

Budget Manager

Sachin Kolekar¹; Aayush Mali²; Mahi Nilesh Bagrecha³; Vivek Gajanan Mali⁴;
Mansvi Pramod Magar⁵; Sayali Bhagavan Mahangade⁶; Manya Puri⁷

¹Project Guide, ²GL-(09), ³AGL-(08), ⁴(10), ⁵(11), ⁶(07), ⁷(12)

^{1,2,3,4,5,6,7}F. Y. B. Tech Students' Applied Science & Engineering Project1 (ASEP2) Paper, SEM 2 A. Y. 2024-25 Vishwakarma Institute of Technology, Pune, INDIA

Publication Date: 2025/11/29

Abstract: This project involves the creation of a web-based Budget Manager designed to assist users in monitoring, organizing, and planning their personal finances. Unlike conventional budgeting platforms, this system introduces advanced functionalities such as AI-based financial insights, gamification features to increase user interaction, shared wallets for joint financial management, and integrated tools for bill splitting and expense tracking. It also supports offline usage and voice assistant integration, addressing the needs of users concerned with data privacy and accessibility. The application prioritizes a user-friendly design and practical utility. Evaluations and feedback indicate that the platform promotes better financial habits, encourages saving, and simplifies everyday expense tracking. Overall, this Budget Manager Website showcases how intelligent features and thoughtful design can enhance digital personal finance tools.

How to Cite: Sachin Kolekar; Aayush Mali; Mahi Nilesh Bagrecha; Vivek Gajanan Mali; Mansvi Pramod Magar; Sayali Bhagavan Mahangade; Manya Puri (2025). Budget Manager. *International Journal of Innovative Science and Research Technology*, 10(11), 1938-1944. <https://doi.org/10.38124/ijisrt/25nov421>

I. INTRODUCTION

A budget is essentially a financial representation of an intended plan of action for an upcoming period. It outlines a structured, monetary plan approved before the start of the budget cycle, detailing expected income, expenses, and any projected capital allocation. This financial plan may be developed by either building upon past budgets or actual results (incremental budgeting) or by constructing it from scratch, as seen in zero-based budgeting approaches.

The terms budget and budgetary control are frequently used synonymously, as they both relate to systems used for managerial oversight. However, budgetary control more specifically refers to the comprehensive process of using budgets to assist management in executing its core functions, including planning, coordinating activities, and ensuring effective control over financial resources.

As digital financial management becomes increasingly prevalent, budget tracking tools have grown in popularity for their ability to help users monitor income, expenses, and savings goals. Despite their widespread adoption, many existing solutions are often rigid, offering limited personalization, minimal user engagement, and inadequate support for collaborative financial planning (Smith & Kumar, 2021). This research presents an enhanced budget management system designed to address these gaps by integrating intelligent, accessible, and user-focused functionalities.

The proposed system introduces an AI-powered budget assistant that delivers real-time financial insights and personalized suggestions, aiming to reduce overspending and promote savings behavior (Lee et al., 2022). To further improve user engagement, the platform employs gamification techniques—including rewards, achievements, and progress indicators—proven to foster long-term behavioral change in financial habits (Nguyen & Hall, 2020).

Recognizing the importance of collaborative financial management, the system supports shared wallets, enabling couples, roommates, or families to manage joint budgets transparently (Chen et al., 2021). It also includes bill splitting and debt tracking tools, which streamline shared expense management and reduce interpersonal financial friction (Rao & Singh, 2020). For privacy-conscious users, an offline and manual mode allows full use of the platform without requiring bank integration. Additionally, the system incorporates voice assistant capabilities, allowing natural language queries such as “How much can I spend today?”, thereby enhancing usability and accessibility (Patel & Deshmukh, 2023).

By combining these innovations, the proposed platform seeks to redefine budget management as an intelligent, interactive, and socially adaptable experience, addressing unmet needs in current digital finance tools.

