Expense Tracker: Empowering Users with Smart Budgeting and Financial Insights

Tanuja Bedre, Hardik Bhagat, Isha Bhamare and Kunal Kapse
Guide: - Prof. Shital Nalgirkar
Department of Computer Engineering (CS) ISBM College of Engineering, Pune-412115, Maharashtra, India

Abstract: Strong fiscal management is very important in today's digital economy, where keeping track of and analyzing expenses manually can be tedious and error-prone. In this paper, the development and enhancement of an intelligent expense monitor with the use of the MERN stack (MongoDB, explicit.js, React.js, Node.js), ML for prediction of future revenue, Optical person recognition (OCR) for programmatic invoice information extraction, and Cloud integration for enhanced scalability and accessibility are provided.

This app allows consumers to easily capture, classify, and process their spends. The OCR feature, accompanying AI-based text reputation, captures important information (date, amount, service provider, description) from uploaded receipts, eliminating guide records access. The ML model, trained on historical spending habits, forecasts impending charges, aiding customers make informed financial decisions. Cloud-based completely deployment, utilizing AWS, F cozy records storing, real-time synchronization across multiple devices, and improved gadget scalability.

The backend, developed using Node.js and express.js, enables high-performance API processing and data manipulation, while MongoDB is a feature-rich NoSQL database for error-free expense tracking. The frontend, developed over React.js, provides a live, user-friendly, and adaptable consumer interface. Experimental effects suggest that the device greatly minimizes guide workload, enhances financial focus, and provides actionable insights to enable better budgeting and control of expenditures.

Through the combination of MERN stack, ML-based completely predictive analytics, OCR-driven automation, and cloud services, this intelligent cost tracker provides a comfortable, scalable, and intelligent economic management solution for modern users.

Keywords: Expense Tracker, MERN Stack, Machine Learning, OCR, Cloud Integration, Expense Prediction, Financial Management, Revenue.

INTRODUCTION

Private and enterprise expenses are a critical financial balance issue. However, traditional methods of recording expenses, including manually entering transactions into spreadsheets or using paper receipts, are ineffective and subject to errors. With the advance of era, computerized expense tracking software has become a method of simplifying monetary management.

This paper presents a MERN stack-based expense Tracker that incorporates device mastering (ML) for forecasting prices, Optical individual recognition (OCR) for automated bill facts extraction, and Cloud-primarily based storage for real-time access and syncing. The device aims to provide an shrewd and mechanized platform for customers to regulate their finances properly, taking the advantage of AI-driven analytics and a consumer-pleasant interface.

The application allows users to include receipts, which are automatically scanned using OCR to fetch significant financial details. furthermore, past spending habits are examined by ML algorithms to make fee forecasts, helping users maximize their budgeting.

The integration in the cloud ensures that financial information is stored securely and is available across multiple devices, enhancing convenience and scalability.

PROBLEM STATEMENT

Traditional cost tracking methods are usually inconvenient, imprecise, and time-consuming. Numerous customers don't report their daily charges due to the hassle of guide availability, leading to poor monetary consciousness and inefficient budgeting. additionally, handling receipts in paper format exposes

them to loss or damage, rendering monetary monitoring even more challenging.

Furthermore, current cost monitoring software is devoid of predictive analytics to assist customers in anticipating destiny spending patterns and maximizing their economic making preparations. without the assistance of AI-based insights, customers struggle to investigate their past expenditures and increase improved spending behavior. further, data synchronization across multiple devices proves unreliable in offline or tool-based solutions, limiting accessibility and usability.

In order to deal with those challenges, this research suggests a smart fee tracker, which incorporates automation of cost recording through OCR, forecasting destiny charges the usage of ML, and ensuring stress-free, cloud-based data storage for real-time access.

IMPORTANCE OF THE IMPLEMENTATION

The application of this innovative expense tracker has full-size importance within the domain of personal and business economic management. some of the most important blessings include:

• Automation of price entering – The combination of OCR era eliminates manual information access with the help of extracting price information from invoices and receipts.

