



TRIBHUVAN UNIVERSITY
FACULTY OF HUMANITIES AND SOCIAL SCIENCE

Project Proposal

on

“KharchaTrack : Smart Expenses Tracker with Insights”

Submitted to

Department of Computer Application
National College of Computer Studies

In partial fulfilment of the requirements
of

Bachelor's Degree in Computer Application

Submitted By:

Reman Buddhacharya
(NCCSBCA099 / 2081)

Table of Content

1. Introduction.....	1
2. Problem Statement.....	2
3. Objective	3
4. Methodology	3
4.1. Requirement Identification	3
4.1.1. Study of Existing Systems	4
4.1.2. Literature Review.....	5
4.1.3. Requirement Analysis.....	6
4.2. Feasibility Study	7
4.2.1. Technical Feasibility	8
4.2.2. Operational Feasibility.....	8
4.2.3. Economic Feasibility	8
4.3. High Level System Design	9
4.3.1. Methodology of the purposed system	9
4.3.2. Use Case Diagram.....	10
4.3.3. System Flowchart.....	11
4.3.4. Working Mechanism of Proposed System.....	12
4.3.5. Description of Algorithm.....	13
5. Gantt Chart.....	15
6. Expected Outcome	15
7. References.....	16

Table of Content

Figure 1: Waterfall Model	9
Figure 2: Use Case Diagram with Online User and Admin.....	10
Figure 3: Flowchart for Admin/User Activities	11
Figure 4: Gantt Chart	15

1. Introduction

Effective financial management is essential for both individuals and businesses. However, tracking expenses manually or using basic expense trackers often lacks deeper insights, making it difficult to forecast future spending or detect fraudulent transactions. This highlights the need for a smarter, data-driven solution.

KharchaTrack is a smart expense tracker designed to address these challenges. It will provide automated expense tracking, forecasting, and fraud detection using machine learning. Unlike conventional expense trackers, KharchaTrack will analyze spending patterns to predict future expenses and identify unusual transactions that may indicate fraud.

This project will be developed using **Laravel, Livewire, and PHP-ML**, ensuring a scalable, interactive, and efficient financial management system. The combination of these technologies will allow for real-time data processing, intuitive user interactions, and advanced predictive analytics.

This proposal outlines the development of **KharchaTrack**, including its objectives, methodology, expected outcomes, and implementation plan. The project aims to create a comprehensive expense management tool that not only records transactions but also helps users make informed financial decisions with AI-powered insights.

2. Problem Statement

In today's fast-paced world, managing personal finances has become increasingly challenging. Many individuals struggle with tracking their expenses, identifying spending patterns, and avoiding unnecessary financial strain. Traditional methods, such as manual logs or basic spreadsheets, lack real-time insights and predictive capabilities [5].

Moreover, fraudulent transactions and unusual spending behaviors often go unnoticed, leading to financial losses. Existing expense trackers mostly focus on recording transactions without providing intelligent forecasting or anomaly detection to help users make better financial decisions.

This project aims to develop a Smart Expense Tracker using Laravel, Livewire, and PHP-ML that integrates Moving Average for expense forecasting and Isolation Forest for anomaly detection. This system will not only allow users to log and categorize expenses but also provide future expense predictions and alert users to potentially suspicious transactions.

By leveraging machine learning techniques, this solution will empower users with data-driven financial insights, helping them make informed decisions, detect anomalies, and improve their spending habits.

3. Objective

The main objective of KharchaTrack is:

- To predict future expenses using the Moving Average Algorithm.
- To detect unusual spending patterns using Isolation Forest.
- To provide users with data-driven insights to manage expenses efficiently

4. Methodology

This section outlines the measures and procedures necessary to achieve the objectives of KharchaTrack : Smart Expenses Tracker with Insights. It details the research design, data collection methods, tools, techniques, and the analytical approach employed to ensure an efficient, accurate, and timely project execution.

4.1. Requirement Identification

Managing personal finances has become increasingly difficult, with traditional methods like manual logs or spreadsheets lacking real-time insights and predictive capabilities. Existing expense trackers often fail to provide intelligent forecasting, fraud detection, or user-friendly interfaces.

KharchaTrack aims to address these issues by offering a smart, easy-to-use platform for tracking expenses. It will provide expense forecasting using the Moving Average algorithm, and detect unusual spending patterns through Isolation Forest for fraud detection. Built with Laravel, Livewire, and PHP-ML, KharchaTrack will empower users to manage their finances more efficiently and securely.

