IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 2, February 2026

DOI: 10.17148/IJARCCE.2026.15250

CloudWise – Intelligent AI-Driven Cloud Expense Optimizer

GOWTHAM CM[1], Dr. K S GOWRILAKSSHMI[2]

Student, Department of Computer Applications, Sri Ramakrishna College of Arts and Science (Autonomous),

Coimbatore – 641006, Tamil Nadu, India[1]

Assistant Professor, Department of Computer Applications,

Sri Ramakrishna College of Arts and Science (Autonomous), Coimbatore – 641006, Tamil Nadu, India[2]

Abstract: Cloud computing has become an essential part of modern software systems, but managing the cost of cloud services is often difficult for organizations. Companies typically receive their cloud billing information as large CSV reports that contain detailed usage and cost data. While these reports contain useful information, understanding them manually can be confusing and time-consuming. It becomes especially difficult to identify sudden cost increases or understand which services are responsible for higher spending.

CloudWise is an intelligent cloud expense analysis system designed to simplify the process of understanding cloud billing data. Instead of manually examining spreadsheets, users can upload their billing dataset directly into the system. The application automatically processes the data and converts it into meaningful insights such as monthly cost trends, service-wise spending distribution, and abnormal cost spikes.

The main feature of CloudWise is its ability to detect unusual cloud spending patterns using statistical anomaly detection. When the system identifies a sudden increase in cost for a specific month, it highlights that period and analyzes which cloud service contributed most to the spike. The system also predicts future cloud expenses using a regression-based forecasting model, allowing organizations to plan budgets more effectively.

Developed using Python Flask, Pandas, NumPy, and SQLite, the system presents its analysis through an interactive dashboard with charts, cost breakdowns, and automated insights. By transforming complex billing reports into simple visual information, CloudWise helps organizations monitor cloud spending more efficiently and respond quickly to unusual cost behavior.

Keywords: Cloud Cost Optimization, Anomaly Detection, Z-Score, Linear Regression Forecasting, Python Flask, SQLite, Interactive Dashboard, Statistical Analysis

I. INTRODUCTION

Over the past few years, cloud computing has become an essential part of how organizations operate. Companies no longer depend entirely on physical servers or fixed infrastructure; instead, they use cloud platforms to deploy applications, store large volumes of data, and scale resources whenever needed. This flexibility has transformed business operations, but it has also introduced a new challenge: managing and controlling cloud expenditure effectively.

Unlike traditional IT infrastructure where costs are relatively fixed, cloud environments operate on a usage-based billing model. Resources such as virtual machines, storage units, and network services are charged based on consumption. While this model offers scalability, it also makes spending highly dynamic. Even small configuration mistakes or unused resources running in the background can result in unexpected charges that significantly increase operational costs.

Another difficulty lies in understanding cloud billing reports. Cloud providers generate detailed usage statements containing technical service breakdowns and fluctuating cost values. In many cases, organizations review billing summaries manually or rely on basic dashboards that only display total expenditure. Such approaches provide limited insight into cost trends, service-level distribution, or irregular spending behavior.

A major limitation in traditional cost monitoring is the lack of predictive analysis. Organizations often focus only on past expenses without examining future patterns. Without forecasting mechanisms or anomaly detection techniques, businesses remain reactive rather than proactive. Identifying cost spikes after they occur does not prevent them from

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 2, February 2026

DOI: 10.17148/IJARCCE.2026.15250

recurring. There is therefore a clear need for a system that not only monitors cloud spending but also analyzes patterns and predicts future financial behavior.

This need led to the development of CloudWise, an intelligent cloud cost optimization system that processes structured billing data, performs statistical analysis, and generates meaningful insights. By integrating forecasting models and anomaly detection techniques, the system helps users understand spending trends, detect irregularities, and take informed actions to control costs through automated processing and interactive visualization.

II. OBJECTIVE

The central objective of this work is to replace manual, time-consuming cloud billing analysis with an automated system that is intelligent, efficient, and accessible to organizations of all sizes.

