

Walletry(Expense Tracker AI)

Hritik Roushan

Department of Computer Science and Engineering
Galgotias University
Greater Noida, India
hritik.22scse1010386@galgotiasuniversity.edu.in

Abhimanyu Kumar

Department of Computer Science and Engineering
Galgotias University
Greater Noida, India
abhimanyu.22scse1010385@galgotiasuniversity.edu.in

Jyoti Ratna

Department of Computer Science and Engineering
Galgotias University
Greater Noida, India
jyoti.ratna@galgotiasuniversity.edu.in

Abstract—The high rate of the development of the digital payment services, internet stores, and online subscriptions has complicated the process of managing personal finances. Individuals are daily undertaking numerous transactions on the various platforms and therefore, the manual nature of tracking the expenses is slow, which may easily generate errors and is difficult to maintain. Such old-fashioned data management tools as handwritten notes and spreadsheets fail to update automatically, are not viewable in real time, and fail to arrange information, thus they are less useful with increased number of transactions. They are also highly dependent on the discipline of the user and may be inconsistent and hence not easy to plan and analyze in the long run.

To solve these issues, numerous online cost management applications have been developed. Nevertheless, the majority of them revolve around the addition of numerous functionalities like analytics and automation and disregard proper architecture. Consequently, they are difficult to maintain, do not scale and have parts that are tightly bound together and are more dependent on external services. This renders them unreliable and difficult to enhance in future. Moreover, not all mobile-only and AI-based systems address the question of privacy, rely on a single platform and introduce additional complexity to operation.

In this paper, the author presents a contemporary web-based expense tracking application called Walletry, developed using the latest technologies of a full stack, focusing on the clean architecture with a deep sense of modularity and type safety. The system relies on a layered model, which divides user interface, authentication, business logic and data storage that makes it stronger and less troublesome to maintain. Sensitive financial data can be safely protected by secure authentication and strict control over sessions. Database interactions are type-safe, which means that the data is more reliable and will lessen the chances of errors during runtime. Unlike platform-specific or AI-intensive trackers, Walletry does not deal much with any particular architecture, but rather with fast performance and any platform. It has been tested and found that the system is very stable, user-friendly, and can expand with time, which is a strong foundation of personal expense management and further smart capabilities.

Index Terms—Expense Tracking, Personal Finance Management, Next.js, TypeScript, Prisma ORM, PostgreSQL, Web Application Architecture.

I. INTRODUCTION

Due to the popularity of online payments, online shopping platforms, and subscriptions, personal financial management has become a necessity of the digital era. E-wallets and instant payment services and recurring bills have enhanced the number of transactions individuals make in a day. These numerous transactions are difficult to process manually that results in missing logs, inaccurate budgets, and financial unawareness. Trying to spend money in multiple platforms also leaves the users unable to have a single view of their money making it hard to make good choices.

The old fashioned ways of keeping track of expenses such as handwritten notes or spreadsheets remain popular as they are easy and simple to follow. However, they are problematic: they are not automated and need manual input, their data is not always consistent, and can be easily misused by a human being. These techniques become slow and slow to scale as the number of transactions increases. They also fail to assist in real time tracking, trending and long term planning. In the absence of clear validation and automated processes, they cannot be trusted to continue to manage personal finances.

Digital expense trackers have come into being to correct these issues. Majority of them have features like categorization of expenses, dashboards and summaries that enhance the user interface. The recent research considers the implementation of AI and machine learning to automatically categorize the expenses and provide individual insights. Despite the automation offered by such systems, they are commonly associated with a complex architecture, increased cost and reliance on third-party services. These dependencies can be detrimental towards reliability, privacy and long term maintenance.

A number of studies have demonstrated that excessive number of features created simultaneously and monolithic design are disadvantageous to the maintenance and upgrading of applications as they grow. The performance is slowed, testing is more difficult and the possibility of adding new features is restricted. Tight coupling also implies that new features involve further significant refactoring that increases the cost of development. As such, there is an increasing

demand to have expense trackers that do not compromise on a multitude of features and, instead, have good architecture.

In this paper, the author introduces Walletry, an expense tracker, written in web technology, designed with an emphasis on clear architecture, modularity and type safety. Walletry strives to provide a reliable, scalable, and maintainable platform to personal finance, protect the data, operate on any platform, and grow in the future by using the modern full-stack technology and the layered design.