• AI-pushed economic Insights – ML-primarily based predictions help customers anticipate future fees and plan their budgets more effectively.

• Cloud-based Synchronization – The integration into the cloud guarantees that customers have right of access to their economic information from any gadget in real-time, enhancing comfort and security.

• Error reduction & enhanced Accuracy – Automated extraction of statistics and predictive analysis minimize human errors, ensuring more accurate economic monitoring.

• Scalability & Accessibility – With the MERN stack, one can have a scalable and over-performance internet software that is suitable for every man or woman and enterprise usage cases.

OBJECTIVES & CONTRIBUTIONS

The key objectives of these studies and deployment are:

1. Create a full-stack cost tracking app the application of the MERN stack (MongoDB, specific.js, React.js, Node.js) to a green and scalable person experience.
2. Implement Optical person reputation (OCR) for automatically extracting fee details from receipts and invoices.
3. Merge ML algorithms to analyze beyond prices and are anticipating future spending trends.
4. Permit cloud-based garage and synchronization to ensure facts security, accessibility, and real-time updates across devices.
5. Facilitate higher consumer engagement and monetary focus by providing insightful visualizations and expenditure tendencies via an interactive dashboard.

Contributions

This paper presents the following original contributions:

• An automated OCR-based completely charge monitoring device that drastically minimizes manual attempt in logging transactions.

• A forecast price prediction version the usage of ML algorithms, which facilitates forward-looking monetary selection-making.

• A scalable, cloud-integrated framework that ensures facts security, accessibility, and synchronizing throughout greater than one devices.

• A contemporary-day, responsive, and user-friendly internet utility developed using the MERN stack, providing a seamless financial control experience.

Through the combination of OCR, ML, and cloud development, this implementation aims to disrupt the way customers monitor and manage their finances, creating a more efficient, accurate, and predictive method of cost control.

RELATED WORK

A number of rate tracking software and money control devices were developed over the years to assist people and businesses in controlling their expenditures. traditional packages such as Mint, Expensify, and PocketGuard provide clients with manual and semi-automatic methods to record transactions, allocate expenses, and create money reports.

The recent developments in Synthetic Intelligence (AI), Optical character popularity (OCR), and cloud computing have created the blending of automated

receipt scanning, predictive analytics, and multi-device synchronization within economic packages. yet, despite these improvements, most current solutions remain to fall short in delivering real-time financial insights, computerized facts extraction, and smart forecasting earlier studies has investigated various methods for price monitoring:

• Guide information entry Rate Trackers – Most traditional programs need clients to manually enter costs, i.e., time-consuming and subject to human errors.

• OCR price tracking – some advanced packages, including Expensify, utilize OCR technology to read textual information from receipts, but they tend to falter with unstructured data and need guide corrections.

• ML-driven monetary Forecasting – AI-driven fiscal assistants have been created to analyze spending habits, but they're often not part of a standalone, integrated program that integrates cost monitoring, OCR-based automation, and predictive modeling.

• Cloud expense management – services such as QuickBooks and FreshBooks provide cloud storage and cross-device access but are primarily set up for enterprise accounting rather than personal cost tracking.

DISCUSSION OF PRIOR WORK AND HOW OUR IMPLEMENTATION DIFFERS

Simultaneously, there are available expense trackers that are useful economic control equipment, yet all of them limited automation, no predictive analytics, and do not integrate more than one advanced technologies in a single platform.

Our implementation varies in the following aspects:

• OCR-enhanced Automation – Unlike applications that need guide corrections, our OCR module incorporates AI-driven details extraction to precisely become aware of important fee information (e.g., date, amount, service provider call, description) with minimal consumer involvement.

