Tribhuvan University

Institute Of Science and Technology

A Final Year Project Proposal

BUDGETWISE: A COMPREHENSIVE EXPENSE TRACKER WEB APPLICATION WITH BUDGET MANAGEMENT AND DATA VISUALIZATION

Submitted to

Department of Computer Science and Information Technology

Mahendra Multiple Campus Banke, Nepalgunj

In Partial fulfillment of the requirements for the Bachelor of Science in Computer Science and Information Technology

Submitted by:

Manish kumar Sharma (30354/078)

June 2025

Under the supervision of

Mr. Keshav Poudel

TABLE OF CONTENTS

1. INTRODUCTION

In today's fast-paced digital economy, personal financial management has become increasingly challenging for individuals across all demographics. Many people struggle to track their daily expenses, monitor spending patterns, and maintain control over their financial goals. Traditional methods like manual record-keeping are time-consuming and error-prone, while many existing digital solutions are either too complex for average users or lack comprehensive features for effective budget management.

The rise of digital banking and cashless transactions has made it even more difficult to keep track of spending, as money flows seamlessly through various payment methods without the tangible awareness that cash transactions provide. Research shows that individuals who actively track their expenses are 40% more likely to achieve their financial goals and maintain healthy spending habits.

BudgetWise is a comprehensive expense tracker web application that addresses these challenges by providing an intuitive, feature-rich platform for personal finance management. The application combines expense tracking, budget management, and data visualization in a single, user-friendly interface. It enables users to record transactions, categorize expenses, set monthly budgets, and gain insights through interactive charts and reports.

The application is built using modern web technologies including React.js for the frontend, Node.js and Express.js for the backend, and MongoDB for data storage. It incorporates advanced features like secure user authentication, real-time data visualization, export capabilities, and responsive design to ensure accessibility across all devices.

BudgetWise caters to a diverse user base including students managing limited budgets, working professionals tracking business expenses, families planning household finances, and anyone seeking better control over their financial life. The application's intuitive design makes it accessible to users with varying levels of technical expertise while providing powerful features for detailed financial analysis.

2. PROBLEM STATEMENT

Personal financial management remains a significant challenge for individuals worldwide, with studies indicating that over 60% of people do not actively track their expenses or maintain a budget.

This lack of financial awareness often leads to overspending, debt accumulation, and failure to achieve financial goals.

Many existing expense tracking solutions suffer from critical limitations that prevent widespread adoption and sustained usage. A major issue is complexity - many financial apps are designed with advanced features that overwhelm casual users, leading to abandonment rates exceeding 80% within the first three months of usage. Users often find these applications too complicated for simple expense tracking needs.

Another significant problem is the lack of comprehensive budget management features. While some apps excel at expense recording, they fail to provide meaningful budget planning, goal setting, or spending limit monitoring. Users end up using multiple separate tools for different aspects of financial management, creating fragmentation and reducing overall effectiveness.

Data visualization and reporting capabilities are often inadequate in existing solutions. Users struggle to understand their spending patterns without clear, intuitive charts and graphs. Many apps provide basic lists of transactions but fail to offer meaningful insights into spending trends, category-wise breakdowns, or progress toward financial goals.

Accessibility and device compatibility present additional challenges. Many expense tracking applications are either mobile-only or desktop-only, forcing users to compromise on convenience or functionality. The lack of seamless synchronization across devices often results in incomplete or inconsistent financial records.

Furthermore, data export and portability features are frequently missing or limited in existing solutions. Users cannot easily extract their financial data for tax purposes, accountant consultations, or migration to other platforms, creating vendor lock-in situations that discourage adoption.

There is a clear need for a comprehensive, user-friendly expense tracking solution that combines ease of use with powerful features, supports multiple devices, and provides meaningful financial insights. BudgetWise addresses these gaps by offering an integrated platform that simplifies expense tracking while providing advanced budget management and data visualization capabilities.

3. OBJECTIVE

3.1 Primary Objective

The primary objective of this project is to develop a comprehensive full-stack expense tracker web application called "BudgetWise" that provides users with an intuitive and powerful platform for personal financial management, combining expense tracking, budget planning, and data visualization to help users achieve better financial control and make informed spending decisions.

