NITTE EDUCATION TRUST

NITTE MEENAKSHI INSTITUTE OF TECHNOLOGY
An Autonomous Institution Approved by UGC/AICTE/Govt. of Karnataka
Accredited by NBA (Tier-I) and NAAC 'A+' Grade
Affiliated to Visvesvaraya Technological University, Belagavi
Post Box No. 6429, Yelahanka, Bengaluru-560064, Karnataka, INDIA

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

MINI PROJECT REPORT

on

EXPENSE TRACKER

Submitted in partial fulfilment of the requirement for the award of Degree of

Bachelor of Engineering

in

Computer Science and Engineering

Submitted by:

DHEERAJ N	1NT23CS140
KHUSHI V	INT23CS109

Under the Guidance of

Mrs. Sowmya M. R.
Assistant Professor, Dept. of CS&E, NMIT

Department of Computer Science and Engineering
(Accredited by NBA Tier-1)

2024-2025

NITTE MEENAKSHI INSTITUTE OF TECHNOLOGY

(AN AUTONOMOUS INSTITUTION, AFFILIATED TO VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELGAUM , APPROVED BY AICTE & GOVT.OF KARNATAKA)

Department of Computer Science and Engineering

(Accredited by NBA Tier-1)

CERTIFICATE

This is to certify that the Expense Tracker is an authentic work carried out by **Dheeraj N (1NT23CS140)** and **Khushi V (1NT23CS109)** bonafide students of **Nitte Meenakshi Institute of Technology**, Bangalore in partial fulfilment for the award of the degree of ***Bachelor of Engineering*** in COMPUTER SCIENCE AND ENGINEERING of Visvesvaraya Technological University, Belgavi during the academic year ***2024-25.*** It is certified that all corrections and suggestions indicated during the internal assessment has been incorporated in the report. This project has been approved as it satisfies the academic requirement in respect of project work presented for the said degree.

Internal Guide	**Signature of the HOD**
Mrs. Sowmya M. R.	Dr. S Meenakshi Sundaram
Assistant Professor, Dept. CSE, NMIT	Professor and Head,
Bangalore	Dept. of CSE, NMIT Bangalore

Signature of Examiner (Internals**)**	**Signature of Examiner (**SEE**)**
1. _____	2. _____

DECLARATION

We hereby declare that

(i) The mini project work is our original work

(ii) This Mini Project work has not been submitted for the award of any degree or examination at any other university/College/Institute.

(iii) This Mini Project Work does not contain other persons' data, pictures, graphs or other information, unless specifically acknowledged as being sourced from other persons.

(iv) This Project Work does not contain other persons' writing, unless specifically acknowledged as being sourced from other researchers. Where other written sources have been quoted, then:

a) their words have been re-written, but the general information attributed to them has been referenced;

b) where their exact words have been used, their writing has been placed inside quotation marks and referenced.

(v) This Mini Project Work does not contain text, graphics or tables copied and pasted from the Internet, unless specifically acknowledged, and the source being detailed in the thesis and in the References sections.

NAME	USN	Signature
DHEERAJ N	1NT23CS140	
KHUSHI V	1NT23CS109	

Date: 03/01/2025

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of the people who made it possible, whose constant guidance and encouragement crowned our effort with success. I express my sincere gratitude to our Principal **Dr. H. C. Nagaraj**, Nitte Meenakshi Institute of Technology for providing facilities.

We wish to thank our HoD**, Dr. S Meenakshi Sundaram.** for the excellent environment created to further educational growth in our college. We also thank her for the invaluable guidance provided which has helped in the creation of a better project.

We hereby like to thank our *Guide Mrs. Sowmya M R*, Assistant Professor, Department of Computer Science & Engineering on her periodic inspection, time to time evaluation of the project and help to bring the project to the present form.

We also thank all our friends, teaching and non-teaching staff at NMIT, Bangalore, for all the direct and indirect help provided in the completion of the project.

NAME	USN	Signature
DHEERAJ N	1NT23CS140	
KHUSHI V	1NT23CS109	

Date: 03/01/2025

ABSTRACT

This report presents the development of an expense tracker tailored to help students manage their finances with ease and efficiency. As students often face challenges in balancing limited budgets, this tool is designed to provide a simple and effective solution for tracking daily expenses. The tracker allows students to record their expenses, categorize them into various groups such as food, transportation, and entertainment, and monitor their spending habits over time. The system includes a user-friendly dashboard that provides a comprehensive overview of the student's financial situation, displaying visual insights such as spending trends, budget comparisons, and expense categories.

