International Journal of Scientific Research & Engineering Trends
Volume 12, Issue 5, Sep-Oct-2026, ISSN (Online): 2395-566X

Cloud-Based Expense Tracking Systems: A Review of Architectures, Intelligent Receipt Processing, Analytics and Security

Bacche Sanket Kailas[1], Prof. S. K. Said[2], Prof. M. C. Kolse[3]
Department of Artificial Intelligence and Data Science
Jaihind College of Engineering

Abstract — Managing personal and organizational expenses with paper receipts and spreadsheets is slow, error-prone and offers little insight into spending behaviour. Cloud-based expense trackers address these problems by centralizing financial records, synchronizing them across devices and applying analytics to reveal patterns. This paper reviews the technologies behind such systems. It examines cloud service and deployment models, intelligent capture of receipts through Optical Character Recognition (OCR) and document-understanding models, automatic categorization of transactions, budget forecasting and anomaly detection, and integration with banking data through consented interfaces such as India's Account Aggregator framework. Security is treated as a core requirement: the review discusses encryption in transit and at rest, multi-factor authentication, role-based access control for multi-user organizations and common web-application vulnerabilities. A five-layer reference architecture is proposed that separates client applications, an API gateway, microservices, data storage and an analytics engine. A research gap analysis, objectives, comparison of storage and capture options and a discussion of challenges are presented. The review concludes that combining automated data capture with explainable analytics and strong security controls can make expense tracking both effortless and trustworthy for individuals and enterprises.

Keywords— Cloud computing, expense tracker, personal finance, OCR, receipt understanding, budget forecasting, role-based access control, data security

I. INTRODUCTION

1. Motivation

Every individual and organization needs to know where money goes, yet expense records are commonly scattered across paper bills, bank statements, UPI notifications and spreadsheets. Manual consolidation is tedious, so records are often incomplete, and the resulting picture of spending is late and inaccurate. For small businesses the problem is compounded by reimbursement approvals, tax documentation and audit requirements.

Cloud computing, defined by NIST as on-demand network access to a shared pool of configurable resources [1], removes the need for users to maintain their own servers and allows data to be available on any device. Armbrust et al. [2] described how elasticity and pay-per-use pricing make cloud platforms attractive for applications whose load varies over time, which is exactly the pattern of expense tracking, with spikes at month end.

This paper reviews how cloud platforms, document-AI and analytics can be combined into a secure and user-friendly expense tracker. Its contributions are a survey of relevant techniques, a reference architecture, an analysis of research gaps and a set of design recommendations.

2. Objectives of the Review

The objectives of this review are:

- To study cloud service models and architectures suitable for expense tracking.
- To review OCR and document-understanding techniques for automatic receipt capture.
- To examine analytics methods for categorization, forecasting and anomaly detection.
- To propose a secure, scalable reference architecture for individual and organizational use.
- To identify challenges and future enhancements for Indian users.

II. OVERVIEW OF KEY CONCEPTS

1. Cloud Service Models

Expense trackers can be built on Infrastructure-as-a-Service (virtual machines), Platform-as-a-Service (managed databases and runtimes) or serverless functions. Serverless computing [3] is especially suitable for event-driven tasks such as processing

© 2026 IJSRET

an uploaded receipt, because resources are consumed only when an event occurs.

2. Functional Components

The major functions of an expense tracker are summarized in Fig. 1. Data capture covers manual entry, receipt scanning and bank imports; organization covers categorization and tagging; insight covers dashboards, budgets and forecasts; and control covers approvals, access rights and audit trails.