4.1.1. Study of Existing Systems

The Various expense tracking tools and applications exist in the market, with many focusing on basic transaction logging and budgeting features. Popular tools such as Mint, YNAB (You Need A Budget), and Expensify provide users with functionalities like expense categorization, budgeting, and reporting. However, these tools primarily focus on manual data entry and lack predictive features or fraud detection.

1. **Mint:** A widely used expense tracker offering budgeting, financial goal tracking, and automatic categorization of transactions. While Mint provides some financial insights, it lacks advanced features like expense forecasting and fraud detection through machine learning. Additionally, it is often criticized for issues with data privacy and the syncing of bank transactions.
2. **YNAB (You Need A Budget):** Known for its detailed budgeting approach, YNAB helps users allocate funds to categories and track their spending. However, it does not provide automatic expense forecasting or fraud detection, and it requires active manual management, which may not be ideal for users seeking automation.
3. **Expensify:** A tool mainly used for managing business expenses, offering features like receipt scanning and reporting. It's more business-focused and doesn't provide personal budgeting or forecasting capabilities based on historical data, and it lacks advanced anomaly detection.

While these applications provide valuable financial tracking tools, KharchaTrack aims to improve upon them by integrating moving average forecasting and anomaly detection using Isolation Forest. Unlike existing solutions, KharchaTrack will empower users not only to track and categorize their expenses but also to predict future expenses and detect fraudulent spending behaviors in real-time. This combination of features makes KharchaTrack a more advanced and user-centric financial tool.

4.1.2. Literature Review

Expense tracking and personal finance management tools have gained significant attention due to the increasing complexity of managing finances in today's fast-paced world. Traditional methods, such as spreadsheets and manual record-keeping, are increasingly seen as inefficient and prone to human error. As a result, automated solutions like Mint, YNAB, and Expensify have emerged, offering features like expense categorization, budgeting, and financial reporting. However, these tools primarily focus on static tracking without advanced capabilities such as forecasting or fraud detection.

One significant challenge in expense tracking is accurately predicting future spending and detecting fraudulent activities. Machine learning algorithms have shown potential in addressing these challenges. For example, Moving Average and Linear Regression are commonly used in time-series forecasting to predict future expenses based on past spending patterns [2]. Similarly, Isolation Forest has proven to be an effective anomaly detection algorithm, identifying unusual spending behaviors that might indicate fraud or errors [1]. Existing systems typically lack advanced features like machine learning-based forecasting or anomaly detection, which are vital for a comprehensive and intelligent expense management system. Research in machine learning and statistical analysis has demonstrated the effectiveness of these algorithms in improving financial predictions and providing better user insights.

The integration of machine learning algorithms like Moving Average for forecasting and Isolation Forest for fraud detection in a user-friendly web application can provide significant improvements over existing solutions. By offering these features, KharchaTrack aims to deliver a more robust and efficient tool for personal finance management, improving the accuracy of financial predictions and enhancing fraud detection.

4.1.3. Requirement Analysis

Requirements will be collected through personal evaluation of different existing systems, along with suggestions from mentors, classmates, and supervisors.

4.1.2.1. Functional Requirements

1 User Account Management

- Users can create an account and log in with unique credentials.
- Password recovery is available if users forget their credentials.

2 Expense Management

- Users can add, view, update, and delete expense entries, with timestamps for each transaction.
- Expenses can be categorized automatically or manually.

3 Expense Forecasting & Anomaly Detection

- The system predicts future expenses using the **Moving Average** algorithm.
- **Isolation Forest** detects and flags unusual or potentially fraudulent transactions.

4 Expense Update History

- Users can view and revert to previous edits/state of their expense entries.

5 Dashboard & Notifications

- A personalized dashboard displays expense summaries, forecasts, and anomalies.
- Users are notified about forecasted expenses and flagged transactions.

6 Export & Compatibility

- Users can export expense data to CSV or PDF formats.
- The system is compatible with modern web browsers.

4.1.2.2. Non-Functional Requirements

1. Performance

- The application should provide real-time updates for expense tracking and forecasting with minimal latency.

2. Scalability

- The system should be able to handle a growing number of users and increasing expense data without performance issues.

3. Usability

- The platform should have an intuitive and user-friendly interface to make navigation and expense management easy.
- The dashboard should display clear insights and be simple to use.

4. Availability

- The system should be available 24/7 with minimal downtime for maintenance or updates.
- Regular backups of user data should be taken to ensure data integrity.

5. Compatibility

- The system should be compatible with all modern web browsers (e.g., Chrome, Firefox, Edge).
- It should be responsive and function well on both desktop and mobile devices.