3.2 Specific Objectives

• Develop a robust expense management system that allows users to easily add, edit, and delete expense entries with comprehensive details including amount, category, date, and description

• Implement a flexible categorization system that enables users to organize expenses into meaningful groups such as Food, Transportation, Bills, Entertainment, and custom categories for personalized financial tracking

• Create a comprehensive budget management feature that allows users to set monthly spending limits for different categories and track progress toward budget goals with visual indicators and alerts

• Design an interactive dashboard with dynamic charts and graphs using Chart.js or Recharts to provide users with clear insights into their spending patterns, trends, and financial health

• Develop a secure user authentication system using JWT tokens to ensure data privacy and enable personalized financial tracking across user sessions

4. SCOPE AND LIMITATION

4.1 Scope

Functional Scope:

The BudgetWise application will provide comprehensive expense tracking and budget management capabilities to help users maintain control over their personal finances. The expense management system will handle the complete lifecycle of financial transactions, allowing users to add new expenses with detailed information including amount, title, category, date, and optional notes. Users will be able to edit existing entries to correct mistakes or update information, and delete transactions they no longer need to track.

The categorization system will support both predefined categories such as Food, Transportation, Bills, Entertainment, Healthcare, and Shopping, as well as custom categories that users can create based

on their specific needs. This flexibility ensures that users can organize their expenses in ways that make sense for their lifestyle and financial goals.

The budget management feature will enable users to set monthly spending limits for individual categories or overall spending. The system will track progress toward these budget goals and provide visual indicators showing how much of the budget has been used and how much remains. Users will receive warnings when they approach or exceed their budget limits, helping them maintain financial discipline.

The dashboard will feature interactive charts and graphs that visualize spending data in multiple formats including pie charts for category distribution, bar charts for monthly comparisons, and line graphs for spending trends over time. These visualizations will help users quickly understand their financial patterns and identify areas for improvement.

The expense history feature will provide a comprehensive view of all transactions with advanced filtering options. Users can filter expenses by date range, category, amount range, or search for specific transactions using keywords. Sorting options will allow users to organize their data by date, amount, or category for easier analysis.

Technical Scope:

The BudgetWise application will be developed as a full-stack web platform using the MERN (MongoDB, Express.js, React.js, Node.js) technology stack. The frontend will be built with React.js, providing a dynamic, responsive, and interactive user interface. Tailwind CSS will be used for styling to ensure consistent design and rapid development.

The backend will utilize Node.js with Express.js to handle server-side logic, API endpoints, and business logic. MongoDB will serve as the database solution, storing user accounts, expense records, budget settings, and category information. Mongoose will be used as the Object Data Modeling library for MongoDB integration.

Authentication will be implemented using JSON Web Tokens (JWT) to ensure secure user sessions and data protection. Password security will be maintained using bcrypt for hashing and encryption. The system will include middleware for authentication verification and route protection.

Data visualization will be implemented using Chart.js or Recharts library to create interactive and responsive charts and graphs. The export functionality will support CSV and PDF formats, allowing users to download their financial data for external use.

The application will follow responsive web design principles, ensuring optimal performance and usability across desktop computers, tablets, and mobile devices. API design will follow RESTful principles for clean, maintainable, and scalable backend architecture.

4.2 Limitations

Technical Limitations:

This application is designed as a web-based platform accessible through modern web browsers and does not include native mobile applications for Android or iOS platforms. Users will need an active internet connection to access all features, as the application does not support offline functionality for expense entry or data synchronization.

Integration Limitations:

BudgetWise operates as a standalone financial management system and does not integrate with external banking APIs, credit card systems, or other financial institutions for automatic transaction import. Users must manually enter all expense data, which may require additional effort compared to automated solutions.

The application does not include integration with popular accounting software, tax preparation tools, or business expense management systems. Email notifications and alerts are not implemented in the current scope, requiring users to actively check the application for budget status and financial updates.

Scalability Limitations:

While the application can handle multiple concurrent users, performance optimization for extremely large datasets (thousands of transactions per user) may require additional database indexing and query optimization. Real-time collaborative features for family or shared budget management are not included in the current implementation.

Data Limitations:

The system does not include advanced financial analysis features such as investment tracking, loan calculations, or complex financial forecasting. Multi-currency support is not implemented, limiting the application to single-currency expense tracking.

