Electronic Journal of Education, Social Economics and Technology
Vol. 6, No. 2, (2025), pp. 1~9, Article ID: 1235
ISSN 2723-6250 (online)
DOI: https://doi.org/10.33122/ejeset.v6i2.1235

SAINTIS publishing

Research Article

Cash Flow Prediction System of PT Gudang Garam Using ERP-Integrated LSTM

Muhammad Ananta Arya Novandi[1]*, Ida Nurhaida[1,2], Irma Paramita Sofia[2,3], and Fitriyah Nurhidayah[2,3]

[1] Department of Informatics, Universitas Pembangunan Jaya, Tangerang Selatan, Indonesia

[2] Center of Urban Studies, Universitas Pembangunan Jaya, Tangerang Selatan, Indonesia

[3] Department of Accounting, Universitas Pembangunan Jaya, Tangerang Selatan, Indonesia

*Corresponding Author: muhammad.anantaarya@student.upj.ac.id | Phone: +6285156880504

ABSTRACT

Enterprise Resource Planning (ERP) applications such as Odoo generally do not have predictive analytics capabilities for time series data and are limited to recording historical financial data. The limitations of ERP systems make companies dependent on traditional statistical models such as ARIMA, which often fail to capture complex non-linear patterns in financial data. However, the ability to accurately predict cash flow is crucial for strategic financial management in companies. This study aims to develop and evaluate a predictive model using a Long Short-Term Memory (LSTM) approach that is accurate and integrated into Odoo ERP. The research method includes designing a microservices architecture with FastAPI as a bridge between Odoo ERP, the predictive model, and prediction graph visualization. The LSTM model is evaluated by comparing it with the ARIMA model using 3,740 Daily cash flow data, with evaluation metrics MAE, MAPE, R2. System testing will use Black Box Testing and White Box Testing. The research results show that LSTM significantly outperforms the ARIMA model with an R2 evaluation of 0.8801 and an accuracy of 96.62%. The system testing results also yielded positive outcomes as the integration architecture runs stably and functionally. This research contributes by providing an Odoo ERP system that has predictive analysis capabilities with interactive graphical visualizations through Grafana, which helps companies make decisions effectively.

Keywords: Cash Flow; LSTM; ERP; Financial Prediction; AI Integratin; Odoo

1. INTRODUCTION

Cash flow represents the financial condition of a business entity and plays a crucial role in a company's ability to maintain liquidity and long-term stability (Karaaslan, 2023). Inaccurate cash-flow forecasting can lead to a decline in the company's equity value and even result in bankruptcy issues (Skogsvik et al., 2023). A study analyzing the bankruptcy of Lehman Brothers using the Z-Score and logit methods demonstrated the urgency of accurate cash-flow prediction. The findings indicated that the company's failure was driven by its inability to project and respond to declining cash flows and market conditions (Sari, 2017). This event signifies that the ability to predict cash flow is no longer optional, but rather a strategic necessity. Another study notes that effective cash-flow management is essential for ensuring business sustainability, particularly in capital-intensive industries (Tandean et al., 2024).

The advancement of technology has enabled companies to adopt Enterprise Resource Planning (ERP) systems as tools for managing their financial data. ERP systems such as Odoo offer the capability to enhance efficiency and transparency in financial management by integrating financial data into a unified platform (Firdausha Litaay et al., 2024). The recorded data are comprehensive and structured, allowing for more advanced analysis. However, ERP systems generally lack predictive analytical capabilities due to high implementation costs and the complexity of data integration (M. A. Rahman et al., 2024). The current uncertain business environment also requires project managers and financial experts to make decisions using various predictive techniques (Shahriar et al., 2021). These challenges have increased the demand for systems capable of integrating ERP transactional data with advanced analytical methods (Sarathi et al., 2022).