More specifically, this project aims to build a working system that accepts cloud billing data in CSV format, automatically cleans and processes it, and generates meaningful insights without requiring manual intervention. The system is designed to detect abnormal spending patterns using Z-score statistical analysis, flag months where costs deviate significantly from historical norms, and identify which cloud service contributed most to the anomaly.

Beyond anomaly detection, the system predicts future cloud expenditure using a Linear Regression model trained on historical billing data. This enables organizations to plan budgets proactively rather than reacting to unexpected charges. All results are presented through an interactive dashboard displaying cost trends, service breakdowns, forecast visualizations, and AI-generated summaries.

This work also aims to demonstrate that intelligent cost analysis can be implemented practically using accessible open-source tools — Flask, SQLite, Pandas, and Scikit-learn — without requiring expensive enterprise-grade infrastructure. The barrier to building something meaningfully better than manual spreadsheet analysis is lower than it might appear, and CloudWise is an attempt to make that argument concretely.

III. EXISTING SYSTEM

Before CloudWise, cloud cost monitoring was generally performed using billing dashboards provided by cloud service providers or through manual analysis of billing reports exported in CSV format and reviewed using spreadsheet software such as Microsoft Excel.

The typical workflow involves exporting billing data from the cloud provider's console, opening it in spreadsheet software, cleaning the data manually, converting date formats, creating pivot tables, performing manual calculations, and inspecting cost values for irregularities. This process is repeated every billing cycle, making it both time-consuming and prone to error.

More sophisticated cloud cost management platforms do exist, but they tend to be expensive, cloud-provider-specific tools that require dedicated setup and integration expertise. For smaller organizations, development teams, or academic projects, such solutions are not practical. What exists in practice is far more modest, and its limitations — long analysis delays, missed anomalies, no forward-looking forecasts — form the motivation for the system proposed here.

Drawbacks of the Existing System

- Reporting Delay: Cost analysis is performed only after the billing cycle ends, providing no real-time visibility.
- Time-Consuming Data Preparation: Multiple manual steps are required before analysis can begin.
- Risk of Human Error: Spreadsheet mistakes and incorrect formulas can lead to inaccurate conclusions.
- No Predictive Capability: Existing approaches cannot forecast future cloud expenses.
- No Automated Anomaly Detection: Unusual spending patterns may go unnoticed until a large bill arrives.
- Lack of Centralized Data Management: Reports are stored as separate files with no unified historical view.

IV. METHODOLOGY

CloudWise was developed using a modular, incremental approach. Each functional component — data upload, processing, anomaly detection, forecasting, and visualization — was designed and tested independently before being integrated into the complete system. This ensured that each module was individually validated before final assembly.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 2, February 2026

DOI: 10.17148/IJARCCE.2026.15250

The system follows a three-tier architecture. The presentation layer manages user-facing interaction through the upload interface and interactive dashboard. The application layer, built with Python Flask, handles data processing, statistical analysis, and forecasting logic. The data layer uses SQLite to store uploaded datasets, monthly cost records, forecast values, anomaly records, and AI-generated insights.

CSV Upload and Data Processing

When a user uploads a billing dataset, the system validates the file to ensure it is in CSV format and contains the required columns: Month, TotalCost, Compute, Storage, Network, and Other. After validation, the data is cleaned by converting date formats to a standardized form, ensuring all cost columns are treated as numeric values, removing null entries, and sorting records chronologically. Month-over-month cost growth is then calculated for each record and stored in the database.