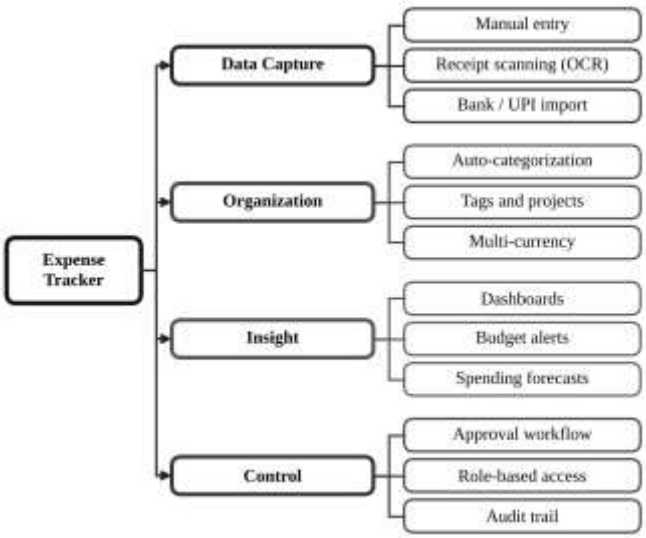


Fig. 1. Functional components of a cloud-based expense tracker.

3. Behavioural Perspective

Thaler's theory of mental accounting [4] explains that people mentally assign money to categories such as food or travel and make decisions within those categories. A tracker that mirrors these categories and shows timely feedback supports better budgeting behaviour.

III. REVIEW OF EXISTING WORK

Automatic receipt capture began with general OCR engines. Smith [5] described the architecture of Tesseract, an open-source engine that performs line finding, word recognition and adaptive classification, and which remains a common baseline. However, receipts are noisy, crumpled and printed on thermal paper, so raw OCR output still needs interpretation.

The ICDAR 2019 SROIE competition [6] formalized this problem by providing scanned receipts with tasks for text localization, recognition and key-information extraction such as company name, date and total. LayoutLM [7] improved extraction by jointly pre-training on text and its two-dimensional position, allowing the model to learn that a number near the word "Total" at the bottom is likely the amount payable. Donut [8] went further by removing the separate OCR step and reading document images end-to-end with a transformer.

For analytics, the Prophet model of Taylor and Letham [9] decomposes a time series into trend, seasonality and holiday effects and is well suited to forecasting monthly spending. Isolation Forest [10] isolates unusual observations with random partitions and can flag duplicate claims or unexpectedly large transactions. Text classifiers based on BERT [11] can categorize transaction descriptions that are short and abbreviated.

Security research provides the remaining building blocks. Role-based access control [12] assigns permissions to roles such as employee, manager and auditor rather than to individuals, simplifying administration in organizations. NIST SP 800-63B [13] gives guidance on authenticators and multi-factor authentication, while the OWASP Top 10 [14] lists the web-application weaknesses most often exploited. Table I summarizes the reviewed studies.

Table 1: Summary of Literature Survey

Ref.	Method	Key Findings	Limitations
[1]	NIST definition of cloud computing	Standard service and deployment model vocabulary	Definitional; no design guidance
[2]	Analysis of cloud economics and obstacles	Elasticity and pay-per-use benefits explained	Predates serverless platforms
[5]	Tesseract OCR engine	Open-source baseline for text recognition	Weak on noisy thermal receipts
[6]	SROIE receipt benchmark	Standard tasks for receipt key-information extraction	English receipts only
[7]	LayoutLM text + layout pre-training	Strong key-field extraction from forms and receipts	Requires OCR first; heavy model

© 2026 IJSRET

Ref.	Method	Key Findings	Limitations
[8]	Donut OCR-free document transformer	End-to-end reading from image to structured fields	Large compute and training data
[9]	Prophet additive time-series model	Robust trend and seasonality forecasting	Limited for very short histories
[10]	Isolation Forest	Fast unsupervised anomaly detection	Score threshold needs tuning
[12]	Role-based access control models	Scalable permission management by role	Static roles; context not considered

Identified Research Gaps

The following gaps are observed:

- Many student and commercial trackers still rely on manual entry; integrated receipt understanding for Indian bills with mixed languages is limited.
- Categorization models are rarely explained to users, reducing trust when a transaction is mislabelled.
- Consent-based bank data sharing through the Account Aggregator framework [15] is seldom integrated in academic designs.
- Few systems combine forecasting with anomaly detection to provide proactive alerts.
- Security evaluations of expense applications against standard checklists are uncommon.

IV. SYSTEM ARCHITECTURE

Fig. 2 shows the proposed five-layer architecture. Each layer can scale independently, and all inter-layer traffic is protected by TLS 1.3 [16].