Over the past decade, artificial intelligence (AI) and machine learning (ML) have opened new opportunities in the field of financial data analysis. One method capable of recognizing patterns in time-series data is Long Short-Term Memory (LSTM). LSTM, a type of Recurrent Neural Network (RNN), has proven effective in modeling time-series data. Its strength lies in its ability to handle long-term dependencies and to learn patterns that are highly influenced by temporal relationships (Muhammad & Nurhaida, 2025). Research indicates that LSTM outperforms traditional statistical models such as ARIMA in predicting nonlinear financial data (Sirisha et al., 2022). This makes LSTM particularly relevant for corporate cash-flow prediction, where cash-flow patterns are shaped by interconnected operational, investment, and financing factors. The advantages of LSTM in handling nonlinear data are also highlighted in another study, which shows that LSTM models perform better than other conventional machine-learning models (Taslim & Murwantara, 2024).

Based on this background, the implementation of the LSTM model within an ERP system holds significant potential to support predictive capabilities and automated decision-making across various business domains (Singh, 2025). This study aims to develop and evaluate a cash-flow prediction system using the Long Short-Term Memory (LSTM) method integrated with the Odoo ERP system. The system employs FastAPI to connect the prediction module with the ERP (Odoo) and uses Grafana to visualize the prediction results. Through this approach, companies are expected to obtain more accurate information that can more effectively support strategic decision-making. This study contributes by equipping the Odoo ERP system with predictive analytics capabilities and interactive graphical visualization through Grafana, which is expected to assist companies in making decisions more effectively.

2. RESEARCH METHOD

This study employs an experimental quantitative method, with its research focus centered on software engineering and time-series predictive analysis. The research methodology is designed according to a structured system-development workflow, beginning with data collection, preprocessing, model development using the LSTM architecture, and integration with the Odoo ERP system based on a microservices approach. The system is designed to provide practical value, scalability, and modularity, enabling it to be implemented within corporate business processes, in line with the growing trend of integrating artificial intelligence into application development to support automation and decision-making (Sofian et al., 2022). The following presents the flowchart of the research process conducted.

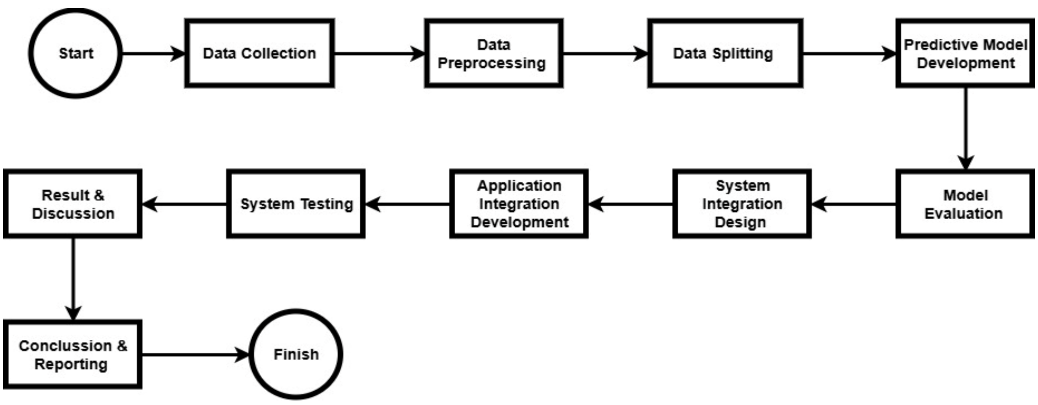

Figure 1. Research Implementation Flowchart

2.1 Data collection and Interpolation

The dataset used in this study is secondary data consisting of six variables. Its validity has been verified by an accounting lecturer at Universitas Pembangunan Jaya to ensure the accuracy of the financial values and their contextual relevance. The dataset was obtained from the Indonesia Stock Exchange (IDX) and consists of the cash-flow statements of PT Gudang Garam Tbk, covering the period from 2016 to the second quarter of 2025.