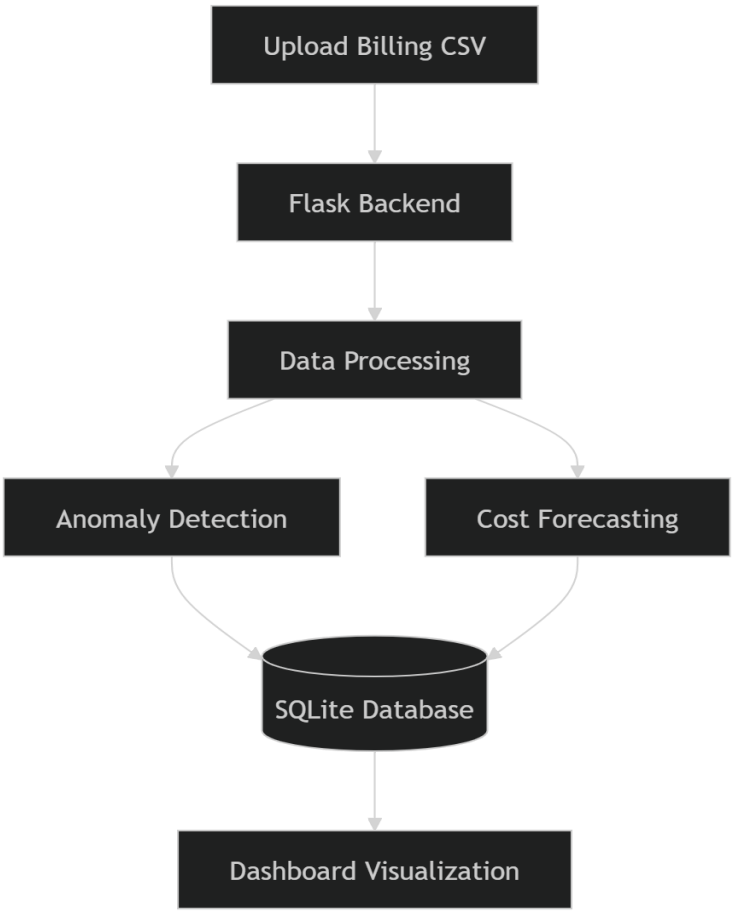
Anomaly Detection

The anomaly detection module analyzes processed monthly cost data using the Z-score statistical method. For each month, the system calculates how far the total cost deviates from the historical mean in units of standard deviation. Months where the absolute Z-score exceeds a defined threshold are flagged as anomalies. When an anomaly is detected, the system identifies which cost category — Compute, Storage, Network, or Other — contributed most to the spike, providing actionable diagnostic context alongside the alert.

Predictive Forecasting

The forecasting module uses a Linear Regression model trained on historical monthly cost data. The model learns the relationship between time and expenditure, then extrapolates this trend to estimate costs for the next three months. This gives organizations a forward-looking view of expected spending, enabling more informed budget planning and earlier identification of potential overspend

Figure 1: System Architecture of the CloudWise Cloud Cost Analysis Platform.



IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 2, February 2026

DOI: 10.17148/IJARCCE.2026.15250

System Architecture Overview

Layer	Technology	Key Responsibilities
Presentation Layer	HTML, CSS, JavaScript, Chart.js	Upload interface, dashboard display, chart rendering, client-side validation
Application Layer	Python Flask, Pandas, NumPy, Scikit-learn	Data processing, anomaly detection, forecasting, routing, AI summary generation
Database Layer	SQLite	Dataset storage, monthly cost records, forecasts, anomalies, insights

Dashboard and Visualization

All analytical results are presented through an interactive dashboard. Three summary cards display the next-month forecast value, the total anomaly count, and an efficiency score derived from cost stability. A line chart shows historical costs alongside the three-month forecast trend, with a dashed line differentiating predicted from historical values. A pie chart shows service-level cost distribution for the most recent month. An AI-generated summary paragraph describes the overall trend direction, highlights anomalous periods, identifies the service responsible for the largest spike, and provides cost control recommendations. Users can also switch between previously uploaded datasets for historical comparison.

V. RESULT AND DISCUSSION

Testing covered four areas: unit tests for individual analytical modules, integration tests for backend-database communication, system tests for end-to-end workflows, and browser compatibility tests. All test cases produced correct and consistent results.

The anomaly detection logic was validated using test datasets containing deliberate cost spikes. The Z-score method correctly flagged anomalous months while leaving stable periods unmarked. The service attribution logic also correctly identified which cost category contributed most to each detected spike, confirming that the module provides both detection and diagnosis in a single pass.

The forecasting module was tested with datasets exhibiting increasing, decreasing, and stable cost trends. In all three cases, the Linear Regression model produced sensible predictions that followed the established historical pattern. Forecast values were displayed correctly on the dashboard with a dashed-line separator between historical and predicted data points.