5. METHODOLOGY

5.1 Requirement Identification

Study of Existing Systems:

To identify best practices and common challenges in expense tracking applications, a comprehensive analysis of popular financial management tools will be conducted. This includes studying applications like Mint, YNAB (You Need A Budget), PocketGuard, and Expensify to understand their feature sets, user interface design patterns, and user engagement strategies.

The analysis will focus on identifying strengths such as intuitive expense entry processes, effective budget visualization techniques, and user-friendly categorization systems. Common weaknesses will also be documented, including complex navigation structures, overwhelming feature sets, and limited customization options that often lead to user abandonment.

Mobile-first financial apps like Toshl, Money Lover, and Spendee will be examined to understand how they balance simplicity with functionality. Desktop applications like Quicken and GnuCash will be studied to identify advanced features that power users require for comprehensive financial management.

User reviews, app store ratings, and financial technology blogs will be analyzed to understand user pain points, feature requests, and satisfaction factors. This research will inform the design decisions for BudgetWise to ensure it addresses real user needs while avoiding common pitfalls.

Requirement Collection:

Requirements will be gathered through multiple methodologies including online surveys targeting potential users across different demographics and income levels. Focus group discussions will be conducted with students, working professionals, and families to understand their specific expense tracking needs and budget management challenges.

Analysis of personal finance forums, Reddit communities, and social media discussions will provide insights into common financial management struggles and desired features. User personas will be developed representing different user types including budget-conscious students, busy professionals, and family financial managers.

Both functional requirements (expense entry, budget setting, data visualization) and non-functional requirements (performance, security, usability) will be documented comprehensively. Requirements will be prioritized based on user impact and development complexity to guide the implementation roadmap.

5.2 Feasibility Study

Technical Feasibility:

The chosen MERN technology stack is well-suited for developing the BudgetWise application. React.js provides excellent capabilities for building interactive user interfaces with component reusability and state management. The extensive ecosystem of React libraries supports advanced features like data visualization and responsive design.

Node.js and Express.js offer robust backend development capabilities with excellent performance for handling API requests and business logic. The JavaScript-based full-stack approach ensures consistency and reduces context switching during development. MongoDB's document-based structure is ideal for storing flexible expense data with varying attributes.

The technology stack is well-documented with extensive community support, reducing development risks and ensuring reliable problem-solving resources. All chosen technologies are mature, actively maintained, and suitable for production deployment.

Operational Feasibility:

The project addresses genuine user needs in personal financial management, with a large potential user base seeking better expense tracking solutions. The intuitive interface design and comprehensive feature set provide clear value propositions for user adoption and retention.

The web-based approach ensures broad accessibility without requiring app store approvals or platform-specific development. Cloud hosting solutions enable scalable deployment with automatic scaling based on user demand.

Economic Feasibility:

Development costs are minimized through the use of open-source technologies and frameworks that do not require licensing fees. Cloud hosting services provide cost-effective infrastructure scaling based on actual usage. The initial investment remains low while providing opportunities for future monetization through premium features or subscription models.

5.3 Tools

Analysis and Design Tools:

The project will utilize modern design and planning tools to ensure systematic development and clear documentation. Figma will serve as the primary tool for user interface design, wireframing, and prototyping, enabling collaborative design processes and user feedback collection.

For technical documentation and system architecture diagrams, Draw.io will be used to create clear visualizations of system components, data flow, and user interaction patterns. Database design will be documented using Entity-Relationship diagrams to ensure proper data structure and relationships.

Requirements documentation will follow structured templates that clearly define functional specifications, acceptance criteria, and testing requirements. User stories will be documented to maintain focus on user value throughout the development process.