Table 1. Variable datasets

No	Variable	Description	Unit
1	kas_operasi	Cash flow from operational activities	Rupiah (IDR)
2	kas_investasi	Cash flow from investing activities	Rupiah (IDR)
3	kas_pendanaan	Cash flow from financing activities	Rupiah (IDR)
4	kurs	Daily exchange rate (IDR to USD)	Rupiah (IDR)
5	kas_awal_tahun	Beginning-of-year cash position	Rupiah (IDR)
6	kas_akhir	Total end-of-day cash	Rupiah (IDR)
7	tanggal	Transaction Date	Datetime

The financial reports issued by the IDX are quarterly, which presents a challenge for training Deep Learning models. Quarterly data tend to be sparse, making them insufficient for effective Deep Learning model training (Yi et al., 2025). To address this challenge, an interpolation technique must be applied to transform quarterly data into daily data, thereby increasing the number of observations. The application of interpolation helps represent the underlying patterns in the company's financial statements, providing enough data points for the Deep Learning training process. Interpolation techniques have been shown to be effective in financial forecasting research, as they help overcome differences in macroeconomic data frequencies (Papaioannou et al., 2022). The data generated from the interpolation process are referred to as surrogate data.

2.2 Data Preprocessing and Transformation

Data preprocessing was performed to reduce noise or scale inconsistencies that may degrade the model's convergence performance. The preprocessing stages in this study include:

1. Stationarity Check & Feature Engineering

Lag features were applied to capture temporal autocorrelation. The function of lag features is to provide the model with information from previous observations $(t-1, t-2, \ldots, t-n)$, as time-series data inherently depend on their past values (Surakhi et al., 2021) In addition, a rolling mean feature was applied to reduce short-term fluctuations and enhance long-term trends, helping the model distinguish fundamental trend signals from random noise (Salsabila et al., 2025).

2. Data Normalization

The min–max scaling method was applied to transform the entire dataset into a uniform range of 0–1. The purpose of normalization is to prevent large numerical values from dominating the training process and to empirically improve the predictive accuracy of LSTM models (Kim et al., 2025). The data normalization equation is defined as follows.

$$X_{scaled} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (1)$$

3. Sequence Generation & Splitting

The collected dataset is transformed into a supervised-learning format using the sliding-window technique. This technique has been shown to improve LSTM accuracy when working with complex time-series data (Chen et al., 2022). The dataset is structured into three components: samples, time steps, and features. The model uses data over N time steps to predict the cash-flow value at time step N + 1. The dataset is divided into 95% training data and 5% testing data.

2.3 LSTM Network Architecture

Long Short-Term Memory (LSTM) is a Recurrent Neural Network (RNN)–based algorithm designed to process and predict time-series data (Lindemann et al., 2021) Its advantage lies in its ability to perform forecasting based on sequential patterns, making LSTM a widely used method for time-series prediction tasks (Kong et al., 2024). LSTM incorporates gating mechanisms and a cell state that allow the network to retain or discard important information over time, ensuring stable gradient flow during training (Lim & Zohren, 2021). The LSTM architecture uses three primary gates to regulate the flow of information.

Input gate: Regulates the amount of new information to be stored in memory.

$$i_t = \sigma(W_{i1}z_{t-1} + W_{i2}y_t + W_{i3}x_t + W_{i4}s + b_{i)} \quad (2)$$

Forget gate: Determines which parts of previous information should be discarded.

$$f_t = \sigma(W_{f1}z_{t-1} + W_{f2}y_t + W_{f3}x_t + W_{f4}s + b_{f)} \quad (3)$$

Output gate: Controls the amount of information to be produced as the hidden output.

$$o_t = \sigma(W_{o1}z_{t-1} + W_{o2}y_t + W_{o3}x_t + W_{o4}s + b_{o)} \quad (4)$$