II. LITERATURE REVIEW

- A. Sharma et al., This study proposed a personal finance application that allows users to set budgets, track expenses, and receive financial summaries. The research emphasized the importance of visual analytics, such as graphs and pie charts, in helping users make better financial decisions. Usability testing revealed that simplicity and intuitive design play a major role in user retention.
- M. Verma and K. Gupta, The authors developed a mobile application with bank integration to automate expense tracking. Their system categorized transactions in real time and provided monthly spending reports. However, the study noted concerns over data privacy and the reluctance of users to link bank accounts.
- S. Das et al., This research introduced a gamified budgeting app aimed at improving financial discipline among students. Features like point-based rewards and budget challenge levels increased user engagement. The study demonstrated that gamification techniques lead to higher app usage and goal completion rates.
- L. Rodriguez and T. Wang, The study focused on a collaborative financial tool for couples and families. It allowed users to create shared wallets and track joint expenses. The research highlighted how transparency in shared financial responsibilities reduces conflicts and improves communication.
- H. Kim et al., This paper examined the integration of voice assistants in mobile finance applications. It found that voice commands improved accessibility, especially for users with disabilities. The study achieved over 80% accuracy in command interpretation and emphasized the importance of natural language processing in financial tools.

III. METHODOLOGY

The prototyping model is a system development approach where an initial version of the product—called a prototype—is created, tested, and refined iteratively until a satisfactory version is achieved, which then serves as the foundation for the final system or product [5]. This model has been selected for the project because it is highly user-focused, aiming primarily to fulfill the end-user's requirements. It emphasizes close collaboration with the user, where developers build a prototype and present it to the client for feedback. The client then suggests improvements, which are incorporated into the prototype. This cycle continues until the client is fully satisfied [6]. The prototyping model consists of six stages: The first stage, *Requirements*, involves collecting and analyzing relevant information. The second stage, *Quick Design*, includes creating a basic structure or layout of the system. In the third stage, a prototype is developed. The fourth stage involves getting feedback from the user, which leads to the fifth stage—refining the prototype. Stages four and five are repeated as needed. The final stage is the *Implementation and Maintenance* of the completed system.

➤ Requirements Phase

The goal of this phase is to gather detailed information about the project to be developed. This includes analyzing similar past projects and engaging with potential users to understand their expectations from the system.[1] Based on this analysis and the collected requirements, developers gain a foundational understanding of the project. At this point, elements such as the problem background, project scope, objectives, and anticipated outcomes are defined.

➤ Quick Design Phase

This phase transforms the initial ideas gathered during the requirements phase into a simple, conceptual design.[2] Although not a finalized version, this preliminary design outlines the essential features such as input and output components. The focus is on the user-visible aspects rather than the internal structure, which helps set the stage for the prototype's development.

➤ Build Prototype Phase

Using the information compiled in the earlier phases, an initial version of the system—the prototype—is developed. This prototype emphasizes areas of interest to the user, such as the interface, key functionalities, and overall layout. Concentrating on these aspects helps control both development time and cost while ensuring the user's priorities are addressed.

➤ User Evaluation Phase

After the prototype is completed, it is presented to the user for evaluation.[4] During this phase, users interact with the system to identify its strengths and weaknesses. Their feedback helps uncover missing features, usability issues, or errors. If users are not fully satisfied, their input is collected to guide the next phase—refining the prototype.

➤ Refining Prototype Phase

Closely linked to the user evaluation phase, this stage involves making improvements to the prototype based on user feedback.[1] All suggested changes and enhancements are incorporated into the new version of the prototype, which is then presented again to the users for further review. This cycle continues until the users are satisfied and no major issues remain. Once approved, the refined prototype becomes the foundation for the final system.

➤ Implementation and Maintenance Phase

Once the objectives of the prototype have been fulfilled, it is no longer used, and the actual system is developed using a suitable software development model. The prototype serves primarily to help developers fully understand the client's requirements. With those needs clearly identified, the final system is built and delivered to the client, ensuring it meets all agreed-upon expectations. [5] After deployment, regular maintenance is carried out to reduce downtime and avoid major system failures.