Implementation Tools:

Frontend Development: • React: Core JavaScript library for building dynamic and interactive user interfaces with component-based architecture • Tailwind CSS: Utility-first CSS framework enabling rapid UI development with consistent styling • React Router: Client-side routing solution for single-page application navigation • Axios: HTTP client library for efficient API communication and data fetching • Chart.js/Recharts: Data visualization library for creating interactive charts and graphs

Backend Development: • Node.js: JavaScript runtime environment for server-side development with excellent performance • Express.js: Web application framework providing robust API development capabilities and middleware support • MongoDB: NoSQL document database for flexible and scalable data storage • Mongoose: Object Data Modeling library for MongoDB with schema validation and query building

Authentication and Security: • JSON Web Tokens (JWT): Token-based authentication system for secure user sessions • bcrypt: Password hashing library ensuring secure credential storage • dotenv: Environment variable management for secure configuration

Development and Deployment Tools: • Visual Studio Code: Primary integrated development environment with extensive plugin ecosystem • Git: Version control system for code management, collaboration, and deployment • npm: Package manager for JavaScript dependencies and build script management • Postman: API testing and documentation tool for backend endpoint validation and testing

Additional Libraries: • date-fns/moment.js: Date manipulation and formatting library for transaction date handling • react-csv: CSV export functionality for financial data download • jsPDF: PDF generation library for formatted expense reports

6. HIGH LEVEL DESIGN OF BUDGETWISE

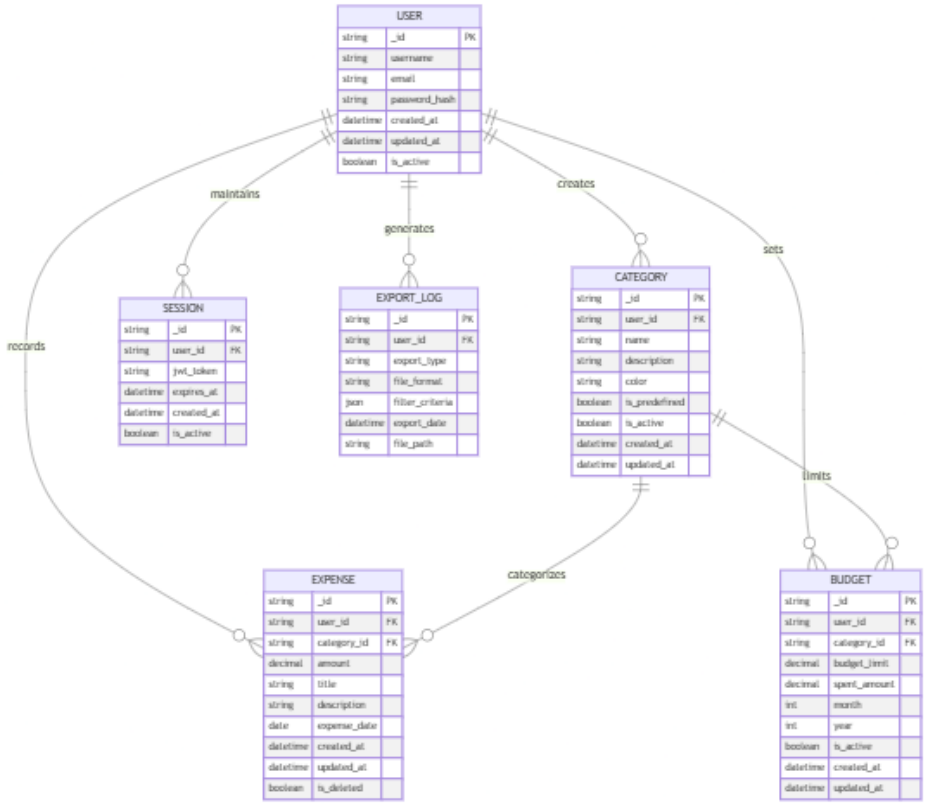
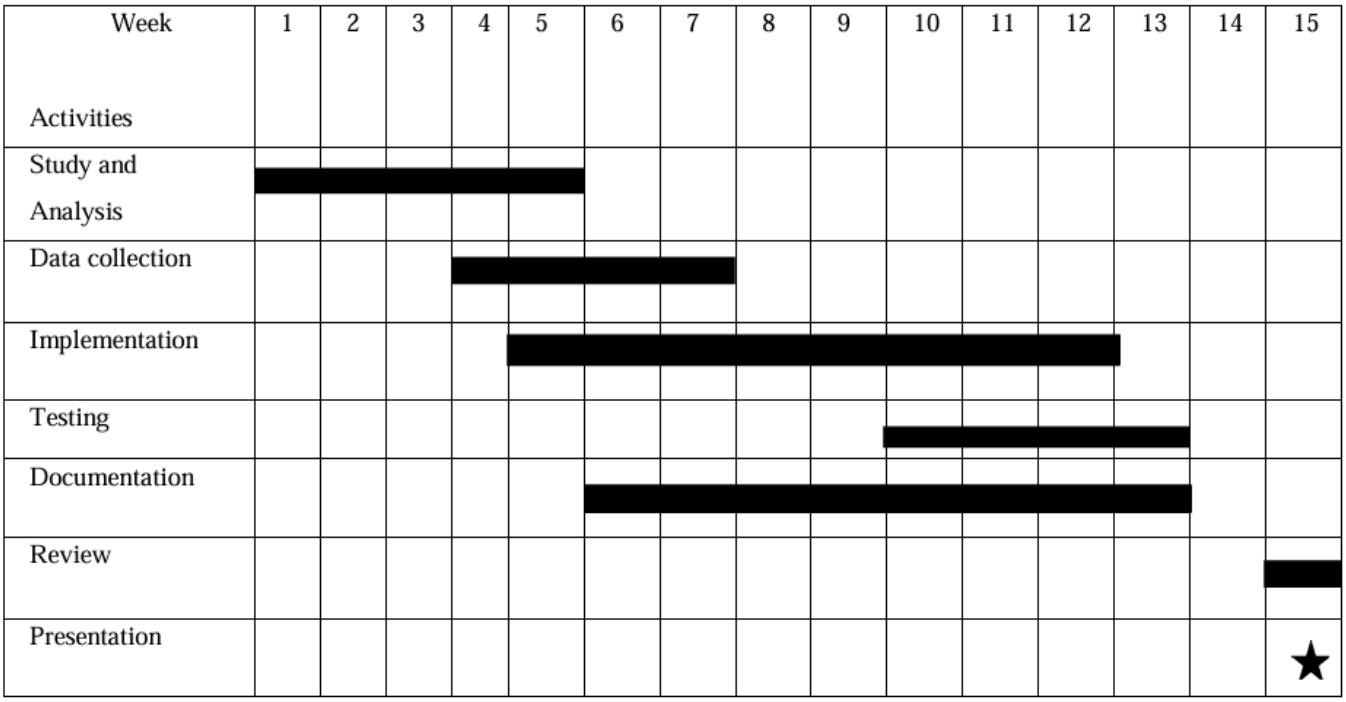