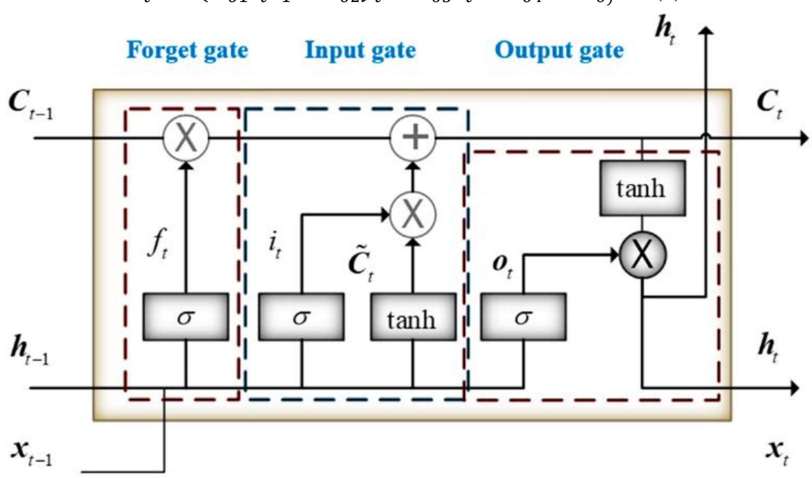

Figure 2. LSTM Architecture

The LSTM model in this study was developed using Keras (TensorFlow). The architectural details of the LSTM model include the following components.

Table 2. LSTM Parameter

Parameters	Value / Configuration	Description / Purpose
LSTM Layer	2 layers, 64 neurons (each)	To understand long-term temporal patterns
Dense Layer (Hidden)	128 neurons, Activation: ReLU	To handle nonlinear representations
Dropout Layer	Rate 0.5	To prevent overfitting
Dense Layer (Output)	1 neuron	To produce the prediction value
Epochs	20	Training looping
Batch Size	32	Datat training batch
Optimizer	Adam	Optimization model
Loss Function	Mean Absolute Error (MAE)	Loss function

2.4 Microservices Integration Design

The application integration design in this study adopts a microservices approach, utilizing FastAPI as the intermediary between the LSTM model and the Odoo ERP system. This system design separates the frontend, backend, and visualization layers to enhance modularity and scalability, while also improving overall maintainability (Charankar & Kumar Pandiya, 2023).

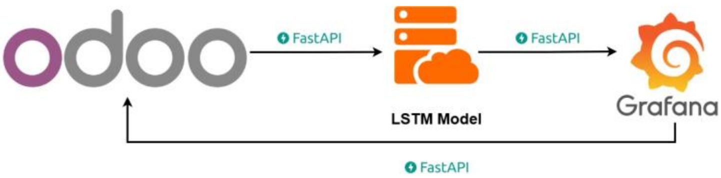

Figure 3. Design system and Architecture

The system architecture consists of three components.

1. Odoo ERP Module

The Odoo ERP module functions as the application responsible for receiving the dataset—specifically the cash-flow report formatted as a CSV file and triggering the prediction process through a custom module. This module sends the prediction request and its parameters to the backend.

2. FastAPI Service

FastAPI operates as the API gateway that connects Odoo to the LSTM model and forwards the model's output to Grafana in JSON format. FastAPI is selected for this research due to its speed and efficient support for asynchronous processing.

3. Grafana visualization

Grafana provides the system's time-series graphical visualization. In Odoo, Grafana is embedded directly, allowing users to view prediction results without leaving the Odoo environment.

2.5 Evaluation And Testing Strategy

2.5.1 Model Evaluation Metrics

Model prediction testing is conducted to evaluate the performance of the Long Short-Term Memory (LSTM) model in projecting cash flow based on historical data. The LSTM model is compared with the AutoRegressive Integrated Moving Average (ARIMA) model using MAE, MSE, RMSE, MAPE, and R^2. The prediction model evaluation is performed quantitatively using the following metrics.

1. Mean Absolute Error (MAE):

MAE measures the average absolute difference between actual values and predicted values.

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \quad (5)$$

2. Mean Squared Error (MSE):

MSE measures the average squared difference between actual values and predicted values.

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \quad (6)$$

3. Root Mean Squared Error (RMSE):

RMSE is the square root of MSE, representing prediction deviation in the same unit as the original data.

$$\text{RMSE} = \sqrt{\text{MSE}} \quad (7)$$

4. Mean Absolute Percentage Error (MAPE):

MAPE measures relative prediction accuracy expressed as a percentage.

$$\text{MAPE} = \frac{100\%}{n} \sum_{i=1}^{n} \left| \frac{y_i - \hat{y}_i}{y_i} \right| \quad (8)$$

5. Coefficient of Determination (R^2)

R^2 indicates the proportion of variability in the dependent variable that can be explained by the model, serving as an indicator of model fit (Tatachar, 2021).