II. LITERATURE REVIEW

There are numerous researches that investigate how to monitor expenditure and financial control. One of the major priorities is AI assistants that categorize expenses and predict further expenditure, depending on the actions taken by the user. Those assistants apply machine learning and linguistic technologies to analyze transaction data and assist the user by reducing the manual labor. They are more convenient and help people to be more conscious about their money, however, they require a great number of outside services and dense models. This complicates the system, increases the cost of running the system and depends on the services of other companies. Adhering data to the system continuously can also be detrimental to speed and expansion in the case of many users.

The other field of work is that of online billing and invoice software applied primarily by businesses. They work with paper bills, store invoices, compare them and forecast the costs in the future. They are business-friendly, though not user friendly by individuals who require fast and convenient logging. The cumbersome, paper-intensive nature of the work is an unnecessary complication in the day to day tracking of money. They also take bills in standard forms, which is not the way on how individual people use their money.

There were certain scientists who created mobile apps which categorize costs automatically by use of text alerts, transactional emails or bank messages. The automation in these phone apps is good in the ecosystem of the phone. They however rely on permissions which are phone-specific. This restricts the usage in other phones and generates privacy concerns as the application continues to check on personal messages and financial alerts. Moreover, due to changes made by phone companies to their rules, it may impact the effectiveness of app functionality and its sustainability.

Personal financial products have been developed in versions that can be accessed through a browser in web format and can therefore be moved everywhere and used. A lot of these web tools are built on a single mega architecture which does not distinctly isolate components and thus becomes difficult to maintain, test or expand as time goes by. According to software research, web apps should be built to be reliable and maintainable through modular and layered architecture and safe data handling. Systems that lack these attributes tend to have a higher level of technical debt because they tend to incorporate new functionality.

More recent research has demonstrated that the strength of their architecture may have a bigger influence on the duration of lifespan of a system, and the ease with which additional features may be added to it, than the number of features it contains. Clean architecture and modular architecture allow applications to be developed in small steps and reduced rewrites. Walletry follows this idea. It is aimed at decoupling components, secure designs and is scalable, such that additional smart features can be added without significant alterations and does not compromise the reliability of the system

III. SYSTEM OVERVIEW

Walletry is a complete spending tracking web tool. It allows the registered users to document, administer and review their own finances securely and conveniently. The site provides a single point to store financial data and assists one by showing how he or she spends at various points in time. With a simple and user friendly layout, Walletry will reduce the amount of mental and manual labor required to track on daily basis and promote frequent usage.

The design is heavily concerned with maintaining data consistency, privacy and an increase in over time. It applies layers that plainly divide what the user sees, program rules and the database. It is easier to maintain the system, read the code and add things to the system later without having to break the already working system with this separation. The modular method also allows the developers to upgrade a section without making changes to the others.

Walletry serves a large number of users simultaneously but the data of a user remains isolated. Privacy, integrity, and reliability are ensured with every user having an individual space in which their records are kept in safe locations and can only be read when they are logged in. It is platform-independent as the system can be accessed to any device via a browser. Other features such as analysis tools or graphs can be introduced in the future after minimal modification in the structure.

IV. SYSTEM ARCHITECTURE

The design of Walletry is based on a layered model which enhances modularity, maintenance and stability in general. The principal layers are the Frontend, Authentication, Application Logic, Service and utility, Data access and Database. Each layer has a single job to perform, and communicates with the other layers by definite rules, maintaining control of communication and responsibilities.

The Frontend controls the interaction of the user and display of the moving content through a responsive web interface, thus, making navigation a breeze and real-time updates. Authentication layer verifies the user passwords and controls the session prior to any other sensitive work so as to guard the unauthorized access to the user information.

Application Logic is the text that employs fundamental spending operations and encompasses regulations such as approval of costs, categorization, and data integrity. Service

and utility layer offers reusable elements, utilities and checks that reduce redundancy codes and enhance maintenance. The Data Access layer sends data to the database safely avoiding any cut offs and maintains data consistency.