➤ System Requirements Analysis

System requirements analysis involves identifying and defining what users expect from a new or updated software application. This phase includes all activities related to

understanding, documenting, verifying, and managing user needs and system specifications. The process is typically divided into several categories, such as functional

requirements, non-functional requirements, hardware and software requirements, and user requirements.

Table 1 System Requirements Analysis

Modules	Functional requirement
Signup	- User needs to insert unique username that different than others. - After signup will directed to login page
Login	- User could login using an already sign up account - After login will directed to profile page
Home feed	- Will show pinned savings module, daily, weekly, and monthly spending, and recent transaction - Can access insert savings and insert spending module - Can access details of pinned savings module
Profile	- Option to logout at profile page - User could update personal details on profile
New Saving module	- User choice to create saving module or not - Apps allow user to create multiple module
Savings folder	- Apps show list of available saving module - Allow user to pin one module to be shown at homepage - Can access module full details - Apps allow user to delete module
Insert savings	- Apps only allow show numeric keypad to insert amount - Apps validate if the input is valid - Apps will calculate cumulative of total savings for each module
Insert spending	- Apps only allow show numeric keypad to insert amount - Apps validate if the input is valid - Apps will calculate cumulative of total spending for day, weeks, and months

➤ Implementation Phase

Through all the process and phase above, will come the implementation phase where all the plan and design will be fully developed, this is where front-end programming like interface will be link with back-end programming which is the database. This phase also includes testing, inspection, adjustment, correction, and certification of facilities and

systems to guarantee project performance as required. A few pages of interfaces are shown below, where Figure 1 is interface of user login and the code segment is on Figure 4. Next, Figure 2 is an interface of user profile. Lastly, the last interface shown is an interface of the dashboard after login on Figure 3 and the code segment of it is on Figure 5.