The key feature of the expense tracker is its ability to display real-time data in an easily understandable format, empowering students to make informed decisions about their spending. By providing an at-a-glance summary of their financial activity, the dashboard helps students track their progress against set budgets and identify areas where adjustments may be needed. The tool aims to foster financial literacy among students, encouraging responsible spending and helping them develop effective budgeting habits. This project ultimately aims to equip students with the tools necessary to maintain control over their finances, promoting long-term financial health and independence.

DECLARATION

ACKNOWLEDGEMENT

ABSTRACT

Table of Contents

CHAPTER 1: INTRODUCTION

In today's world, financial literacy is a crucial skill, especially for students who are navigating the complexities of managing their personal finances on limited budgets. Students often face challenges such as balancing academic responsibilities with part-time jobs, managing day-to-day living expenses, and planning for future financial needs. Without proper tools or guidance, it can be easy to lose track of spending, which may lead to stress and financial instability. As a result, having an efficient system to manage finances is essential for students to achieve financial independence and security.

The expense tracker presented in this report is designed to address the need for a practical solution that enables students to track and control their spending. The tool aims to provide students with a clear understanding of their financial habits, empowering them to make more informed decisions and avoid unnecessary financial strain. With features tailored to the unique needs of students, such as simple categorization of expenses and an intuitive dashboard, this tracker serves as a comprehensive financial management tool. It promotes better money management practices, helping students stick to budgets and save for their future goals.

Motivation:

The motivation behind developing this expense tracker stems from the increasing importance of financial literacy in the student demographic. Many students enter higher education without a solid understanding of managing finances, and as a result, they may struggle to make responsible financial decisions. According to studies, poor financial management during the student years can lead to long-term debt, stress, and lack of savings. Furthermore, traditional methods such as manual tracking or spreadsheets may be time-consuming and prone to errors, making it difficult for students to consistently monitor their finances.

By creating a digital tool that simplifies the process of tracking expenses, students can have easy access to real-time data about their spending habits. The tracker also serves as a platform for developing good financial habits, as it provides immediate feedback on whether spending is within budget limits. The ease of use and accessibility offered by the expense tracker can significantly improve financial discipline, which is crucial for long-term financial well-being.

Objectives:

The primary objectives of this project are as follows:

1. **Design an Intuitive Expense Tracker**: Develop a user-friendly tool that allows students to easily input and categorize their daily expenses without requiring complex financial knowledge.

2. **Provide Financial Insights**: Offer students a clear overview of their financial situation through an interactive dashboard that visualizes spending trends, budgets, and expense categories, allowing for easier decision-making.

3. **Promote Budgeting and Savings**: Encourage students to set and adhere to personalized monthly budgets, thereby improving their budgeting skills and fostering healthy financial habits.

4. **Simplify Financial Monitoring**: Provide a simple solution for tracking and reviewing financial activity, eliminating the need for manual methods or cumbersome spreadsheets, and making financial management accessible to all students.

5. **Increase Financial Literacy**: Through the tracker, increase students' awareness of their spending habits and the importance of financial planning, ultimately helping them build a strong foundation for managing their finances in the future.

By achieving these objectives, the expense tracker aims to empower students to take control of their financial lives, avoid common financial pitfalls, and gain the skills necessary to make informed decisions that will positively impact their financial future.

CHAPTER 2: METHODOLOGY

The development of the **Expense Tracker App** follows a structured methodology focusing on web application development, incorporating best practices in software design, user experience, and data management. The application was built using **C# .NET 7 MVC** and **Microsoft SQL Server**, ensuring high performance, scalability, and security. Below is the step-by-step methodology used in creating the app:

1. Requirements Gathering:

The first step in the development process was to gather requirements and define the goals of the expense tracker. The main objective was to build an intuitive platform that helps users manage their personal finances by tracking expenses, setting budgets, and gaining insights into spending habits. Features such as user registration, categorization of expenses, budget tracking, and data visualization were prioritized to offer users an all-encompassing tool for financial management.