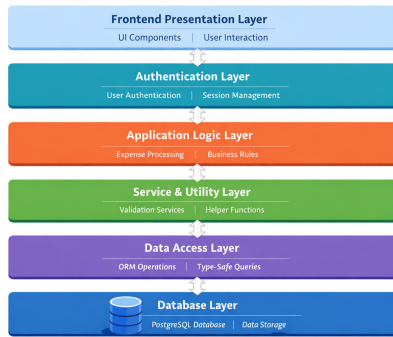


Fig. 1. Layered system architecture of the Walletry expense tracking application

The Database layer is used to store the user profiles and the spending records so as to be constantly available, ensuring that data is not lost. The storage can be kept independent of the business rules, making the system flexible and able to upgrade the database in the future. This multi-layered structure allows individual development and testing, and makes debugging easier and efficient extension of the app as more users and requirements emerge.

V. TOOLS AND TECHNOLOGIES USED

Walletry was developed using the best web tools to ensure that it is fast, reliable and easily maintainable. Next.js and TypeScript are used in the front-end, which allows pages to be loaded in a short period of time, as well as to ensure security of the code.

We apply Tailwind CSS to get a responsive appearance that remains the same when it is used on phones, tablets, and desktops, as well as get a faster design process.

On the server side is a type-safe tool of database access Prisma. It simplifies query writing and minimizes errors in the code. Information is saved in PostgreSQL, which is an effective database, which maintains information dependable and expands with the utilization. Clerk is in charge of user login and authentication and session management and provides secure access and session maintenance. All these tools are selected in such a way that the system is able to expand and new features can be easily added.

VI. RESULTS AND DISCUSSION

We tested Walletry on the speed, ease of use and system building. It was well integrated with the user having to log in, add expenses, retrieve data, and view the dashboard. The server-side rendering and good routing were used to achieve the smooth navigation and efficient server-side rendering. The

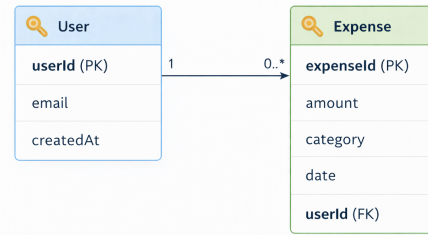


Fig. 4. Database - schema illustrating user and expense entity relationships in Walletry

Fig. 2. Database schema illustrating user and expense entity relationships in Walletry

site was also responsive, indicating good session and back-end management, when multiple users were using the site simultaneously.

Walletry has a user-friendly interface. It simplifies day-to-day recording of expenses and reduces cognition. It reduces errors, eliminates needless typing, and maintains uniformity of data as compared to spreadsheets. Financial records can be maintained in an organized manner without complicated formulas and manual verification by the users.

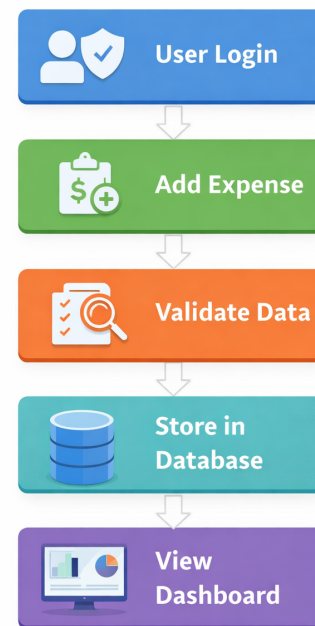


Fig. 3. Workflow of expense recording and retrieval process in Walletry

Other trackers of expenses with numerous functions or basing on AI tend to be unruly and closely connected to external services. Walletry maintains a clean manageable design which is less complex. It remains concise and interactive in the sense that it can be easy to debug, test and include new functions in the future.

The tiered design will allow every part to develop independently as a growth in need occurs. These findings demonstrate that an effective architecture and the separation of functions are the key to the construction of scalable, reliable and user-friendly expense managers. Comprehensively, the Walletry is usable, fast, and well-designed and can be used in long-term personal finance.

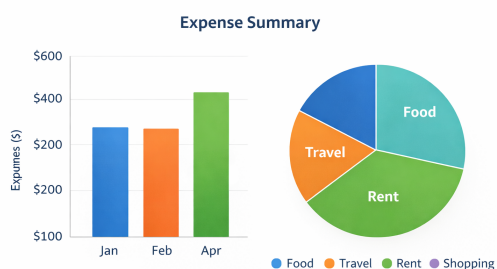


Fig. 4. Sample expense summary visualization generated by the system

VII. CONCLUSION

We launched the new, web-based, fully-type, and full-modular expense-tracker Walletry. It addresses the issue of growing pains of personal finance in a digital age where the user is transacting through various platforms.