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2} \qquad (9)$$

2.5.2 System Integragtion Testing

System integration testing evaluates the feasibility and stability of integrating the prediction model with the ERP system. The testing follows standard software testing practices, categorizing system testing into black-box testing for external functionality and white-box testing for internal logic implemented in the system (Sukaina Izzat & Nada N. Saleem, 2023).

1. Black Box Testing

Black-box testing ensures that module inputs and outputs operate correctly without examining internal code structure. This testing focuses on the functional behavior of the developed modules.

2. White Box Testing

White-box testing validates the internal logic of the designed processes. It includes analyzing execution flow and testing functions within the module. This type of testing helps identify potential logical errors within the system.

3. RESULTS AND DISCUSSION

3.1 Experimental Results

This research was conducted in a virtual server environment using Odoo version 18 Community Edition. The dataset was obtained from the cash-flow data of PT Gudang Garam Tbk's financial statements for the period 2016 to the second quarter of 2025, consisting of 3,740 daily interpolated data points with surrogate characteristics, meaning artificially generated data used to enrich the dataset. Surrogate data can enhance model performance, improving prediction accuracy by up to 80% (Kotios et al., 2022).

The primary model is compared with a classical statistical model, the AutoRegressive Integrated Moving Average (ARIMA), using evaluation metrics including Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE), prediction accuracy, and the Coefficient of Determination (R^2). The comparison results for the evaluation metrics are presented in **Table 3**.

Table 3. Metrix Evaluation Result

Model	MAE	MSE	RMSE	MAPE (%)	Accuracy (%)	R^2
LSTM	89.694,34	8.390.891.817,56	91.601,81	3,38%	96,62%	0,8801
ARIMA	478.275,28	368.635.739.740,81	607.153,80	16,14%	83,86%	-0,3446

Based on **Table 3**, the LSTM model demonstrates significantly better performance than ARIMA across all evaluation metrics.

1. Accuracy and Error Rate

The much lower MAE and RMSE values in the LSTM model indicate smaller absolute errors and deviations compared to the ARIMA model. The LSTM model achieves a MAPE of 3.38% and an accuracy of 96.62%. According to prior research, a MAPE value below 10% is categorized as indicating a highly accurate forecasting model (Putu et al., 2024).

2. Analysis of the Coefficient of Determination

he coefficient of determination for the LSTM model shows an R^2 value of 0.8801, indicating that the model can explain 88% of the variability in cash-flow data. In contrast, ARIMA produces a negative R^2 value of –0.3446, suggesting that the model fails to adequately capture the data's variability (Chicco et al., 2021).

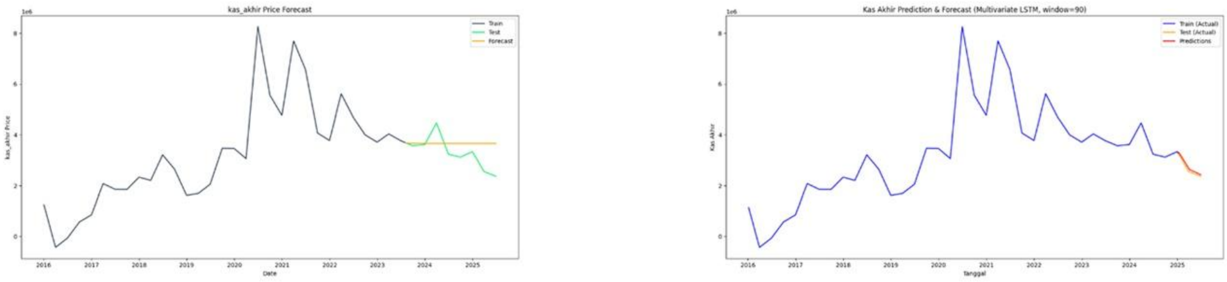

Figure 4. ARIMA Model End-of-Cash Prediction Chart and LSTM Model End-of-Cash Prediction Chart

The quantitative findings are reinforced by the predictive visualization. Figure 4 on the right illustrates that the LSTM prediction model (red line) closely follows the pattern of the actual ending cash flow (blue line). In contrast, Figure 4 on the left shows that the ARIMA prediction model (orange line) tends to remain flat and fails to adjust to the dynamics of the actual data, particularly during the 2023–2025 period. The graphical visualization demonstrates that the mechanisms implemented in the LSTM model enable it to learn long-term dependencies without experiencing vanishing gradients (Al-Selwi et al., 2023).