2. System Design:

The system was designed to be modular and scalable. The key components of the design included:

- **User Authentication and Registration:** Leveraging **ASP.NET Identity**, the app enables secure user registration and login functionality. This ensures that each user's data is protected and can be accessed only by the respective account holder.
- **Expense Management:** Users can add, update, or delete expenses, categorizing them into predefined or custom expense categories. This modular structure allows users to track different types of expenses effectively.
- **Budget Management:** The app allows users to set spending limits for each category and track whether their spending is within the set budget, offering insights on budget adherence.
- **Data Visualization:** To provide a clear understanding of spending patterns, the app includes interactive charts and graphs powered by **Chart.js**, enabling users to visualize their financial behavior and make informed decisions.

3. Technology Stack:

The following technologies were used in the development of the Expense Tracker App:

- **Framework:** The app was developed using **C# .NET 7 MVC**, an ideal framework for building robust, scalable, and maintainable web applications. The MVC architecture ensured that the application was well-organized, with separate layers for data, business logic, and presentation.
- **Database: Microsoft SQL Server** was chosen as the relational database management system for storing user data, expenses, categories, and budgets. It provides reliable data storage and querying capabilities.
- **Authentication: ASP.NET Identity** was used to manage user authentication and authorization. This system provides a secure method for registering, logging in, and managing user sessions.
- **Charting: Chart.js** was integrated into the application to generate dynamic charts and graphs that allow users to visually analyze their spending trends and budget performance.

CHAPTER 3: IMPLEMENTATION

The implementation of the Expense Tracker App followed a systematic approach, with each feature and functionality carefully designed and developed to meet the requirements. This section provides a step-by-step overview of the implementation process for each key feature of the application.

1. User Registration and Authentication:

The first component of the app is user registration and authentication, enabling secure user access to the application. This was implemented using ASP.NET Identity, a powerful framework for managing user accounts in a .NET-based web application.

Steps for Implementation:

- **ASP.NET Identity Configuration:** The ASP.NET Identity library was added to the project using NuGet Package Manager to handle user registration, login, and password recovery.

- **Registration:** A registration form was created where users can enter details such as email, password, and confirm their password. Upon successful registration, the user is automatically logged in and redirected to the dashboard.

- **Login and Logout:** The login functionality checks the user's credentials (email and password) against the database. Upon successful authentication, the user gains access to their personal dashboard. A logout feature allows users to end their session and be redirected to the home page.

- **Password Recovery:** A password recovery option was integrated to enable users to reset their passwords via email if forgotten.

2. Expense Management:

The core functionality of the app is expense management, where users can log, edit, and delete their expenses. The Expense Management feature allows users to track their spending by adding expenses, categorizing them, and associating them with specific amounts.

Steps for Implementation:

- **Expense Model:** A model was created to represent an expense in the application, with fields such as Amount, Description, Category, Date, and UserId. The UserId field establishes a one-to-many relationship between users and their expenses.

```
public class Expense

{

   public int Id { get; set; }

   public decimal Amount { get; set; }

   public string Description { get; set; }

   public string Category { get; set; }

   public DateTime Date { get; set; }

   public string UserId { get; set; }

}
```

- **Add Expense:** A form was created where users can input the amount spent, a description of the expense, and choose a category. Upon submission, the expense data is saved in the Microsoft SQL Server database.

```
public async Task<IActionResult> Create(Expense expense)

{

   if (ModelState.IsValid)

   {

      expense.UserId = _userManager.GetUserId(User); // Associating expense with logged-in user

      _context.Add(expense);

      await _context.SaveChangesAsync();

      return RedirectToAction(nameof(Index));

   }

   return View(expense);

}
```

- **Update and Delete Expense:** Users can view their existing expenses and modify or delete them as needed. Each expense item has an edit and delete button that triggers the respective actions.

```
public async Task<IActionResult> Edit(int? id)

{

   if (id == null)

   {

      return NotFound();

   }



   var expense = await _context.Expenses.FindAsync(id);

   if (expense == null)

   {

      return NotFound();

   }



   return View(expense);

}



public async Task<IActionResult> Delete(int? id)

{

   if (id == null)

   {

      return NotFound();

   }
```

```
    var expense = await _context.Expenses.FindAsync(id);
    if (expense == null)
    {
        return NotFound();
    }

    _context.Expenses.Remove(expense);
    await _context.SaveChangesAsync();
    return RedirectToAction(nameof(Index));
}
```

3. Expense Categorization:

Categorizing expenses is a key feature to help users better understand their spending. The app allows users to categorize their expenses into predefined categories, such as food, transportation, entertainment, etc. Users can also add custom categories.