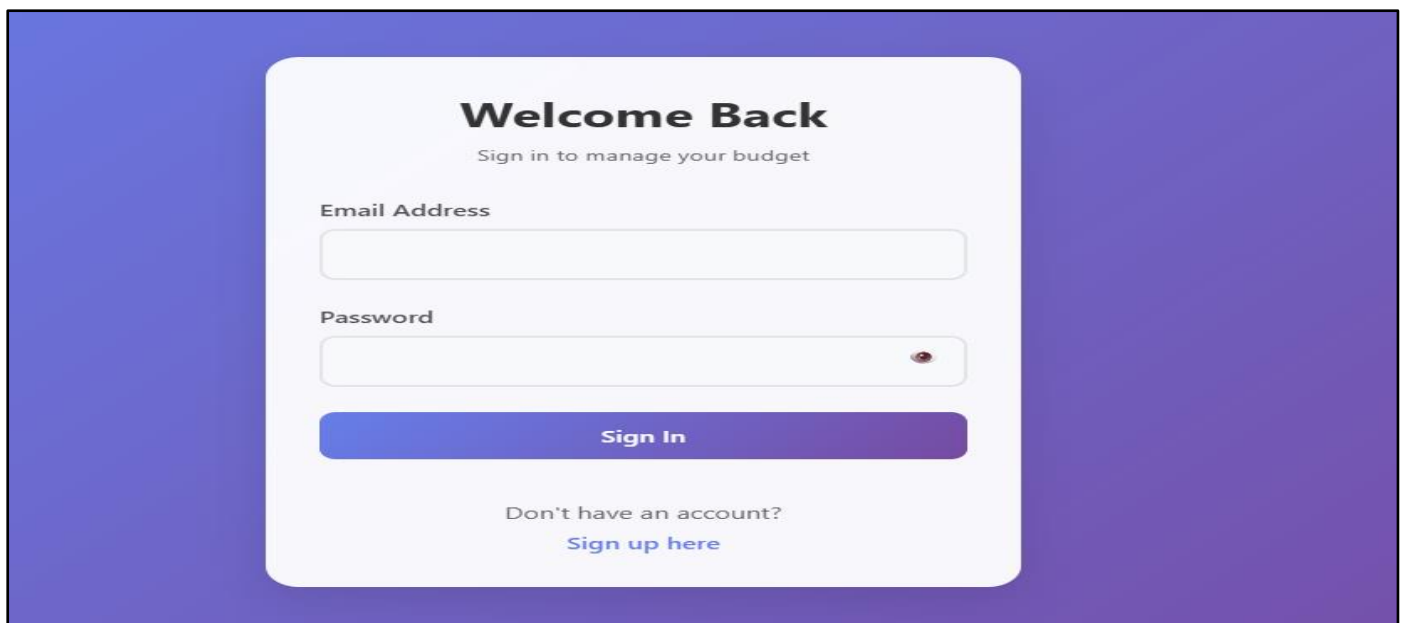
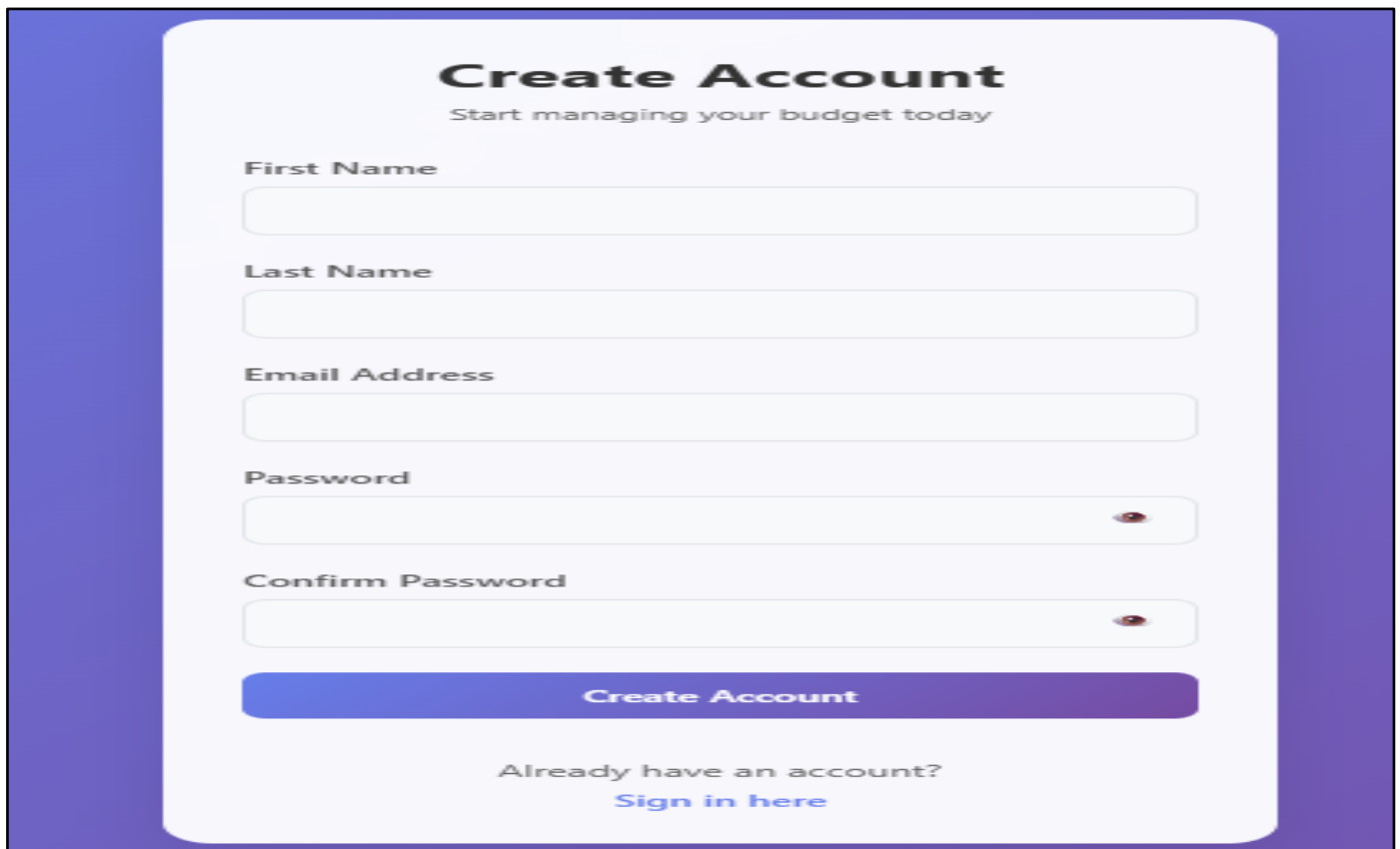