The research demonstrates that reliability and maintenance can be enhanced with the help of good architecture and the current web tools. The hierarchical structure manages the flow of data, eliminates complexity and allows components to be modified without the entire system being compromised. Increased consistency and minimized run time errors make the system more resilient as type safe database use and structured data manipulation ensure that the database is consistent and free of errors at run-time.

To a user, Walletry makes it easy to enter daily expenses and also offers a central point to handle money. It reduces work, improves precision, and is more accessible in comparison with spreadsheets. It is compatible with any platform unlike AI-heavy solutions and mobile-only solutions and does not rely excessively on external services, which fosters trust and reliability in the long term.

The findings have shown that design is one of the most important elements in the development of a scalable personal finance application. Rather than over-automating, Walletry establishes a firm foundation that can acquire new capabilities such as analytics or insights without making minor redesigns. Additions to new functions can be made one after another.

Overall, Walletry is a reliable and scalable platform of personal expense tracking and contributes to the body of research on web-based finance tools. The conclusions state the need to design to be modular, type-safe, as well as safe in the use of data in designing future-proof software. The system satisfies the existing requirements and provides a powerful foundation of the future research and development of intelligent and scalable personal finance.

REFERENCES

- [1] A. Sharma and R. Mehta, "Personal finance management systems: A review," *International Journal of Computer Science and Information Technology (IJCSIT)*, 2019.
- [2] V. Gupta and A. Jain, "Web-based financial management systems," *International Journal of Information Systems (IIS)*, 2020.
- [3] P. Kumar and N. Singh, "Digital expense tracking platforms," *International Journal of Engineering Research and Technology (IJERT)*, 2021.
- [4] S. Pawar et al., "ExpenseXpert: AI-driven expense tracking," *Indian Journal of Computer Science and Technology (IJCSIT)*, 2024.
- [5] S. Raihan et al., "AI-powered personal expense tracking," in *Proc. International Conference on Quantum Photonics, Artificial Intelligence, and Networking (QPAIN)*, 2025.
- [6] M. Fowler, *Patterns of Enterprise Application Architecture*. Addison-Wesley, 2002.
- [7] E. Johri et al., "Expense management system," in *Proc. IEEE Global Conference for Advancement in Technology (GCAT)*, 2023.
- [8] M. O. Adebisi et al., "Voice-based expense tracking," in *Proc. International Conference on Science, Engineering and Business for Driving Sustainable Development Goals (SEB4SDG)*, 2024.
- [9] S. Newman, *Building Scalable Web Applications*. O'Reilly Media, 2019.
- [10] G. Raj et al., "Digital billing systems," *International Journal of Engineering Research and Technology (IJERT)*, 2021.
- [11] R. Thakare et al., "Expense tracker using Naive Bayes," *International Journal of Recent Engineering Science (IJRES)*, 2023.
- [12] S. Chandini et al., "Online income and expense tracker," *International Research Journal of Engineering and Technology (IRJET)*, 2019.
- [13] K. Nale et al., "AI-powered budget tracker," *International Journal of Scientific Research and Technology (IJSRT)*, 2025.
- [14] P. Bhatale et al., "TrackEZ expense tracker," in *Proc. International Conference on Engineering Technologies (ICET)*, 2023.
- [15] W. Wang et al., "AI finance tracker," in *Proc. International Conference on Computer Science and Engineering (ICSE)*, 2024.
- [16] R. Pressman, *Software Engineering: A Practitioner's Approach*. McGraw-Hill, 2014.
- [17] L. Bass, P. Clements, and R. Kazman, *Software Architecture in Practice*. Addison-Wesley, 2013.
- [18] M. Richards, *Software Architecture Patterns*. O'Reilly Media, 2015.
- [19] Prisma, "Prisma ORM documentation," 2024. [Online]. Available: <https://www.prisma.io>
- [20] Vercel, "Next.js documentation," 2024. [Online]. Available: <https://nextjs.org>