Steps for Implementation:

- **Category Model:** A model for categories was created to store predefined and custom expense categories.

```
public class Category
{
    public int Id { get; set; }
    public string Name { get; set; }
    public string UserId { get; set; }
}
```

- **Category Dropdown:** When adding or editing an expense, users can select a category from a dropdown list populated with existing categories. The list is dynamic and updated based on the user's selected categories.

```
public IActionResult Create()

{

   ViewData["CategoryId"] = new SelectList(_context.Categories.Where(c => c.UserId ==
_userManager.GetUserId(User)), "Id", "Name");

   return View();

}
```

4. Budget Management:

The Budget Management feature allows users to set a budget for each expense category and track how their actual spending compares with the set limits. This helps users avoid overspending and stay within their financial limits.

Steps for Implementation:

- **Budget Model:** A budget model was created to store the budget amount for each category.

```
public class Budget

{

   public int Id { get; set; }

   public decimal Amount { get; set; }

   public string Category { get; set; }

   public string UserId { get; set; }

}
```

- **Setting and Tracking Budgets:** Users can enter their budget for each category, which is then compared with their actual spending. The app calculates whether the user is within budget or over budget for each category.

```
var budget = _context.Budgets.FirstOrDefault(b => b.UserId ==
_userManager.GetUserId(User) && b.Category == expense.Category);

if (budget != null && expense.Amount > budget.Amount)

{

   // Notify user or visually indicate overspending }
```

5. Data Visualization:

To visualize spending patterns, the app integrates Chart.js to generate interactive charts and graphs that display data such as total spending, spending by category, and budget adherence. This enables users to get a visual representation of their financial habits and track progress.

Steps for Implementation:

- **Chart.js Integration:** The data is sent from the controller to the view, where it is formatted in JSON format and passed to Chart.js to generate charts.

```
var ctx = document.getElementById('myChart').getContext('2d');

var myChart = new Chart(ctx, {

    type: 'pie',

    data: {

        labels: ['Food', 'Transport', 'Entertainment'],

        datasets: [{

            label: 'Expense Distribution',

            data: [500, 300, 200],

            backgroundColor: ['#FF6347', '#36A2EB', '#FFCE56']

        }]

    }

});
```

- **Real-Time Data:** The charts update in real-time based on the user's input. For instance, when new expenses are added, the chart automatically reflects the changes.

6. Deployment and Hosting:

After developing and testing the application, it was deployed on a local machine using Visual Studio. To host the application online, the following steps were taken:

- **SQL Server Configuration:** The SQL Server database connection string was configured in the appsettings.json file to point to the production database.

CHAPTER 4: RESULTS AND DISCUSSION

This chapter includes screenshots of the application interface to provide a visual reference for the functionalities described in the previous chapters.