6. Maintainability

- The application should have clean, modular code that is easy to update and maintain over time.

4.2. Feasibility Study

The system is evaluated for future development with a set of constraints. The feasibility study is done regarding the available technologies, time constraints, area of application, cost of deployment and upkeep, and future possibilities of the project.

4.2.1. Technical Feasibility

KharchaTrack will be developed using stable, open-source technologies. The frontend will use HTML, CSS, and JavaScript, with Livewire for real-time updates. The backend will be powered by Laravel, ensuring a secure and efficient architecture [4]. MySQL will store user data securely. PHP-ML will handle machine learning algorithms like Moving Average and Isolation Forest for forecasting and anomaly detection [3]. The system will be web-based and compatible with modern browsers and mobile devices, ensuring accessibility from anywhere. Development will be done using free software like Visual Studio Code and GitHub.

4.2.2. Operational Feasibility

The user interface of KharchaTrack will be designed to be intuitive and user-friendly, ensuring ease of use for individuals of all technical backgrounds. Similar to existing expense tracking applications, the interface will feature simple navigation and interactive elements for a smooth user experience. The platform will be responsive, ensuring compatibility across various devices, including desktops, tablets, and mobile phones. By following basic design principles, the application will be easy to understand, enabling users to quickly adopt and effectively manage their expenses.

4.2.3. Economic Feasibility

Most of the software used for developing this project will be open source and free. Along with it, suitable cloud hosting will be used. Most of the technology will be free.

4.3. High Level System Design

This section shows high level system design, behaviour, and interaction with external entities.

4.3.1. Methodology of the purposed system

This project is on a small scale and has well-defined requirements and a linear approach. Under such development circumstances, the simplest development model, the waterfall model, is applicable. The waterfall model produces a set of documents after each stage, as well as a time frame that is enough to implement the project under the methodology.

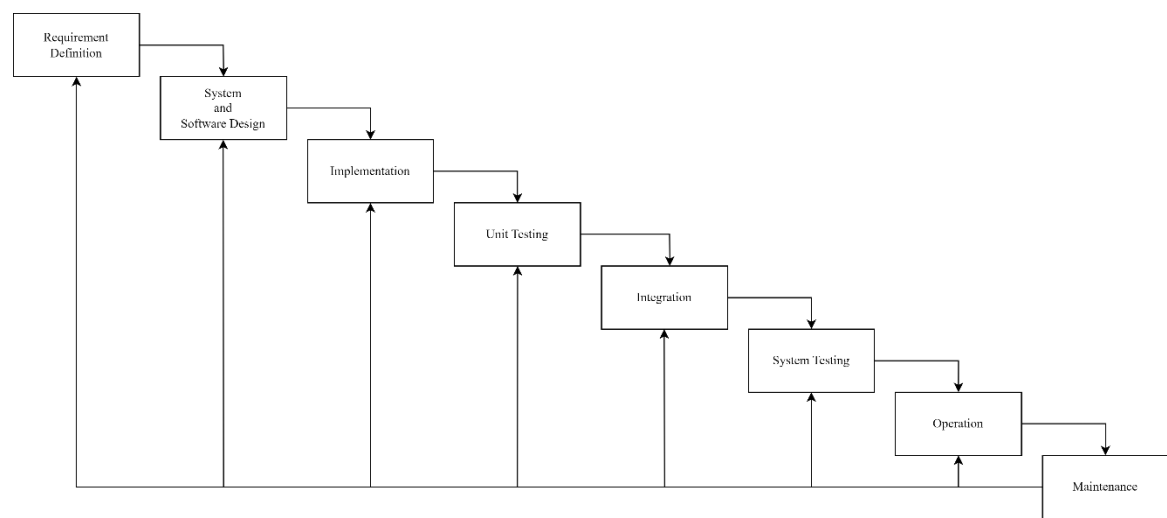


Figure 1: Waterfall Model

4.3.2. Use Case Diagram



Figure 2: Use Case Diagram with Online User and Admin.