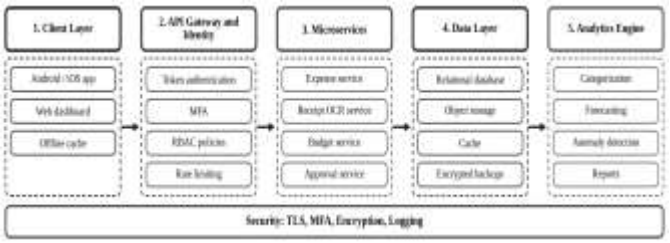


Fig. 2. Proposed five-layer cloud architecture for the expense tracker.

1. Client Layer

Android, iOS and web clients allow users to add expenses, photograph receipts and view dashboards. An offline cache stores entries when connectivity is poor and synchronizes them later.

2. API Gateway and Identity Layer

The gateway authenticates every request with tokens issued after multi-factor login, enforces rate limits and routes calls to the correct service. Roles determine whether a user may view, submit, approve or audit expenses.

3. Microservices Layer

Independent services handle expenses, receipts, budgets, approvals and notifications. The receipt service triggers a serverless OCR function that extracts merchant, date, tax and total, and returns a draft entry for user confirmation.

4. Data Layer

Transactions are stored in a relational database, receipt images in object storage, and frequently used summaries in a cache. Data at rest is encrypted with AES-256 [17], and automated backups are replicated across availability zones.

5. Analytics Layer

This layer categorizes transactions, forecasts next-month spending and flags anomalies. Results feed dashboards, budget alerts and exportable reports for tax filing.

VI. IMPLEMENTATION METHODOLOGY

The processing of a new receipt follows the workflow in Fig. 3. Budget utilization for category c in month m is computed as

$$U(c,m) = \Sigma\, spend(c,m) / budget(c,m) \times 100\% \quad (1)$$

and an alert is raised when U exceeds a user-defined level, for example 80 percent. Forecasts are generated from historical totals using an additive model in which spending y(t) is the sum of trend, seasonal and holiday components:

$$y(t) = g(t) + s(t) + h(t) + \varepsilon(t) \quad (2)$$

Anomaly detection runs nightly, assigning each transaction a score; transactions with high scores, such as a duplicate restaurant bill submitted twice, are highlighted for the approver.

© 2026 IJSRET

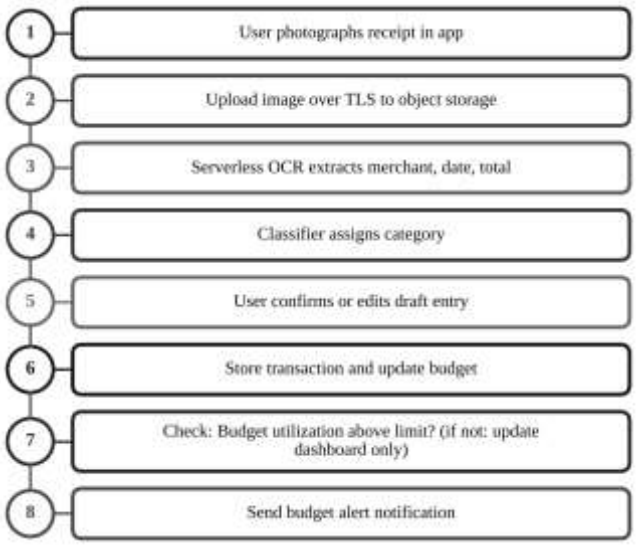


Fig. 3. Receipt processing and budget alert workflow.

VII. PERFORMANCE COMPARISON

Table II compares data-capture approaches. Manual entry is accurate but slow; plain OCR is fast but needs correction; layout-aware models and OCR-free transformers provide the best extraction quality but require more computation, which cloud deployment makes affordable.