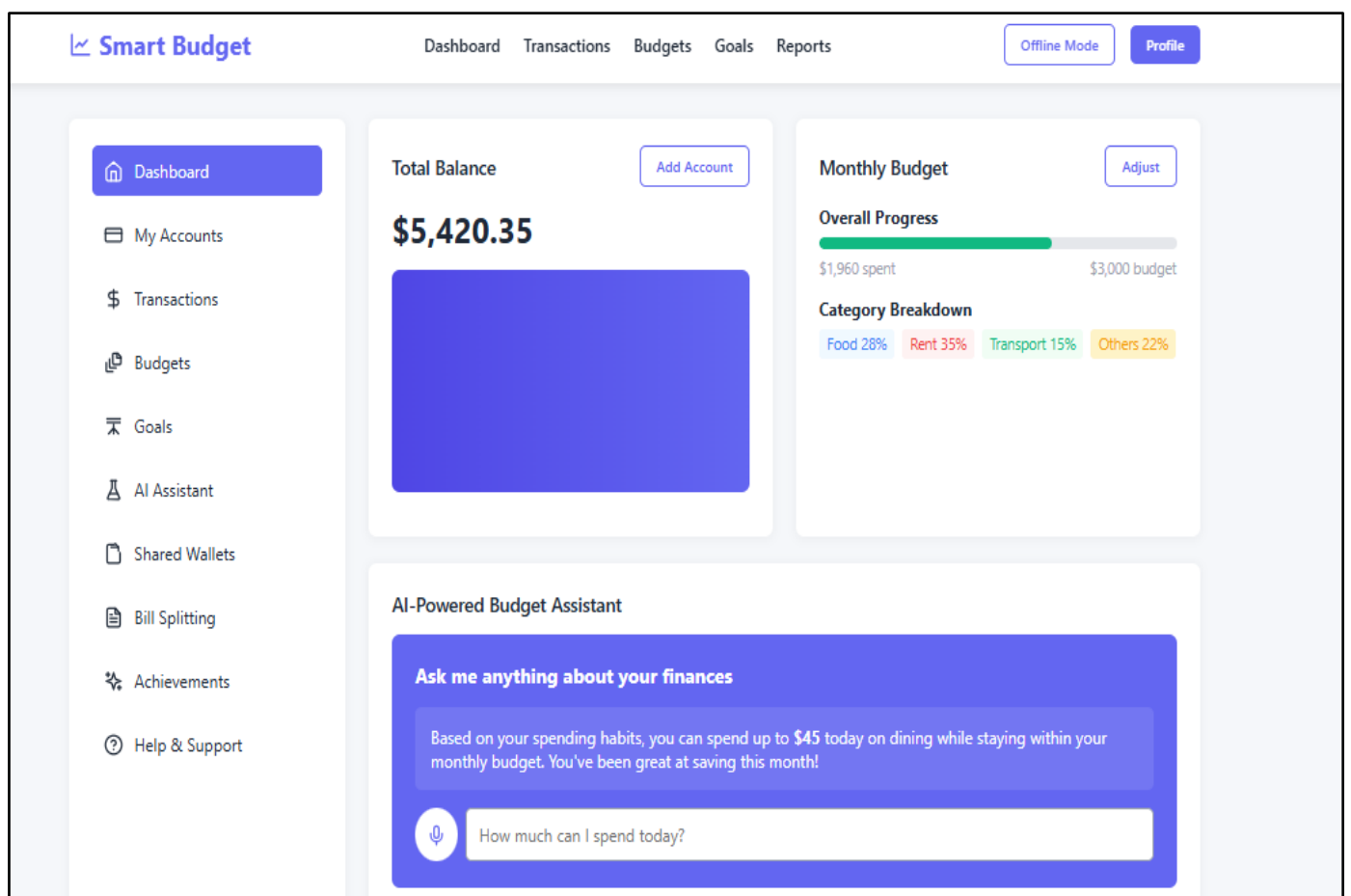