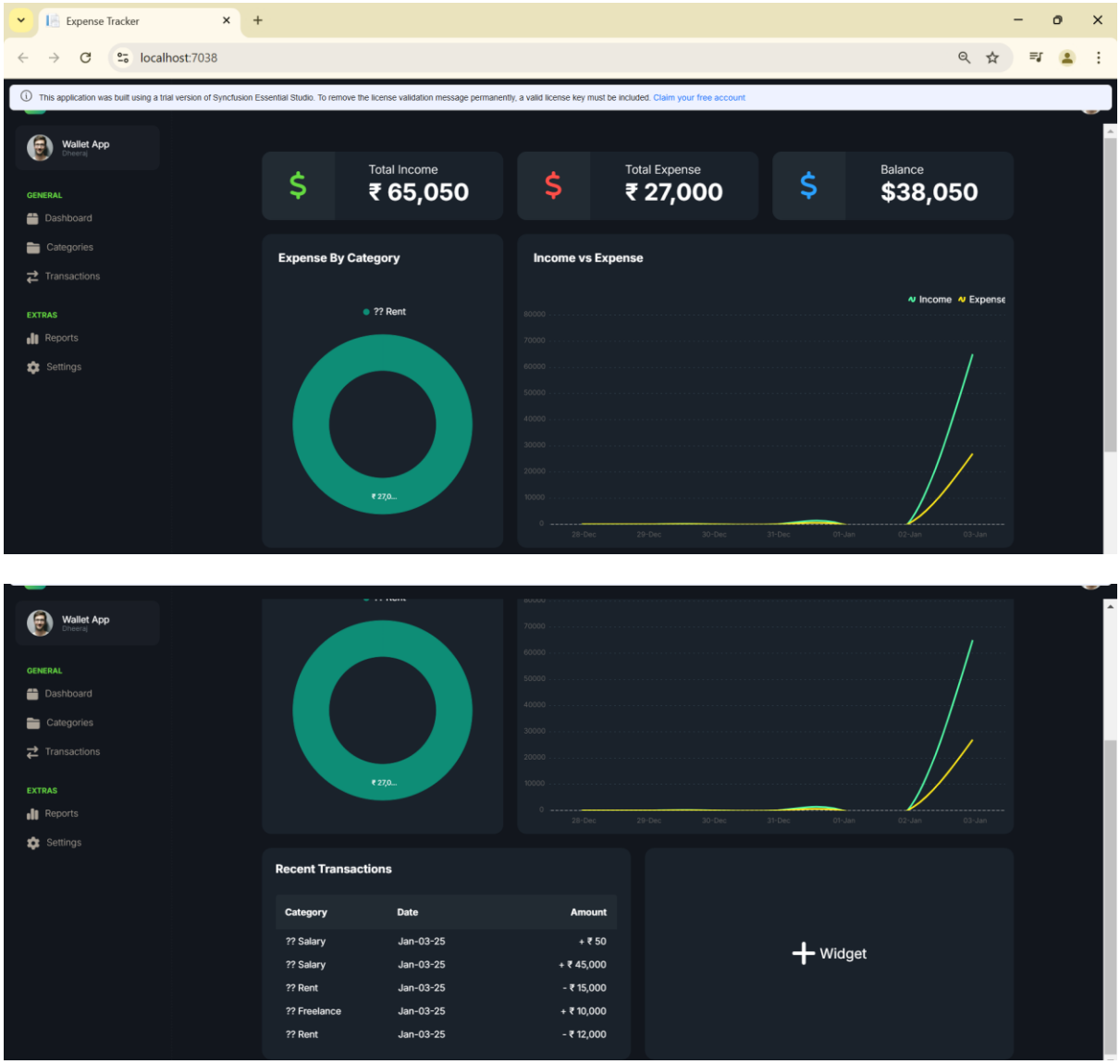


Fig 4.1 Dashboard Screenshot

- A snapshot of the user's total income, expenses, and remaining balance for the month.
- Pie chart visualizing the breakdown of expenses by category.