4.3.3. System Flowchart

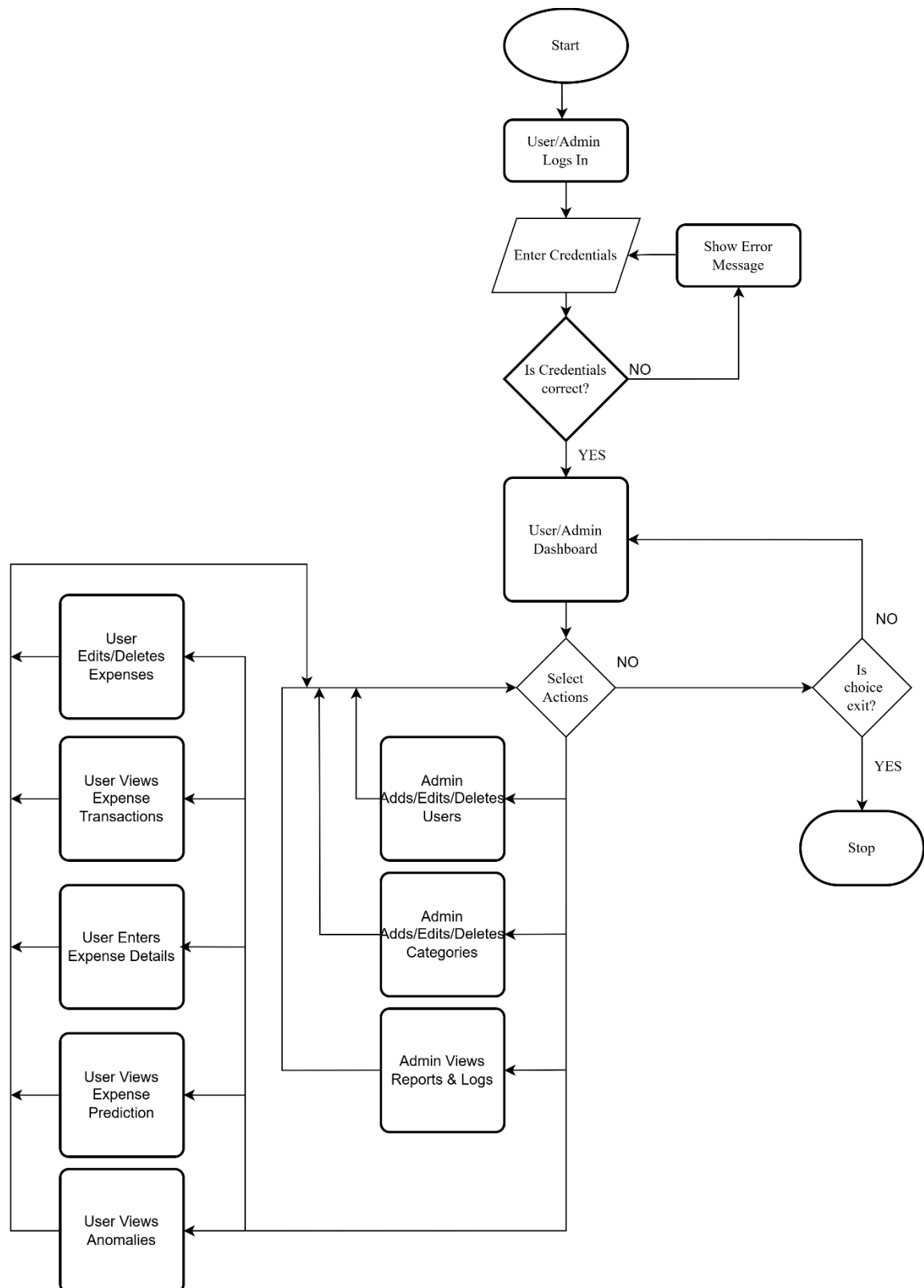


Figure 3: Flowchart for Admin/User Activity

4.3.4. Working Mechanism of Proposed System

A description of the working mechanism of KharchaTrack:

1. **Registration:** Users can create a KharchaTrack account by providing basic information such as their username, email address, and password.
2. **User Login:** Registered users can access their accounts by entering their credentials (username and password).
3. **Expense Management:** Users can add, view, update, and delete expense entries. Each entry includes details such as amount, category, and date.
4. **Expense Categorization:** Expenses are automatically or manually categorized to help users track spending patterns.
5. **Expense Forecasting:** The Moving Average algorithm predicts future expenses based on past spending behavior.
6. **Anomaly Detection:** Isolation Forest algorithm detects unusual spending patterns, flagging potential fraudulent or unexpected expenses.
7. **Edit History** Users can track changes made to their expenses over time and revert to previous state when necessary.
8. **Dashboard & Insights:** Users have access to a personalized dashboard displaying expenses, forecasts, and anomalies, offering clear financial insights.
9. **Notifications:** Users are notified about forecasted expenses and flagged anomalies to keep them informed.
10. **Data Export:** Users can export their expense data in CSV or PDF formats for easy sharing and analysis.
11. **Password Recovery:** If users forget their login credentials, they can securely recover their password through a recovery process.
12. **Admin Control:** Admins can manage user accounts, monitor system activity, and generate reports to maintain system integrity.

