WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Paisa Expense Tracker

Amogh Shetty M[1], Badami Ethihas Setty[1], Mr.Kiran Kumar Mn[2]

[1]UG Scholar, Department Of Computer Applications, BMS College Of Commerce And Management, India
[2] HOD, Department Of Computer Applications, BMS College Of Commerce And Management, India

A B S T R A C T

The Paisa Expense Tracker is a desktop application designed to streamline personal finance management using the MERN stack—MongoDB, Express.js, React.js, and Node.js. It provides users with a secure platform to record, categorize, and analyse financial transactions efficiently. Users can manage their expenses via a dynamic dashboard, filter transactions using multiple criteria, and download reports in structured formats like CSV or PDF for offline use. The application features default categories for essential classifications while allowing users to create, edit, and delete custom categories for personalized tracking. Its responsive React.js-powered interface ensures a seamless user experience, while Express.js and Node.js handle server-side operations efficiently. MongoDB offers flexible storage and management of financial data. Beyond simplifying financial tracking, the Paisa Expense Tracker demonstrates the integration of modern web technologies in building secure, scalable, and user-centric applications. By combining interactivity, flexibility, and security, it helps users gain better control over their income and expenses while leveraging advanced development tools for optimal financial management

Keywords:: Personal Finance Management, Expense Tracking, MERN Stack, MongoDB, Express.js, React.js, Node.js

INTRODUCTION

The Effective personal finance management is crucial for maintaining financial stability and ensuring long-term security. In an era where financial transactions have become increasingly digital, many individuals struggle with tracking their income and expenses efficiently. Poor financial management can lead to unplanned expenditures, debt accumulation, and a lack of financial awareness, making it essential to adopt structured methods for monitoring and controlling personal finances. Recent studies underscore the significance of digital tools in financial tracking, highlighting their role in fostering responsible spending habits and improving budgeting strategies. By leveraging technology, individuals can gain clearer insights into their financial behavior, reduce unnecessary expenses, and work towards financial independence. The Paisa Expense Tracker is a modern web application designed to simplify financial management through automation and intuitive features. Built using the MERN (MongoDB, Express.js, React, and Node.js) stack, it provides users with a secure platform to log, categorize, and analyze their financial transactions. With real-time reports, budget alerts, and cloud-based accessibility, the application empowers users to make informed financial decisions and develop disciplined spending habits. By integrating cutting-edge technology into personal finance, the Paisa Expense Tracker serves as an essential tool for achieving financial awareness and long-term economic stability.

LITERATURE REVIEW

A. Challenges in Expense Tracking

- Managing personal finances is crucial for financial stability, but many struggle to track income and expenses effectively.
- Existing applications range from manual entry to automated tools with analytics and cloud storage.
- Many solutions lack customization, data integrity, and advanced filtering, limiting their effectiveness.

B. Existing Expense Tracking Systems

- Traditional bookkeeping and spreadsheets require effort and are prone to errors.
- Automated apps like Mint, YNAB, and Pocket Guard depend on bank integration, which may not suit users preferring manual entry.
- Research highlights the importance of real-time insights, secure storage, and customization in financial software.

C. MERN Stack in Web Development

- The MERN stack (MongoDB, Express.js, React.js, Node.js) offers scalability and responsiveness for financial applications.
- MongoDB's NoSQL architecture efficiently stores financial data.

- Express.js and Node.js handle server-side tasks and authentication.
- React.js improves user experience with a dynamic interface.

D. Features of Paisa Expense Tracker

- Provides advanced filtering, automated categorization, and report downloads.
- Users can create, edit, and delete custom categories while default categories remain unchanged.
- The dashboard displays transactions in a table format with filtering by category, type, and date range.
- Reports can be downloaded in CSV or PDF for offline tracking.
- Front-end and back-end validation ensure data accuracy and integrity.

METHOLOGY

A. Technology Stack Selection

The Paisa Expense Tracker is built using the MERN stack for scalability and real-time data processing:

- MongoDB – NoSQL database for efficient storage of transactions and user data.
- Express.js – Handles server-side functions and API routing.
- React.js – Provides a dynamic and user-friendly interface.
- Node.js – Enables smooth interaction between the front-end and back-end.