3.2 System Implementation

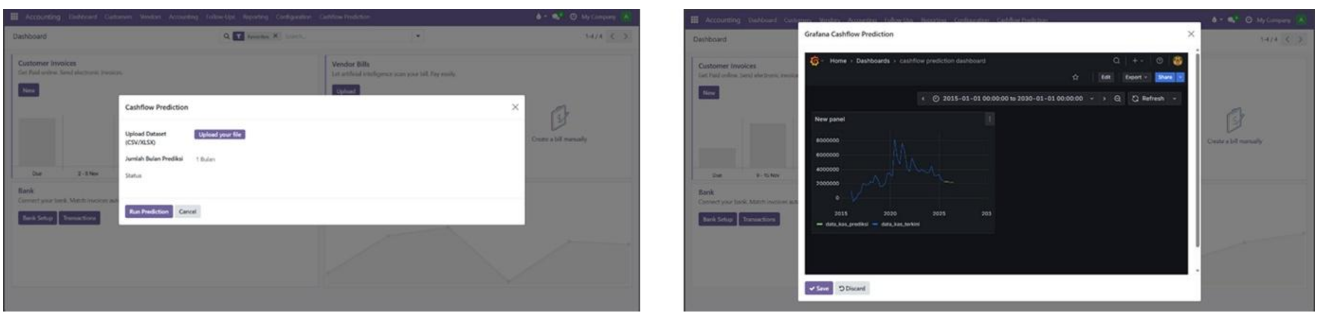

Figure 5. Odoo ERP Integrated Prediction Interface

The custom module then processes the prediction by sending the dataset through FastAPI, and the prediction results are automatically visualized in Odoo by embedding a Grafana dashboard, as shown in Figure 5. System functionality testing will be conducted using Black Box Testing and White Box Testing methods to validate the functionality and stability of the system integration (M. Rahman & David, 2022).

1. Black Box Testing

Black Box Testing focuses on testing the user flow without considering the internal code structure.

Table 4. Black Box Testing

Test ID	Test Scenario	Test Steps & Input	Expected Result	Status
BBT-01	Required File Validation	Press "Run Prediction" without uploading a file.	status_message displays: "Please upload a dataset file..." and the dialog remains open.	Pass
BBT-02	Success Flow (CSV)	Upload a valid CSV file, n_forecast=1, click "Run Prediction". (Assuming server 200 OK).	status_message displays a success message from the server (e.g., "OK") and the "View Chart" button appears.	Pass
BBT-03	Success Flow (XLSX)	Upload a valid XLSX file, n_forecast=3, click "Run Prediction". (Assuming server 200 OK).	status_message displays a success message from the server (e.g., "Prediction complete").	Pass
BBT-04	n_forecast Boundary Validation	Test n_forecast = 1 and n_forecast = 12 with a valid file.	No UI error, value is successfully sent to the server, status_message displays a success message.	Pass
BBT-05	Server Error Handling (500)	Upload valid file. (Assuming server responds 500 Internal Server Error).	status_message displays an error message (e.g., "Error: 500 Server Error...").	Pass
BBT-06	Server Handling (Non-JSON)	Upload valid file. (Assuming server responds 200 OK but with a non-JSON body).	status_message displays a fallback (e.g., "Prediction successful, but no JSON response...").	Pass
BBT-07	Server Handling (Empty JSON)	Upload valid file. (Assuming server responds 200 OK with JSON {}).	status_message displays a fallback (e.g., "Prediction successful, no message from server.").	Pass

BBT-08	Server Handling (Offline)	Upload valid file. (Assuming server is down / connection refused).	status_message displays a connection error message (e.g., "Error: ... Connection refused ...").	Pass
BBT-09	Server Handling (Timeout)	Upload valid file. (Assuming server is slow > 120 seconds).	status_message displays a timeout error message.	Pass
BBT-10	File Handling (Wrong Format)	Upload an .exe file or corrupt file.	status_message displays an error message from the server (if it rejects) or an HTTP error.	Pass

Based on the **Table 4**, the system successfully passed the black box testing, indicating that it is capable of validating user input and correctly displaying the Grafana charts within Odoo.