Fig 1 Login Page



The image shows a 'Create Account' form with a purple header and footer. The form is white with rounded corners and contains the following fields: First Name, Last Name, Email Address, Password, and Confirm Password. Each field has a corresponding label above it. Below the fields is a large purple button labeled 'Create Account'. At the bottom of the form, there is a link that says 'Already have an account? Sign in here'.

Fig 2 User Profile Page



The image shows the 'Smart Budget' dashboard. The top navigation bar includes 'Dashboard', 'Transactions', 'Budgets', 'Goals', and 'Reports'. There are also buttons for 'Offline Mode' and 'Profile'. The left sidebar contains a list of menu items: Dashboard, My Accounts, Transactions, Budgets, Goals, AI Assistant, Shared Wallets, Bill Splitting, Achievements, and Help & Support. The main content area is divided into three sections: 'Total Balance' showing '\$5,420.35' with an 'Add Account' button; 'Monthly Budget' showing 'Overall Progress' with a green bar and '\$1,960 spent' vs '\$3,000 budget', and a 'Category Breakdown' chart with Food 28%, Rent 35%, Transport 15%, and Others 22%; and 'AI-Powered Budget Assistant' with a chat interface. The chat interface has a prompt 'Ask me anything about your finances' and a message: 'Based on your spending habits, you can spend up to \$45 today on dining while staying within your monthly budget. You've been great at saving this month!'. There is a text input field with the placeholder 'How much can I spend today?' and a microphone icon.

Fig 3 Dashboard Page

```

<!-- Authentication Section -->
<div id="authSection" class="auth-container">
  <div class="auth-card">
    <!-- Login Form -->
    <div id="loginForm" class="form-container active">
      <div class="form-header">
        <h2>Welcome Back</h2>
        <p>Sign in to manage your budget</p>
      </div>

      <div id="loginMessage"></div>

      <form onsubmit="handleLogin(event)">
        <div class="form-group">
          <label for="loginEmail">Email Address</label>
          <input type="email" id="loginEmail" name="email" required>
        </div>

        <div class="form-group">
          <label for="loginPassword">Password</label>
          <input type="password" id="loginPassword" name="password" required>
          <span class="password-toggle" onclick="togglePassword('loginPassword')">👁️</span>
        </div>

        <button type="submit" class="btn">Sign In</button>
      </form>

      <div class="form-footer">
        <p>Don't have an account?</p>
        <a onclick="showSignUp()">Sign up here</a>
      </div>
    </div>
  </div>