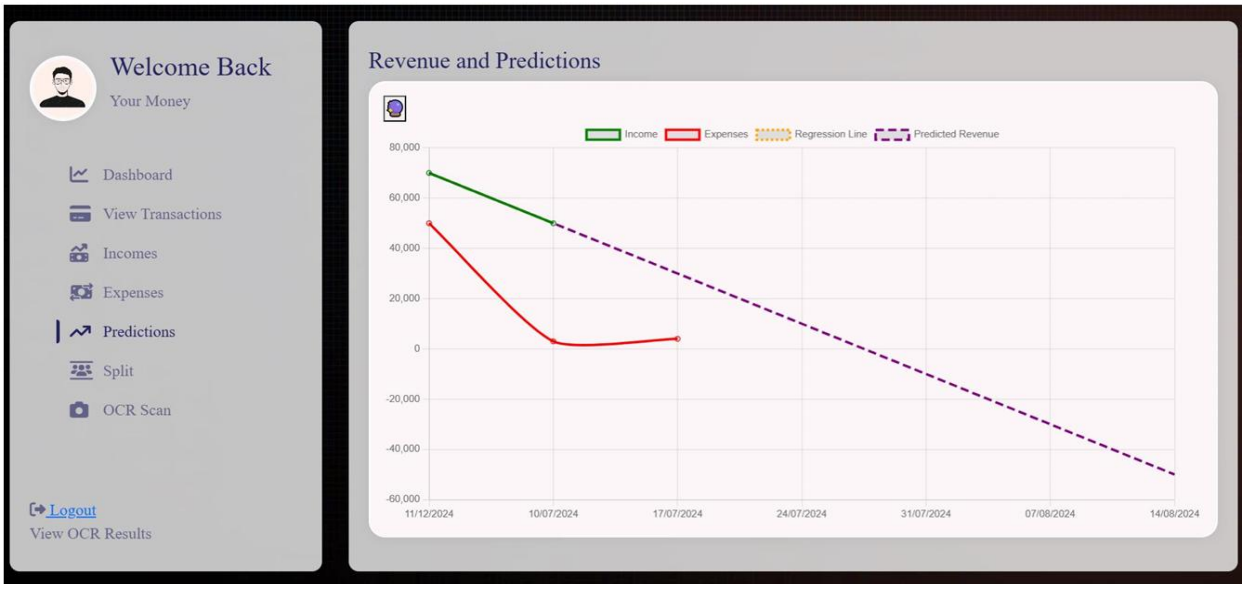
• ML-based expense Forecasting – Our platform examines historical spending habits using device mastering to deliver personalized forecasts, unlike conventional applications that supply static price reports.

• Cloud Integration for real-Time Accessibility – Unlike most packages that depend on neighborhood storage, our machine makes use of cloud technologies (AWS) to facilitate at ease, actual-time information synchronization across devices.

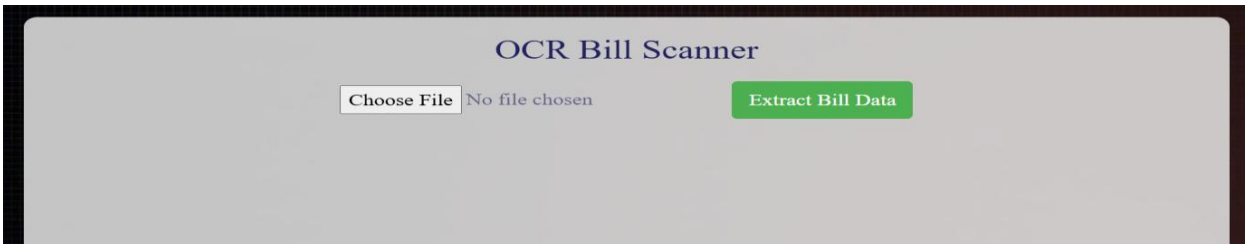
• Seamless MERN Stack Implementation – Current programs often rely on monolithic architectures, whereas our machine makes use of a modular MERN stack for excessive overall performance, scalability, and versatility

• Improve monetary insights – The app offers interactive dashboards with visual analytics, enabling customers to gain deeper insights into their spending habits.