2. White Box Testing

White Box Testing focuses on validating the functionality of the FastAPI API endpoints that serve as the interface between Odoo and the prediction model.

Table 5. White Box Testing

	Test Category	Test Scenario	Expected Result	Actual Results	Status
WBT-01	Input Validation (Empty)	User clicks "Run Prediction" without uploading a file.	status_message contains "Please upload a dataset file...". Wizard remains open.	Exactly the same (validation if not self.dataset_file).	Pass
WBT-02	Success Flow (Happy Path)	User uploads valid CSV, n_forecast=6, clicks "Run Prediction". FastAPI is running.	File sent to /run-prediksi/. Response JSON status: success. status_message is filled with a success message.	status_message is filled from default message ("Prediction successful") since API response doesn't contain 'message' field.	Pass
WBT-03	Error Handling (Connection)	Clicks "Run Prediction" while FastAPI is down (connection refused).	requests.post exception occurs. status_message contains connection error message.	Same, error is caught and status_message is filled with the error message.	Pass
WBT-04	Input Validation (n_forecast)	User uploads valid file, but n_forecast is not filled.	Required field in Odoo prevents submit. (If sent empty, FastAPI uses default 12).	Cannot submit (due to required field).	Pass
WBT-05	Error Handling (Data)	User uploads a non-CSV file or CSV with incomplete columns.	FastAPI raises parsing error. Response JSON status: error. status_message is filled with error from server.	Same, parsing/format error appears in status_message.	Pass
WBT-06	Error Handling (Logic)	User uploads valid file but data is too small for windowing.	FastAPI raises error "Data too small...". status_message is filled with error from server.	Same, error appears in status_message.	Pass
WBT-07	Error Handling (Server)	FastAPI returns non-JSON (e.g., 500 error).	Exception on response.json(). status_message is filled: "Prediction successful, but no JSON response..."	Same, falls back to that message.	Pass
WBT-08	Response Handling (Empty)	FastAPI returns empty JSON ({}).	status_message is filled: "Prediction successful, no message from server."	Same, falls back to that message.	Pass
WBT-09	Response Handling (Message)	FastAPI returns JSON with 'message' field.	status_message is filled with the content of the 'message' field.	Same, taken from result.get('message', ...).	Pass
WBT-10	Response Handling (Default)	FastAPI returns JSON without 'message' field.	status_message is filled: "Prediction successful."	Same, falls back to default.	Pass
WBT-11	Error Handling (Timeout)	User uploads very large file, process exceeds timeout (120 seconds).	Timeout exception occurs. status_message is filled with timeout error message.	Same, error is caught and status_message is filled.	Pass
WBT-12	UI Functionality (Odoo)	After clicking "Run Prediction", wizard remains open.	Wizard remains open, status_message is updated.	Same, return action opens wizard form with the same res_id.	Pass
WBT-13	UI Functionality (Odoo)	After prediction, no temporary file remains in Odoo.	File is only in memory, no temp file on Odoo server.	Same, file is only in memory.	Pass

Based on the table above, the system successfully performed white box testing in accordance with the implemented API logic. The application of error handling functions effectively captures exceptions, ensuring system robustness against invalid input.Based on the research findings, several key points can be highlighted. First, in the context of financial forecasting, the LSTM model provides more accurate cash-flow predictions, which are essential for supporting strategic decision-making such as liquidity management, investment planning, and operational cost control. Second, in terms of business process efficiency, integrating the LSTM prediction model with the Odoo ERP platform accelerates the data-analysis workflow without requiring external data processing. Users can submit and process data within the same system in an automated manner. This research is also consistent with the study by (Taslim & Murwantara, 2024), which demonstrates that the LSTM model outperforms ARIMA in capturing nonlinear patterns and produces more stable predictions for highly fluctuating time-series data. LSTM's ability to retain long-term information through its memory-cell structure and layered mechanisms is a crucial factor explaining why it performs better than ARIMA, which relies on linear and stationary data, making it less capable of modeling dynamic corporate cash flows influenced by various external variables.With respect to system integration, the prediction system in this study offers architectural advantages. FastAPI enables efficient communication between the LSTM model and Odoo ERP, while Grafana provides interactive visualizations that can be easily interpreted by users. However, the prediction system also exhibits several limitations. The training and prediction processes of the LSTM model require relatively high computational resources, particularly for large datasets, which results in longer execution times. Additionally, the data-transfer process is performed manually by users, as the system does not automatically retrieve real-time data from Odoo ERP. Users must extract cash-flow data manually from Odoo ERP before sending it back for prediction. Furthermore, the Odoo module does not yet display evaluation metrics directly. These issues represent potential areas for improvement in future versions of the system.

