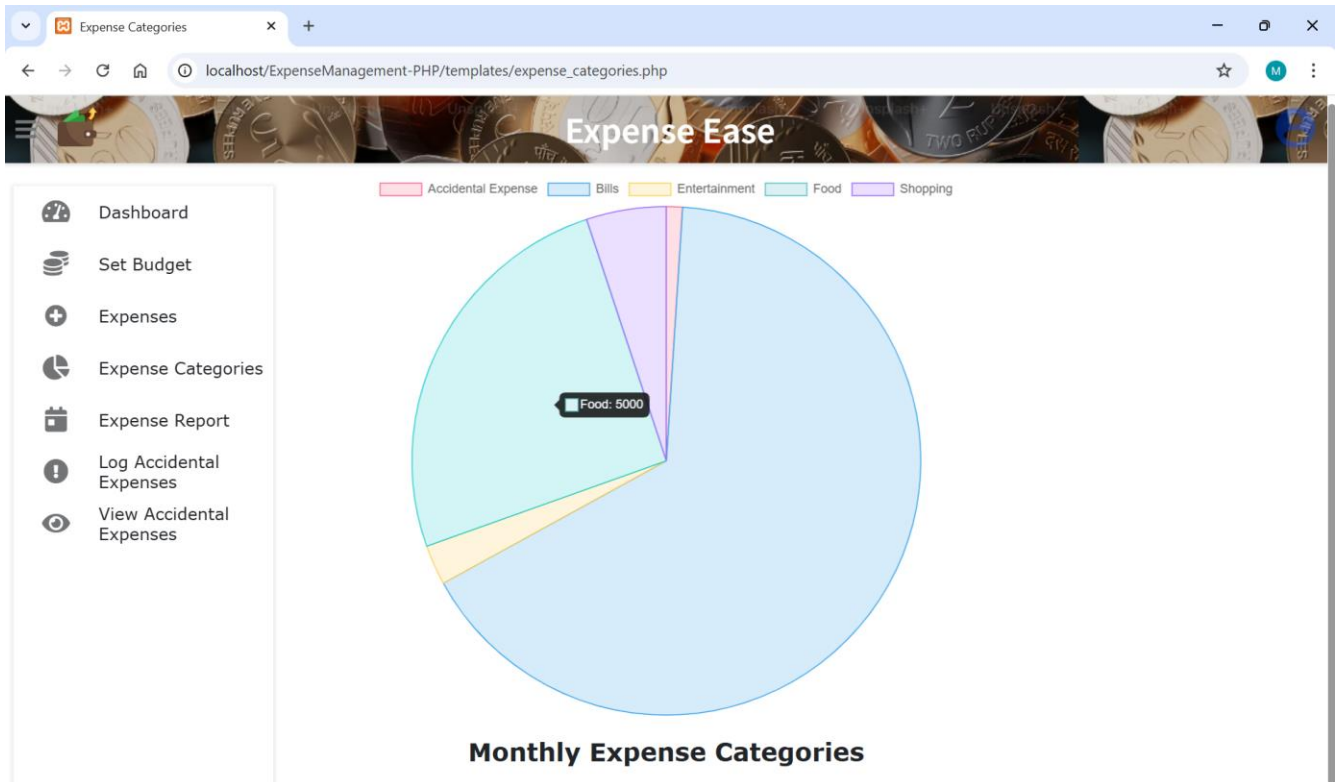
**Fig 4.2.5 Expense Ease – Add expenses**



The screenshot shows the 'Manage Expenses' page in the Expense Ease application. The page is located at the URL `localhost/ExpenseManagement-PHP/templates/6-Manage-Expenses.php#`. It features a sidebar with navigation options: Dashboard, Set Budget, Expenses, Add Expenses, Manage Expenses, Expense Categories, Expense Report, Log Accidental Expenses, and View Accidental Expenses. The main content area displays a table titled 'Expenses' with the following data:

| # | Item               | Cost       | Date       | Category           | Action  |
|---|--------------------|------------|------------|--------------------|---|
| 1 | Rent               | ₹ 13000.00 | 01-10-2024 | Bills              |  |
| 2 | Grocery            | ₹ 5000.00  | 04-10-2024 | Food               |  |
| 3 | clothes            | ₹ 1000.00  | 10-10-2024 | Shopping           |  |
| 4 | movie              | ₹ 500.00   | 12-10-2024 | Entertainment      |  |
| 5 | Accidental Expense | ₹ 200.00   | 25-10-2024 | Accidental Expense |  |