Dataset isolation was verified by uploading multiple CSV files sequentially. Each dataset was stored independently in the database with its own monthly records, forecasts, anomaly entries, and insights. Switching between datasets in the dashboard correctly loaded the corresponding analysis without any data leakage between sessions.

All processing operations completed within acceptable response times for typical billing datasets. The dashboard rendered correctly and consistently across Google Chrome, Mozilla Firefox, and Microsoft Edge.

Validation Test Results

Scenario	Input Conditions	Expected Output	Result
Anomaly detection – spike	Single month with cost 3x the historical average	Month flagged as anomaly	Pass
Anomaly detection – stable	All months with consistent cost values	No anomalies flagged	Pass
Forecasting – upward trend	Six months of steadily increasing costs	Forecast shows continued increase	Pass
Forecasting – stable trend	Six months of consistent costs	Forecast shows stable values	Pass
Service attribution	Spike caused by Compute category	Compute identified as top spike service	Pass
Dataset isolation	Two separate CSV uploads processed	Each dataset retrieved independently	Pass

International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 2, February 2026
DOI: 10.17148/IJARCCE.2026.15250

VI. CONCLUSION

Cloud cost management is a challenge that quietly grows more complex as organizations scale their infrastructure. Without automated analysis, billing reports remain difficult to interpret, unusual spending goes undetected, and future costs cannot be anticipated. CloudWise was built to address these gaps directly.

CloudWise — Intelligent AI-Driven Cloud Expense Optimizer — was developed to simplify cloud billing analysis and reduce the effort associated with manual cost monitoring. The system demonstrates how complex billing data can be automatically processed and converted into actionable insights through statistical analysis and interactive visualization. The Z-score-based anomaly detection reliably identifies unusual spending periods, and the Linear Regression forecasting model provides a practical forward-looking view of expected expenditure.

The modular architecture separates the user interface, processing logic, and database layer, making the system straightforward to maintain and extend. Testing confirmed that the system processes datasets correctly and produces consistent analytical results across multiple configurations and browsers. As a functional, accessible cloud cost analysis platform built entirely on open-source tools, CloudWise achieves its stated objectives and establishes clear pathways for future enhancements including real-time API billing integration, advanced forecasting models, and multi-user access control.

FUTURE ENHANCEMENTS

Future Enhancement	Description
Real-Time Cloud Billing Integration	Connect directly with AWS, Azure, or Google Cloud APIs to automatically import real-time billing data
Advanced Forecasting Models	Implement Random Forest or LSTM models for improved prediction accuracy
Real-Time Cost Monitoring	Continuous monitoring with automated alerts when abnormal cost spikes are detected
Cost Optimization Suggestions	Recommendations for underutilized resources, rightsizing, and cost-saving strategies
Multi-User Access Control	User authentication and role-based access for secure multi-user environments
Budget Monitoring Module	Define monthly budgets and compare predicted spending with allocated budget
Enterprise Database Support	Replace SQLite with PostgreSQL or MySQL for larger datasets and concurrent access
Automated Report Generation	Downloadable PDF or Excel reports summarizing trends, forecasts, and anomalies

REFERENCES

[1]. Grinberg, M., Flask Web Development: Developing Web Applications with Python, O'Reilly Media, 2nd Edition, 2018.
[2]. McKinney, W., Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython, O'Reilly Media, 2nd Edition, 2017.
[3]. Pedregosa, F. et al., "Scikit-learn: Machine Learning in Python," Journal of Machine Learning Research, vol. 12, pp. 2825-2830, 2011.
[4]. Owens, M. and Allen, G., The Definitive Guide to SQLite, Apress, 2nd Edition, 2010.
[5]. Chandola, V., Banerjee, A., and Kumar, V., "Anomaly Detection: A Survey," ACM Computing Surveys, vol. 41, no. 3, pp. 1-58, 2009.
[6]. Hyndman, R.J. and Athanasopoulos, G., Forecasting: Principles and Practice, OTexts, 3rd Edition, 2021.