- Revenue and ML prediction integrated in the application



- OCR Bill scanning for faster data processing and making task efficient



IDENTIFICATION OF GAPS IN EXISTING SOLUTIONS

Even with enhancements in cost monitoring software, there are still a number of gaps in existing solutions:

1. Loss of full Automation – most current systems both demand manual facts entry or have restricted OCR skills that conflict with established and semi-autonomous fiscal information.
2. Lack of Predictive Analytics – Most software does not make use of ML-based completely predictive fashions to help consumers plan their price range based mostly on past bills.
3. Limited Cloud Synchronization – although some packages provide cloud-based storage, they often do not have real-time synchronization on multiple devices for rapid access to economic records.
4. Inefficient fee Categorization – modern solutions sometimes fall short of categorizing transactions properly, erroneous budgeting findings that need guide adjustments.
5. Scalability and general performance problems – traditional packages often rely on dated architectures that fail to scale nicely for handling vast datasets and high-frequency monetary transactions.

How Our solution Bridges these Gaps

Our MERN-based price tracker with ML and OCR abilities is aimed at bridging these gaps through:

• Automating receipt extraction with an AI-better OCR module for correct receipt scanning.
• Forecasting upcoming charges the utilization of an ML-powered forecasting model primarily based on past spending patterns.
• Providing actual-time cloud synchronization for easy get admission to across devices.
• Enhancing economic understanding through an interactive and visually informative dashboard with analytics supported by AI.
• Ensuring high scalability and overall performance with the help of using the MERN stack and cloud technology.

Limitations

1. OCR Precision Limitations

• At the quality of the receipt image depends on OCR-based receipt records extraction accuracy.
• Bad image, folded receipts, or illegible fonts can also lead to misreading of textual material.
• Handwritten receipts versions won't be recognized efficiently.

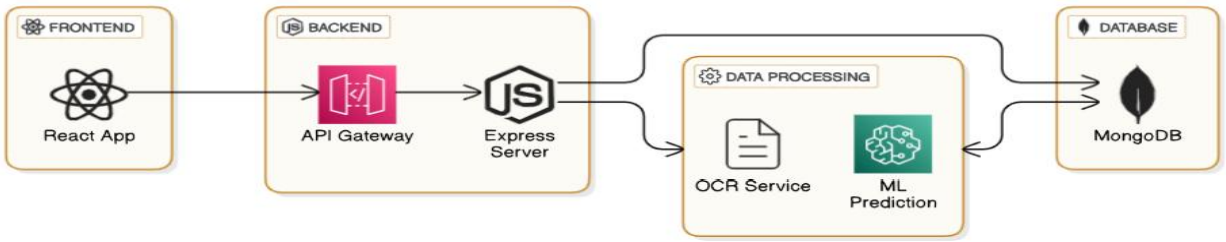
2. Machine learning prediction challenges

• The cost forecasting model requires ample historical transaction data to provide proper predictions.
• ML styles such as Regression also experience overfitting or underfitting mainly on the basis of dataset satisfactory.
• Predictions will not always be accurate for significantly irregular spending patterns.

3. Scalability issues

• Processing large amounts of prices and receipt images can also lead to latency issues, especially when multiple users add receipts at the same time.
• The performance of the machine may also slow down without adequate cloud infrastructure and optimization.

METHODOLOGY / SYSTEM DESIGN



SYSTEM OVERVIEW

The envisioned machine is a smart price tracker utilizing the MERN stack for full-stack development, OCR technology for invoice statistics extraction, and machine learning gaining knowledge for fee prediction and categorization. The software is programmed to enhance monetary management through automating records entry, future expense prediction, and informative analytics.

Workflow of the system

1. Consumer uploads a receipt → OCR extracts and systems price information.
2. Person manually input or modify fee information if required.
3. ML model examines spending behaviors and forecasts expenses in the future.
4. NLP-based categorization categorizes the charges into their respective classes.
5. User dashboard graphical presentations analytics and cost patterns.

ARCHITECTURE OF THE SYSTEM

8.1 System Architecture Overview

1. Frontend – React.js

• provides a user-friendly interface for price tracking, viewing analytics, and importing receipts.
• manages interactions with the backend APIs.