The application uses a client-server model, where the front-end interacts with the back-end via RESTful APIs, consisting of:

- Front-End (React.js) – Manages user interactions and displays financial data.
- Back-End (Express.js & Node.js) – Handles authentication, data processing, and API requests.
- Database (MongoDB) – Securely stores user transactions and categories.

B. User Authentication and Security

- JWT-based authentication – Secure access through token-based login.
- Password encryption – Hashing techniques protect user credentials.
- Input validation – Ensures only structured and valid data is stored.

C. Transaction Management and Categorization

- Transaction Form – Users input details like amount, category, type, and date.
- Category Management – Users can create, modify, or delete categories while default ones remain unchanged.
- Filtering Options – Transactions can be filtered by category, type, and date range.

D. Data Storage and Retrieval

- Indexing & query optimization – Speeds up transaction retrieval.
- Data validation – Prevents duplicate or incorrect entries.

E. Report Generation and Exporting

For offline financial tracking:

- CSV & PDF export – Users can download structured transaction reports.
- Formatted data presentation – Ensures clarity in financial audits.

F. Performance Evaluation and Testing

The system undergoes rigorous testing:

- Unit Testing – Validates individual components.
- Integration Testing – Ensures seamless interaction between front-end, back-end, and database.
- User Testing – Assesses usability and identifies improvements.

SOFTWARE USED

- **Operating System:** Windows 10/11, macOS (10.15 or above), or any Linux distribution.
- **Node.js**: Version 16.x or higher
- **MongoDB**: Local MongoDB server
- **Code Editor**: Visual Studio Code (preferred) or any compatible IDE.
- **Package Managers**: npm (bundled with Node.js)
- **Database:** Mongo DB Server is selected for its ease of maintenance, robust querying capabilities using simple English queries, and reliable data retrieval.

RESULTS

The development and implementation of Paisa Expense Tracker have provided valuable insights into financial management efficiency, user experience, and system performance. The results obtained through testing highlight the following key aspects:

- Expense Tracking Accuracy: Users recorded income and expenses with high accuracy using predefined categories for structured financial analysis.
- Budget Control Effectiveness: The budget monitoring system successfully sent alerts when expenditures exceeded predefined limits, promoting financial discipline.
- Data Visualization Insights: Graphical reports, including pie charts and bar graphs, improved financial assessment, allowing users to analyse spending patterns efficiently.
- System Performance: The application maintained fast response times, with API requests processing within 200 to 300 milliseconds, ensuring a smooth user experience.
- Security and Accessibility: JWT-based authentication secured user data, while cloud storage enabled access from multiple devices, ensuring flexibility and usability.

User feedback from initial testing demonstrated that the system efficiently addressed the challenges of manual financial tracking by offering a modern, automated, and structured approach to expense management.

CONCLUSION

The Paisa Expense Tracker is a modern financial management solution that effectively addresses the challenges of expense tracking, budgeting, and data visualization. Built using the MERN stack, the system ensures seamless financial monitoring through structured logging, automated alerts, and graphical insights. The application enhances expense tracking accuracy, offering users an intuitive way to categorize and analyse transactions. Its budget control system promotes financial discipline, while real-time visualizations improve spending pattern analysis. The system also demonstrates high performance, handling transactions efficiently while ensuring secure authentication and cloud-based accessibility. Comparative analysis confirms the superiority of Paisa Expense Tracker over traditional financial tracking methods, providing automation, scalability, and universal accessibility. Despite its effectiveness, future improvements such as AI-driven insights, automated expense categorization, and bank API integrations will further refine its capabilities and enhance user experience. By leveraging modern web technologies, the Paisa Expense Tracker serves as a reliable and scalable financial management tool, empowering users to make informed financial decisions and develop better spending habits. The application bridges the gap between traditional budgeting techniques and advanced digital solutions, marking a significant step toward effective financial management in the digital era.

.References

1. MongoDB Documentation.(n.d).MongoDB for developers Retrieved from https://www.mongodb.com/docs/manual/
2. Express.js Documentation.(n.d.). *Express.js Guide*. Retrieved from https://expressjs.com/
3. React.js Documentation. (n.d.). *React.js – A JavaScript Library for Building User Interfaces*. Retrieved fromhttps://react.dev/
4. Node.js Documentation. (n.d.). *Node.js API Documentation*. Retrieved from https://nodejs.org/en/docs/