Table 2: Comparison of Expense Capture Approaches

Approach	Speed	Accuracy	Cost / Effort
Manual entry	Slow	High (if careful)	High user effort
Plain OCR (Tesseract)	Fast	Medium	Low compute; needs correction
OCR + LayoutLM	Fast	High	Moderate cloud compute
OCR-free (Donut)	Fast	High	High training cost
Bank / AA import	Instant	High	Needs consent and integration

Open Issues and Future Directions

Table III summarizes the main challenges. In future, expense trackers can integrate UPI transaction streams directly, provide conversational queries such as "How much did I spend on travel this quarter?" using language models, support GST-compliant invoice validation, and apply federated learning so that categorization models improve without centralizing users' financial data.

Table 3: Challenges and Mitigation

Challenge	Mitigation
Poor-quality receipts	Image enhancement; user confirmation step
Data breaches	AES-256 at rest, TLS 1.3, MFA, OWASP testing
Mis-categorization	Explain labels; learn from corrections
Connectivity gaps	Offline-first sync with conflict resolution
Regulatory compliance	DPDP-aligned consent and retention policies

VIII. CONCLUSION

Cloud-based expense trackers turn scattered financial records into an organized, searchable and analysable resource. The reviewed literature shows that automatic receipt understanding has matured rapidly, that simple forecasting and anomaly-detection models can provide valuable alerts, and that well-established security practices such as role-based access, multi-factor authentication and encryption are sufficient to protect sensitive data when applied consistently. A layered, microservice-based design combining these elements can give individuals and enterprises timely visibility over cash flow and support better financial decisions.

Acknowledgment

The authors express their sincere gratitude to the Department of Artificial Intelligence and Data Science, Jaihind College of Engineering, Pune, for providing the facilities, guidance and encouragement needed to carry out this work. The authors also thank the faculty members and colleagues whose valuable suggestions helped improve the quality of this review.

REFERENCES

1. P. Mell and T. Grance, "The NIST definition of cloud computing," NIST Special Publication 800-145, 2011.
2. M. Armbrust et al., "A view of cloud computing," Commun. ACM, vol. 53, no. 4, pp. 50–58, 2010.
3. E. Jonas et al., "Cloud programming simplified: A Berkeley view on serverless computing," arXiv:1902.03383, 2019.
4. R. H. Thaler, "Mental accounting matters," J. Behav. Decis. Mak., vol. 12, no. 3, pp. 183–206, 1999.

© 2026 IJSRET

5. R. Smith, “An overview of the Tesseract OCR engine,” in Proc. 9th Int. Conf. Document Anal. Recognit. (ICDAR), 2007, pp. 629–633.
6. Z. Huang et al., “ICDAR2019 competition on scanned receipt OCR and information extraction,” in Proc. Int. Conf. Document Anal. Recognit. (ICDAR), 2019, pp. 1516–1520.
7. Y. Xu et al., “LayoutLM: Pre-training of text and layout for document image understanding,” in Proc. 26th ACM SIGKDD, 2020, pp. 1192–1200.
8. G. Kim et al., “OCR-free document understanding transformer,” in Proc. Eur. Conf. Comput. Vis. (ECCV), 2022, pp. 498–517.
9. S. J. Taylor and B. Letham, “Forecasting at scale,” Amer. Statistician, vol. 72, no. 1, pp. 37–45, 2018.
10. F. T. Liu, K. M. Ting, and Z.-H. Zhou, “Isolation forest,” in Proc. 8th IEEE Int. Conf. Data Min. (ICDM), 2008, pp. 413–422.
11. J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” in Proc. NAACL-HLT, 2019, pp. 4171–4186.
12. R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman, “Role-based access control models,” Computer, vol. 29, no. 2, pp. 38–47, 1996.
13. P. A. Grassi et al., “Digital identity guidelines: Authentication and lifecycle management,” NIST Special Publication 800-63B, 2017.
14. OWASP Foundation, “OWASP Top 10: 2021,” 2021. [Online]. Available: https://owasp.org/Top10/
15. Reserve Bank of India, Master Direction – Non-Banking Financial Company – Account Aggregator (Reserve Bank) Directions, 2016. Mumbai: RBI, 2016.
16. E. Rescorla, “The Transport Layer Security (TLS) protocol version 1.3,” IETF RFC 8446, 2018.
17. National Institute of Standards and Technology, “Advanced Encryption Standard (AES),” FIPS PUB 197, 2001 (updated 2023).

© 2026 IJSRET