**Fig 4.2.6 Expense Ease – Manage expenses**

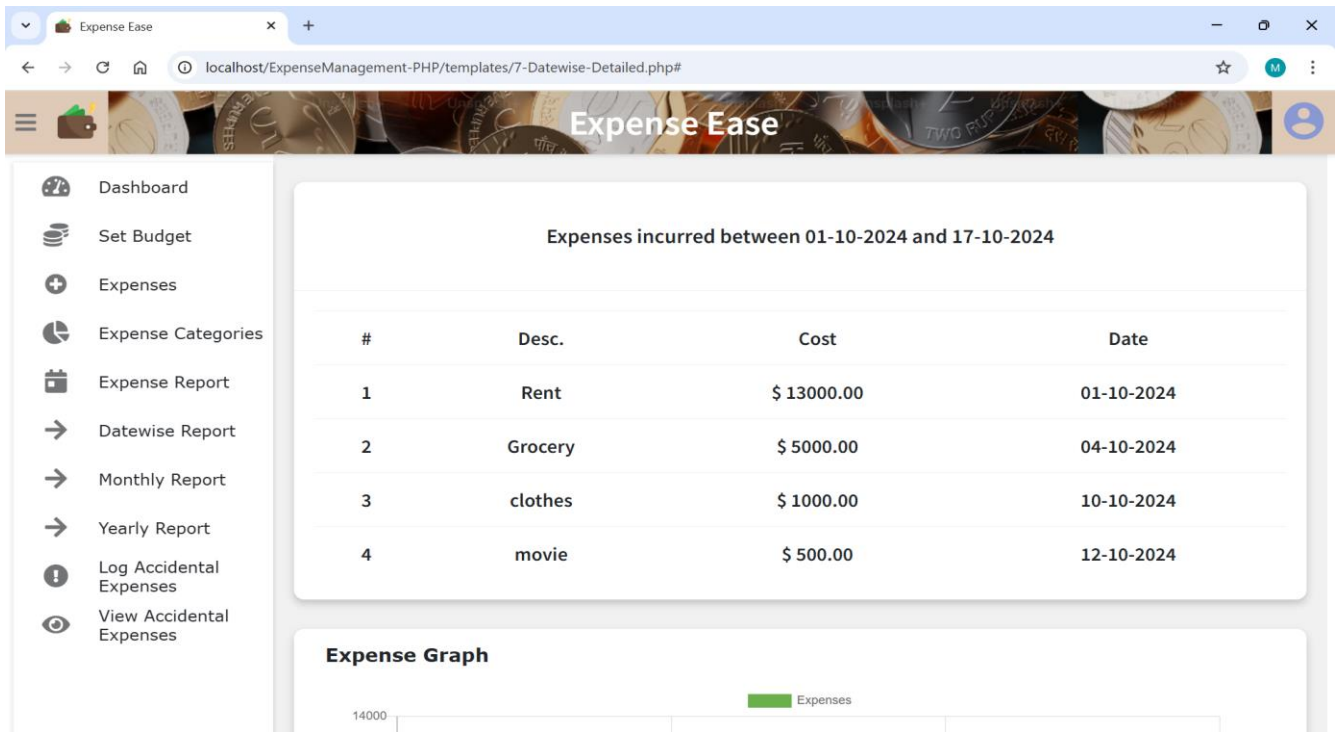


**Fig 4.2.7 Monthly Spending Categories**

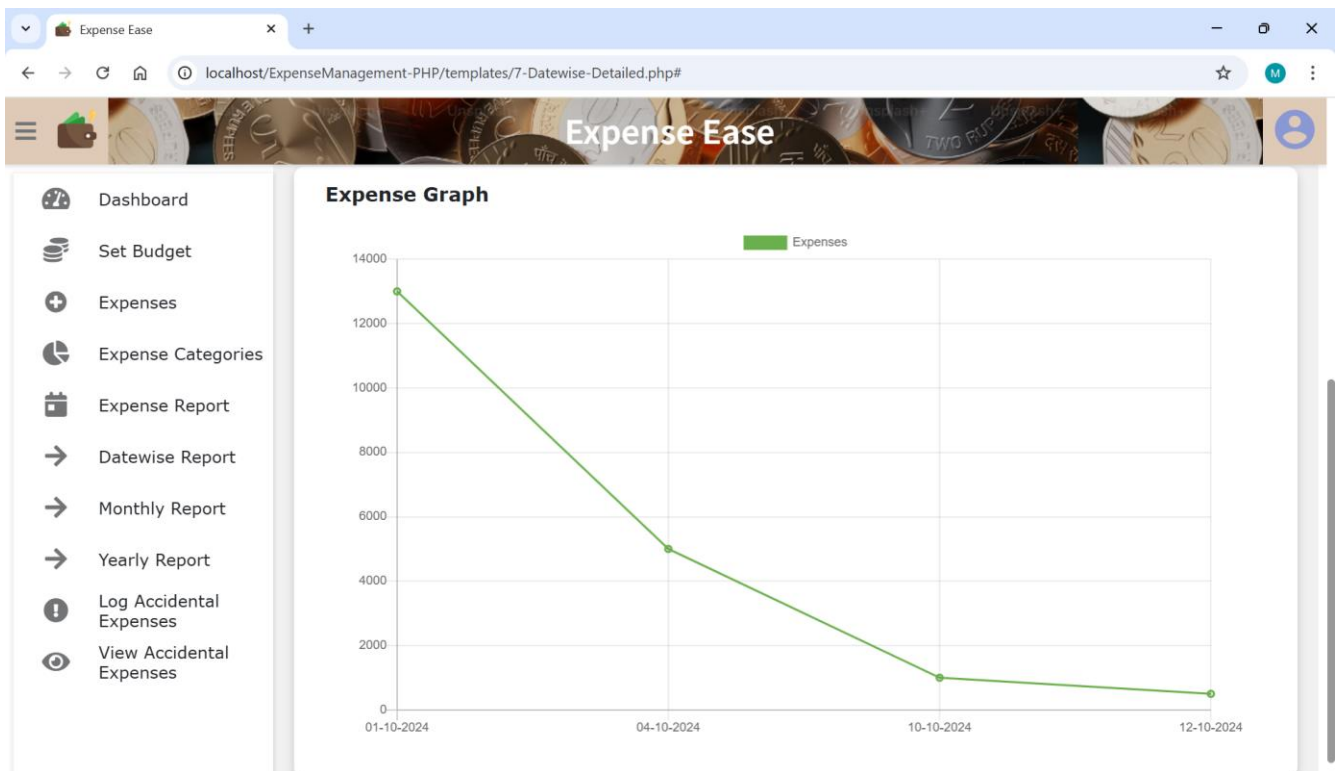
The image shows the "Expense Ease" web application interface for the "Datewise Expense Report". The browser address bar shows the URL `localhost/ExpenseManagement-PHP/templates/7-Datewise.php#`. The sidebar menu is the same as in the previous figure, but the "Datewise Report" item is highlighted. The main content area displays a form titled "Datewise Expense Report". The form has two date input fields: "From:" with the value "01-10-2024" and "To:" with the value "29-10-2024". Both fields have a calendar icon to the right. Below the date fields is a green "Submit" button.

| Field | Value      |
|-------|------------|
| From: | 01-10-2024 |
| To:   | 29-10-2024 |