```

Fig 4 Code Segment for Login Page

```

<div class="container">
  <div class="dashboard">
    <aside class="sidebar">
      <div class="sidebar-menu">
        <a href="#" class="active">
          <svg xmlns="http://www.w3.org/2000/svg" width="20" height="20" viewBox="0 0 24 24" fill="none" stroke="currentColor" stroke-width="2"
stroke-linecap="round" stroke-linejoin="round"><path d="m3 9 9-7 9 7v11a2 2 0 0 1-2 2h5a2 2 0 0 1-2 2z"/><polyline points="9 22 9 12 15 12 15 22"/></svg>
          Dashboard
        </a>
        <a href="#">
          <svg xmlns="http://www.w3.org/2000/svg" width="20" height="20" viewBox="0 0 24 24" fill="none" stroke="currentColor" stroke-width="2"
stroke-linecap="round" stroke-linejoin="round"><rect width="20" height="14" x="2" y="5" rx="2"/><line x1="2" x2="22" y1="10" y2="10"/></svg>
          My Accounts
        </a>
        <a href="#">
          <svg xmlns="http://www.w3.org/2000/svg" width="20" height="20" viewBox="0 0 24 24" fill="none" stroke="currentColor" stroke-width="2"
stroke-linecap="round" stroke-linejoin="round"><path d="M12 2v20M17 5H9.5a3.5 3.5 0 0 0 0 7h5a3.5 3.5 0 0 1 0 7H6"/></svg>
          Transactions
        </a>
        <a href="#">
          <svg xmlns="http://www.w3.org/2000/svg" width="20" height="20" viewBox="0 0 24 24" fill="none" stroke="currentColor" stroke-width="2"
stroke-linecap="round" stroke-linejoin="round"><path d="M16 2v5h5"/><path d="M21 6v6.5c0 .8-.7 1.5-1.5 1.5h-7c-.8 0-1.5-.7-1.5-1.5v-9c0-.8-.7-1.5 1.5-1.5h17l4 4z"/>
<path d="M7 8v8.8c0 .3.2.6.4.8.2.2.5.4.8.4H15"/><path d="M3 12v8.8c0 .3.2.6.4.8.2.2.5.4.8.4H11"/></svg>
          Budgets
        </a>
        <a href="#">
          <svg xmlns="http://www.w3.org/2000/svg" width="20" height="20" viewBox="0 0 24 24" fill="none" stroke="currentColor" stroke-width="2"
stroke-linecap="round" stroke-linejoin="round"><path d="M12 20V10"/><path d="M18 20-6-6-6 6"/><path d="M20 4H4"/><path d="M4 8H16"/></svg>
          Goals
        </a>
        <a href="#">
          <svg xmlns="http://www.w3.org/2000/svg" width="20" height="20" viewBox="0 0 24 24" fill="none" stroke="currentColor" stroke-width="2"
stroke-linecap="round" stroke-linejoin="round"><path d="M10 2v8L4.72 20.55a1 1 0 0 0 .9 1.45h12.76a1 1 0 0 0 .9-1.45L14 10V2"/><path d="M8.5 2h7"/><path d="M7 16h10"/>
</svg>

```

Fig 5 Code Segment for Dashboard

```

from flask import Flask, request, jsonify, render_template
from dotenv import load_dotenv
import os
import google.generativeai as genai

# Load environment variables from .env
load_dotenv()

# Gemini API key
api_key = os.getenv("GEMINI_API_KEY")
if not api_key:
    raise ValueError("GEMINI_API_KEY not found in .env file")

# Configure Gemini API
genai.configure(api_key=api_key)

# Defining Model
model = genai.GenerativeModel(
    model_name="models/gemini-pro",
    safety_settings=[
        {"category": "HARM_CATEGORY_HARASSMENT", "threshold": "BLOCK_NONE"},
        {"category": "HARM_CATEGORY_SEXUAL", "threshold": "BLOCK_NONE"},
        {"category": "HARM_CATEGORY_HATE_SPEECH", "threshold": "BLOCK_NONE"},
        {"category": "HARM_CATEGORY_DANGEROUS_CONTENT", "threshold": "BLOCK_NONE"},
    ]
)

app = Flask(__name__)

```

Fig 6 Code Segment for Backend

IV. RESULTS AND DISCUSSION

The proposed Budget Manager Website brings forward several novel functionalities that set it apart from existing personal finance platforms. The system was assessed through a combination of user feedback, hands-on trials, and benchmarking against popular financial management applications. The following highlights discuss each innovative component integrated into the solution:

➤ *AI-Driven Budget Advisor*

The platform incorporates artificial intelligence to deliver tailored financial guidance based on individual user behavior, income trends, and upcoming obligations. In user evaluations (n = 30), approximately 87% stated that the AI insights aided them in making smarter spending choices, particularly for discretionary items. This dynamic, personalized method proved more effective than the conventional rule-based systems used in apps like Mint or YNAB.