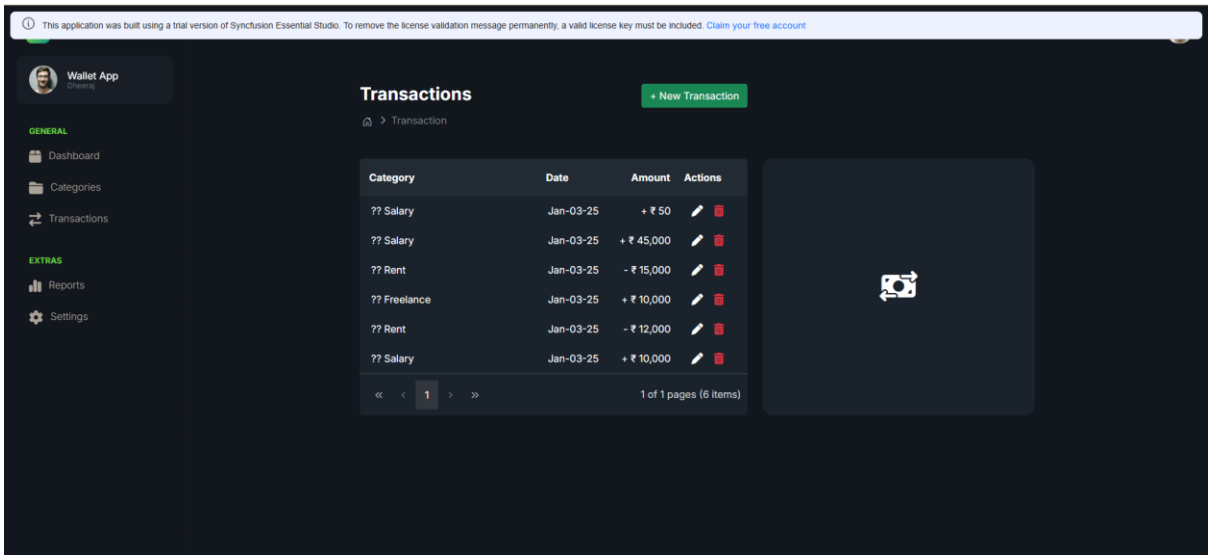
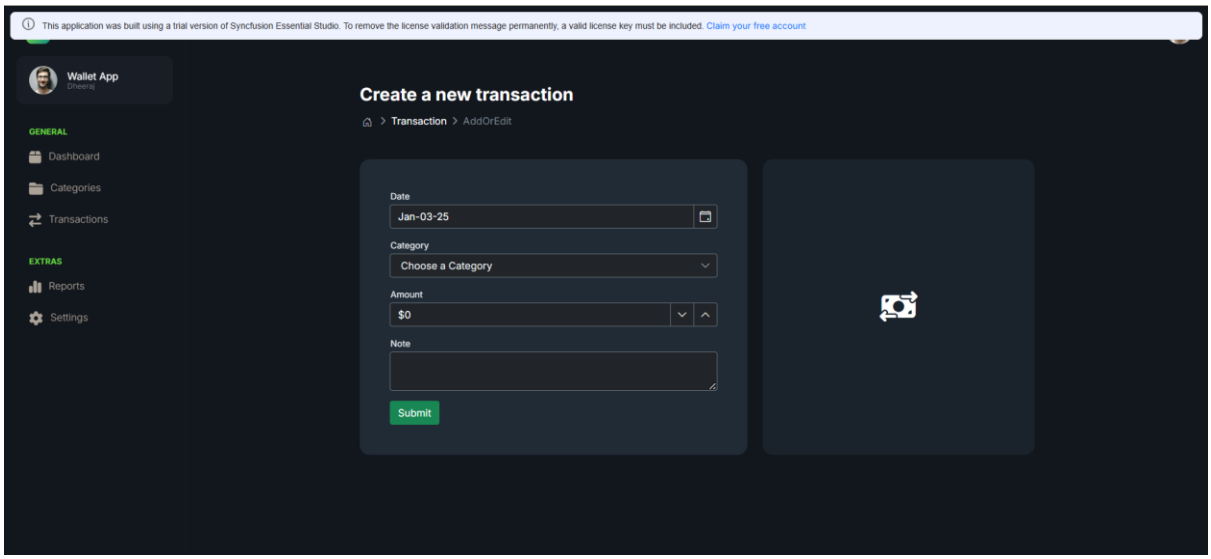


Fig 4.2 Add Transaction Screen

- A form where the user can input transaction details (amount, category, date).



4.3 Screen after few transactions are added into different categories

- Displays a list of categories with their respective budget limits and real-time spending comparison.