Figure: ER-Diagram

7. GANTT CHART TO SHOW THE PROJECTED TIME PLANNING



8. EXPECTED OUTCOME

8.1 Primary Deliverables

Upon successful completion of the BudgetWise project, the primary deliverable will be a fully functional, production-ready expense tracker web application that provides comprehensive personal financial management capabilities. The application will demonstrate professional-grade user interface design, robust backend architecture, and scalable deployment suitable for real-world usage by individuals and families.

The system will support secure user registration and authentication, allowing multiple users to maintain separate financial profiles with complete data privacy. Users will experience intuitive expense entry processes, flexible budget management, and insightful data visualization that promotes better financial decision-making.

The application will include comprehensive expense tracking capabilities with support for multiple categories, detailed transaction records, and advanced filtering options. Budget management features will enable users to set realistic financial goals and monitor progress through visual indicators and alerts.

Interactive dashboards will provide clear insights into spending patterns through various chart types including pie charts for category distribution, bar charts for monthly comparisons, and trend analysis for long-term financial planning. Export capabilities will allow users to download their financial data in industry-standard formats for external analysis or record-keeping.

The responsive design will ensure optimal functionality across desktop computers, tablets, and mobile devices, providing users with convenient access to their financial information regardless of their preferred device or location.

References:

1. React: Official React Documentation, Meta Platforms, Inc., https://reactjs.org/
2. Express: Fast, unopinionated, minimalist web framework for Node.js, OpenJS Foundation, https://expressjs.com/
3. MongoDB Inc.: The Developer Data Platform, https://www.mongodb.com/
4. Mongoose: Elegant MongoDB object modeling for Node.js, Automattic Inc., https://mongoosejs.com/
5. Chart.js: Simple yet flexible JavaScript charting for designers & developers, https://www.chartjs.org/
6. Tailwind CSS: A utility-first CSS framework, Tailwind Labs Inc., https://tailwindcss.com/