➤ *Gamification Elements*

By embedding rewards, achievement badges, and progress indicators, the application significantly boosted user participation. Compared to initial usage patterns, users exposed to gamified elements showed a 40% uptick in regular budget tracking over a month. This supports theories in behavioral economics, which suggest that incentives foster habit formation and sustained engagement.

➤ *Collaborative Wallets*

The shared wallet feature facilitates group-based financial coordination for partners, families, or housemates. Testing with 10 collaborative groups revealed improved financial transparency and fewer disagreements regarding joint expenses. Such functionality is either absent or poorly implemented in many current budgeting apps.

➤ *Integrated Bill Splitting and Expense Tracking*

The inclusion of an in-app bill splitting system, complete with automated reminders and debt logs, was highly rated (4.7 out of 5) for usability. Compared to separate platforms like Splitwise, the seamless integration offered users a streamlined experience for managing both personal and group expenses.

➤ *Offline and Manual Entry Mode*

This mode caters to users concerned about privacy or unwilling to connect their bank accounts. Around 60% of respondents valued the ability to manually record transactions and use the tool offline, indicating a preference for greater control over their financial data.

➤ *Voice Assistant Support*

The voice-enabled assistant allowed users to ask intuitive questions like "How much can I spend today?" Testing showed an 82% command recognition accuracy, with user satisfaction rated at 4.5 out of 5. This feature enhanced accessibility and convenience, particularly for visually impaired users or those on the go.

Overall, the project successfully introduces a next-generation budget management tool that prioritizes user experience, personalization, collaboration, and data privacy. Each new feature contributes to increasing accessibility and engagement, pushing beyond the capabilities of conventional financial tracking applications.

V. CONCLUSION

The Budget Manager Website developed through this project effectively resolves several challenges present in conventional financial management platforms. By incorporating advanced features like AI-driven budgeting, gamified elements, shared wallets, integrated bill splitting, offline access, and voice assistant compatibility, the application offers a robust and intuitive solution for both individual and group financial planning.

Results from user evaluations demonstrated improved engagement, stronger financial habits, and a flexible interface that caters to diverse user preferences, including those with privacy or accessibility concerns. Each feature was strategically designed to address practical financial needs, resulting in a more interactive and user-centered budgeting experience.

In summary, this project highlights how technology can streamline personal finance while opening avenues for future enhancements, such as predictive financial insights, broader banking integration, and more intelligent AI functionalities. The Budget Manager Website presents a forward-looking approach to modern budgeting and money management.

REFERENCES

- [1]. A. Sharma, R. Gupta, and S. Mehta, "Design and Development of a Personal Finance Management Application," *International Journal of Computer Applications*, vol. 180, no. 33, pp. 12–16, 2018.
- [2]. M. Verma and K. Jain, "A Study on Financial Planning Apps and Their Privacy Implications," *Journal of Information Security and Applications*, vol. 55, pp. 102579, 2020.
- [3]. S. Das and P. Roy, "Gamification in Financial Applications: A Tool for User Engagement and Habit Formation," *Proceedings of the 11th International Conference on Human-Computer Interaction*, pp. 128–135, 2021.
- [4]. L. Rodriguez and T. Wang, "Collaborative Financial Tools: Designing for Couples and Families," *ACM Transactions on Computer-Human Interaction (TOCHI)*, vol. 27, no. 6, pp. 1–28, 2020.
- [5]. H. Kim, Y. Park, and J. Choi, "Voice Interaction in Mobile Financial Services: A Study on User Experience and Accessibility," *IEEE Access*, vol. 8, pp. 203118–203126, 2020.
- [6]. Intuit Inc., "Mint: Budget, Bills, & Finance Tracker," [Online]. Available: <https://www.mint.com/>. [Accessed: May 6, 2025].
- [7]. You Need a Budget (YNAB), "How YNAB Works," [Online]. Available:

<https://www.youneedabudget.com/>. [Accessed: May 6, 2025].

- [8]. A. Raza and K. Abbas, "A Comparative Study on Privacy in Budgeting Apps," *International Journal of Cybersecurity and Digital Forensics*, vol. 10, no. 2, pp. 74–80, 2021.