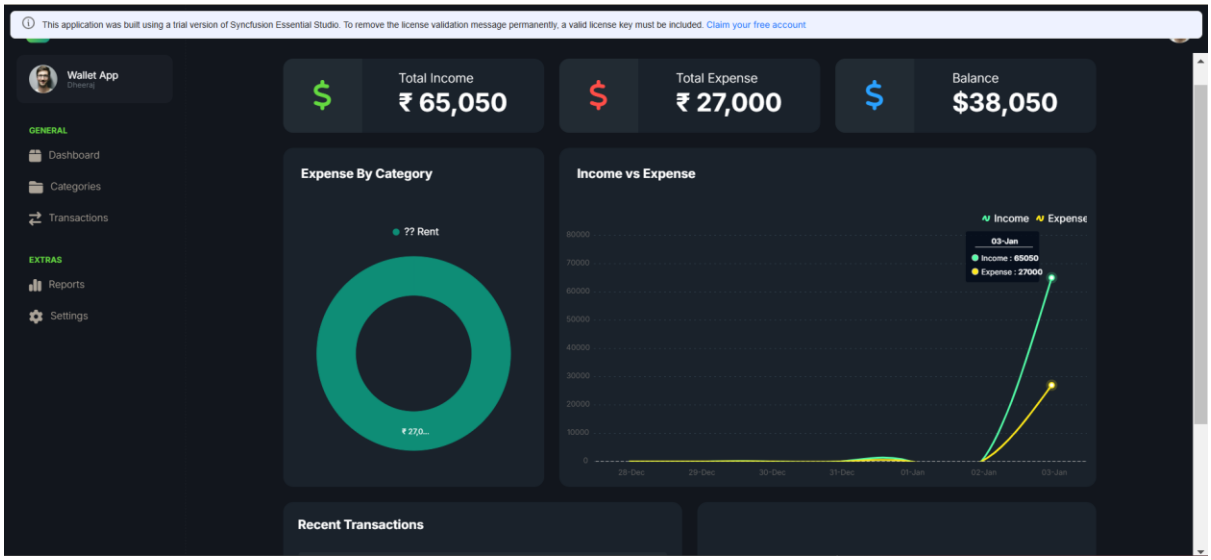


Fig 4.5 Expense Analysis Graphs

- Interactive bar charts or pie charts that show a breakdown of expenses by category over a given period.

CHAPTER 5: CONCLUSIONS

The Expense Tracker App is an effective tool designed to help individuals, especially students, track and manage their personal finances. The app provides a simple yet powerful interface for users to log their expenses, categorize them, set budgets, and visualize their spending habits through charts. This makes it easier for users to understand where their money is going and make smarter financial decisions.

Built using C# .NET 7 MVC and Microsoft SQL Server, the app ensures a smooth user experience and secure data management. The integration of ASP.NET Identity allows users to securely register, log in, and manage their personal data, while Chart.js is used to generate interactive graphs that display spending trends, helping users visually track their finances.

The key features of the app include the ability to add, update, and delete expenses, categorize them, set spending limits, and track budget adherence. These features are designed to be intuitive, making it easy for users to keep track of their finances and stay within their budget.

Throughout the development process, the focus was on building a user-friendly application that can be easily used by individuals without requiring any advanced technical knowledge. While the current version of the app provides essential features for personal finance management, future improvements could include adding recurring expenses, advanced reports, and reminders to further enhance its functionality.

In conclusion, the Expense Tracker App successfully meets its goal of helping users track and manage their expenses. It provides a solid foundation for personal finance management and offers a valuable tool for anyone looking to better understand and control their spending. With future updates and improvements, the app could become an even more powerful tool for financial planning and budgeting.

REFERENCES

1. **Microsoft Documentation – ASP.NET Identity**
   Available at: https://learn.microsoft.com/en-us/aspnet/core/security/authentication/identity

This resource provided a comprehensive guide to setting up ASP.NET Identity for user authentication, which was implemented in the Expense Tracker App for secure user registration and login.

2. **Chart.js Documentation**
   Available at: https://www.chartjs.org/docs/latest/

This documentation helped in integrating Chart.js for data visualization. It provided detailed instructions on creating interactive charts and graphs to display spending trends within the app.

3. **Microsoft Documentation – Entity Framework Core**
   Available at: https://learn.microsoft.com/en-us/ef/core/

This source provided insights into Entity Framework Core, which was used for database management and to manage relationships between different models, such as expenses and categories.

4. **Microsoft Documentation – SQL Server**
   Available at: https://learn.microsoft.com/en-us/sql/sql-server/

This documentation was essential for understanding how to set up and configure SQL Server for storing user data, expenses, categories, and budgets in the app.

5. **C# Programming Guide**
   Available at: https://learn.microsoft.com/en-us/dotnet/csharp/

This guide was used throughout the development process to reference the syntax and best practices for C# programming, specifically for building a web application with C# .NET 7 MVC.

6. **ASP.NET MVC Documentation**
   Available at: https://learn.microsoft.com/en-us/aspnet/core/mvc/

This resource provided essential information on developing web applications using the MVC architecture in ASP.NET, which was key in structuring the Expense Tracker App.

7. **Stack Overflow**
   Available at: https://stackoverflow.com/

Stack Overflow was used throughout the development process to troubleshoot errors, seek coding solutions, and get advice on implementation best practices from the developer community.

8. **W3Schools – SQL Tutorial**
   Available at: https://www.w3schools.com/sql/

This tutorial was useful for learning SQL queries to interact with the Microsoft SQL Server database, particularly when handling operations like adding, updating, and deleting expenses.