4.3.5. Description of Algorithm

The system for KharchaTrack employs two key algorithms for its functionalities: Moving Average for expense forecasting and Isolation Forest for anomaly detection.

1. **Moving Average Algorithm (for Expense Forecasting):**

The **Moving Average** (MA) algorithm is used to predict future expenses based on past spending behavior [2]. The MA method smooths out fluctuations in the data to provide a trend line that represents average expenditure over a set period. This helps users forecast their future expenses, enabling better budget management.

Steps:

- **Data Collection:** The algorithm collects the user's expense data over a specified time period.
- **Calculation of Average:** It calculates the average of expenses over the given period (e.g., weekly, monthly).
- **Prediction:** The algorithm then uses the average to forecast future expenses, making the assumption that spending patterns will remain relatively consistent.

2. **Isolation Forest Algorithm (for Anomaly Detection):**

The **Isolation Forest** algorithm is used to detect outliers or unusual expense patterns, which could indicate fraudulent activity or errors in data entry [1]. This algorithm isolates anomalies by recursively partitioning the data into smaller subsets, ensuring that anomalies are more likely to be isolated in fewer partitions compared to normal data.

Steps:

- **Data Splitting:** The algorithm divides the expense data into partitions, progressively isolating subsets of data.
- **Scoring:** It assigns a score to each transaction based on how isolated it is from the rest of the data. Transactions that are isolated early on are flagged as anomalies.
- **Detection:** Anomalous or unusual transactions are identified and flagged for review. These transactions might indicate unexpected spending or potential fraud.

In summary, The **Moving Average** algorithm helps forecast future expenses by calculating the average of past spending patterns, providing users with useful insights for budgeting. Meanwhile, the **Isolation Forest** algorithm detects outliers or anomalies, ensuring that any unexpected or potentially fraudulent expenses are flagged for further investigation. These algorithms work together to provide accurate forecasting and enhance the security of the expense tracking process in **KharchaTrack**.

5. Gantt Chart

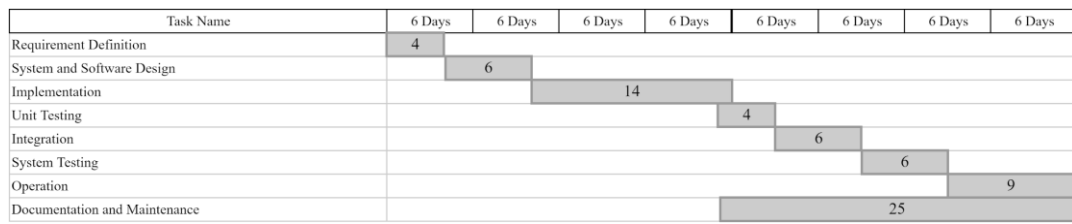


Figure 4: Gantt Chart

6. Expected Outcome

The expected outcome of this project is:

- A fully functional, user-friendly web application for tracking expenses with forecasting and anomaly detection.
- The system will efficiently handle a large volume of user expense data without performance issues.
- Users will have access to features like automated expense categorization, spending forecasts, and anomaly detection to flag unusual transactions.
- The system will address limitations in existing expense trackers by integrating predictive analytics and fraud detection.
- Regular updates and testing will ensure reliability, accuracy, and ease of use.

7. References

- [1] F. T. Liu, K. M. Ting, and Z. H. Zhou, "Isolation Forest: A Robust Method for Anomaly Detection," in *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 10, pp. 2356-2370, Oct. 2013.
- [2] J. D. Hamilton, "Time Series Analysis and Forecasting using Moving Average Models," *Journal of Business & Economic Statistics*, vol. 16, no. 3, pp. 253-264, Jul. 1998.
- [3] M. Razzaq et al., "PHP-ML: Machine Learning Library for PHP Developers," in *Proceedings of the International Conference on Computational Intelligence and Computing*, Hyderabad, India, 2019, pp. 1-6.
- [4] T. Otwell, "Laravel: The PHP Framework for Web Artisans," *Laravel Documentation*, 2024. [Online]. Available: <https://laravel.com/docs>. [Accessed: Mar. 5, 2025].
- [5] A. Khare, R. Gupta, and S. Malik, "A Comprehensive Survey of Expense Tracking Applications: Challenges and Opportunities in Personal Financial Management," *International Journal of Financial Technology*, vol. 12, no. 2, pp. 45-62, Jun. 2022.