2. Backend – Node.js & express.js

• handles authentication, OCR processing, ML predictions, and API requests.
• interacts with the MongoDB database for facts storage.

3. Database – MongoDB

• stores user data, fee records, and prediction insights in a based format.

4. Cloud Integration – AWS

• stores receipts securely on AWS S3.
• utilizes Firebase Authentication or JWT for secure consumer access.

5. Machine Learning Module

• Trained on historical fee data to be expecting future prices using models such as Linear Regression.
• Utilized using Linear Regression, Python (Flask), or FastAPI for backend integration.

6. OCR Processing Module

• Pulls records from uploaded receipts using Tesseract OCR and API.
• Parses principal financial facts including date, merchant, and category, description.

ALGORITHMS, FRAMEWORKS, AND TOOLS USED

This subsection enumerates the most important techn ologies and frameworks employed in your implementation.

9.1 Algorithms Used

- OCR Algorithm – Tesseract OCR
  - Screens text from receipts, applies image pre-processing, and retrieves structured data.
- Machine Learning Algorithm – Expense Prediction
  - Linear Regression trained on past spending data to forecast future expenses.
- Data Categorization Algorithm
  - NLP-based classification TensorFlow to categorize expenses automatically.

9.2 Frameworks & Libraries

- Frontend: React.js (UI framework), Redux (State Management), Material UI/Tailwind CSS (Styling)
- Backend: Node.js, Express.js (REST API Development)
- Database: MongoDB (NoSQL database for storing expenses)
- Machine Learning: Regression (for training ML models), Flask (for ML model integration)
- OCR Processing: Tesseract OCR
- Authentication & Security: Firebase Auth, JWT (for secure login and data access)
- Cloud Services: AWS S3

IMPLEMENTATION DETAILS

10.1 Software Requirements

- Operating System: Windows
- Programming Languages: JavaScript (React.js, Node.js, Express.js), Python (ML & OCR)
- Database: MongoDB (Atlas for cloud storage)
- Cloud Services: AWS for data storage and authentication
- ML & OCR Tools: Tesseract OCR.

10.2 Hardware Requirements

- Development Machine: Minimum 8GB RAM, i5 processor
- Server Requirements: Cloud-based AWS EC2
- Storage: Cloud storage for receipts, MongoDB Atlas for database

10.3 Dataset Used

- Expense Dataset: Custom dataset containing past expenses, categories, and user transactions.
- Receipt Dataset: Public datasets or manually collected receipts used for OCR training.

REFERENCES

[1] Agile Methodology:
- Beck, K., et al. (2001). *Manifesto for Agile Software Development*. Available at: https://agilemanifesto.org

[2] MERN Stack Development:
- MongoDB, Inc. (2024). *MongoDB Documentation*. Available at: https://www.mongodb.com/docs/
- React Team. (2024). *React Documentation*. Available at: https://react.dev/
- Node.js Foundation. (2024). *Node.js Documentation*. Available at: https://nodejs.org/en/docs/
- Express.js. (2024). *Express.js Guide*. Available at: https://expressjs.com/en/starter/guide.html

[3] OCR Technology:
- Google Cloud. (2024). *Google Cloud Vision API Documentation*. Available at: https://cloud.google.com/vision/docs/

[4] Machine Learning for Expense Prediction:
- Breiman, L. (2001). *Random Forests*. *Machine Learning*, 45(1), pp. 5-32. Available at: https://doi.org/10.1023/A:1010933404324

[5] Natural Language Processing for Expense Categorization:
- Abadi, Kutner, M. H., Nachtsheim, C. J., & Neter, J. (2004). Applied Linear Statistical Models (5th ed.). McGraw-Hill/Irwin. https://www.amazon.com/Applied-Linear-Statistical-Models-Michael/dp/007310874X

[6] Cloud Integration for Expense Tracker:
- Amazon Web Services (AWS). (2024). *AWS S3 Documentation*. Available at: https://docs.aws.amazon.com/s3/index.html