**Fig 4.2.8 Expense Ease Report**



**Fig 4.2.9 Datewise Expense Report**

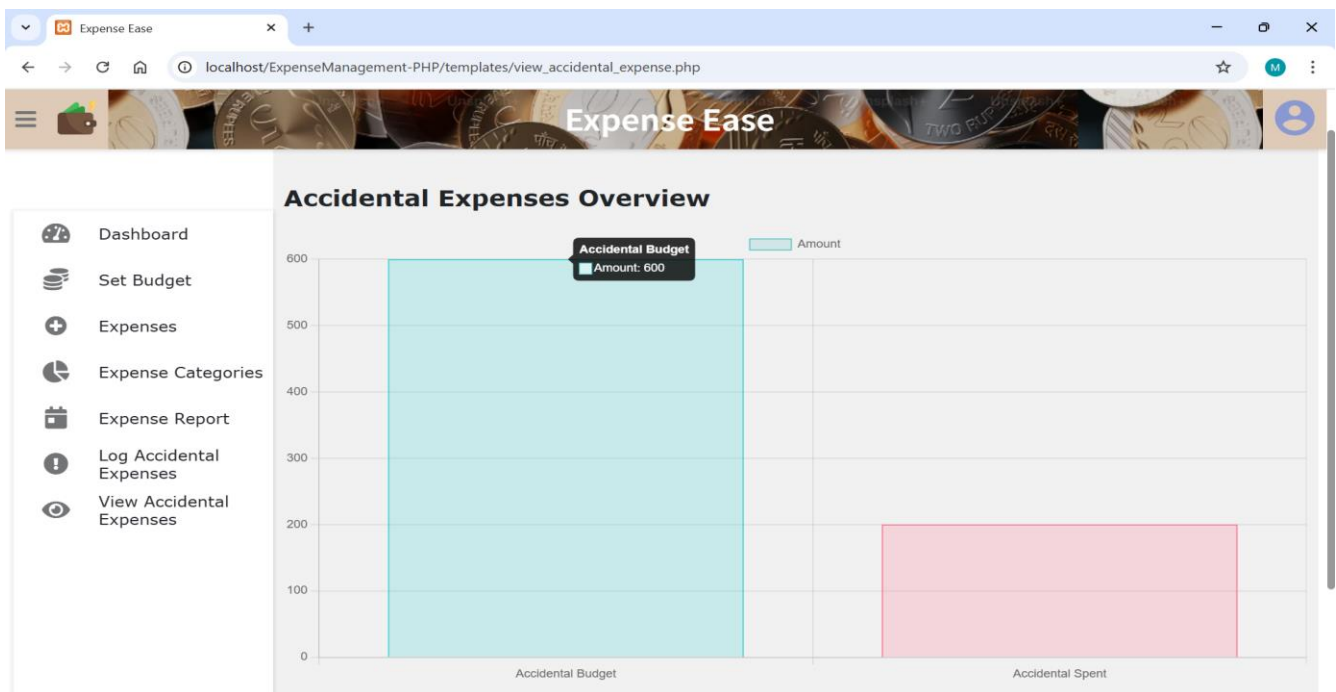


**Fig 4.2.10 Spending Graph**



The interface shows two main sections. The first section, 'Set Accidental Budget', has a text input for 'Accidental Budget Amount' with the value '500' and a green 'Set Budget' button. The second section, 'Log Accidental Expense', has a text input for 'Accidental Expense Amount' with the value '250' and a green 'Log Expense' button. A sidebar on the left contains navigation links: Dashboard, Set Budget, Expenses, Expense Categories, Expense Report, Log Accidental Expenses, and View Accidental Expenses. The browser address bar shows 'localhost/ExpenseManagement-PHP/templates/log\_accidental\_expense.php'.

**Fig 4.2.11 Accidental Budget Setup and Expense**



**Fig 4.2.12 Accidental Expenses Summary**



### **4.3 Testing:**

Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design and code generation.

## **5. CONCLUSION & FUTURE SCOPE**

The Expense Ease project provides a robust, user-friendly platform for efficient financial management and tracking. With its comprehensive features for managing income, expenses, budgeting, and generating detailed reports, it enables users to gain valuable insights into their financial habits and make informed decisions. Categorization options, multi-tenancy support, and the inclusion of features like profile image upload and accidental expense logging enhance the overall usability and personalization of the application. By simplifying financial tracking and reporting, it serves as an essential tool for individuals seeking to improve their financial awareness and accountability, ultimately fostering better budgeting and financial planning practices.

The future scope of this project includes expanding its capabilities with AI-driven spending insights, predictive budgeting, and personalized financial advice. Integrating with bank APIs for automatic transaction logging, adding multilingual support and introducing customizable budget templates can further enhance user experience. Additional features like collaborative budgeting for family members, goal-setting for savings and reminders for bill payments will make this project more comprehensive tool, meeting diverse financial needs and encouraging proactive financial management.

## 6. BIBILOGRAPHY

References are from research gateway, YouTube and google.

1. Dr. C K Gomathy, Article: A Study on the recent Advancements in Online Surveying, International Journal of Emerging technologies and Innovative Research (JETIR) Volume 5 | Issue 11 | ISSN :2349-5162, P. No:327-331, Nov-2018
2. “Why We Need to Start Budget Tracking Now” by David Ning.
3. <https://www.w3schools.com>
4. <https://youtu.be/r0lDDeVkaks?si=T-A8vQ51SD0ijSfK>
5. <https://youtu.be/N676lCQq4n8?si=AygcszsveLOzpfmK>