4. CONCLUSION

For future development, several promising research directions can be pursued. First, the system can be enhanced by retrieving cash values used as predictive variables directly from Odoo ERP, enabling real-time forecasting based on transactional data recorded during operational hours. Second, additional external data sources may be integrated as predictive variables. The current study incorporates only exchange-rate data as an external input, but it does not include other relevant external factors. Advancing this aspect would transform the ERP system into more than a corporate information-management platform; it would evolve into an intelligent system capable of supporting business processes by learning from data in real time. Based on the research findings, the integrated LSTM prediction system within Odoo ERP demonstrates significant potential to improve the efficiency, accuracy, and strategic value of corporate financial systems. This model provides opportunities for implementing machine-learning-driven intelligent ERP systems, offering enhanced capabilities in the financial domain.

REFERENCES

Al-Selwi, S. M., Hassan, M. F., Abdulkadir, S. J., & Muneer, A. (2023). LSTM Inefficiency in Long-Term Dependencies Regression Problems. Journal of Advanced Research in Applied Sciences and Engineering Technology, 30(3), 16–31. https://doi.org/10.37934/araset.30.3.1631

Charankar, N., & Kumar Pandiya, D. (2023). Microservices and API Deployment Optimization using AI. International Journal on Recent and Innovation Trends in Computing and Communication, October. http://www.ijritcc.org

Chen, C., Zhang, Q., Kashani, M. H., Jun, C., Bateni, S. M., Band, S. S., Dash, S. S., & Chau, K. W. (2022). Forecast of rainfall distribution based on fixed sliding window long short-term memory. Engineering Applications of Computational Fluid Mechanics, 16(1), 248–261. https://doi.org/10.1080/19942060.2021.2009374

Chicco, D., Warrens, M. J., & Jurman, G. (2021). The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation. PeerJ Computer Science, 7, 1–24. https://doi.org/10.7717/PEERJ-CS.623

Firdausha Litaay, A. Y., Rahmi, I., & Yusna, Y. (2024). Implementation of Enterprise Resource Planning (Erp) Odoo on the Financial Statements of Dhuafa Schools. Simposium Nasional Akuntansi Vokasi (SNAV) XII, 520–535.

Karaaslan, O. (2023). Analysis Based on the Importance of Cash Flow Management in Business Management. Journal of Humanities, Arts and Social Science, 7(9), 1879–1883. https://doi.org/10.26855/jhass.2023.09.033

Kim, Y. S., Kim, M. K., Fu, N., Liu, J., Wang, J., & Srebric, J. (2025). Investigating the impact of data normalization methods on predicting electricity consumption in a building using different artificial neural network models. Sustainable Cities and Society, 118(October 2023), 105570. https://doi.org/10.1016/j.scs.2024.105570

Kong, Y., Wang, Z., Nie, Y., Zhou, T., Zohren, S., Liang, Y., Sun, P., & Wen, Q. (2024). Unlocking the Power of LSTM for Long Term Time Series Forecasting. 11968–11976. http://arxiv.org/abs/2408.10006

Kotios, D., Makridis, G., Fatouros, G., & Kyriazis, D. (2022). Deep learning enhancing banking services: a hybrid transaction classification and cash flow prediction approach. Journal of Big Data, 9(1). https://doi.org/10.1186/s40537-022-00651-x

Lim, B., & Zohren, S. (2021). Time-series forecasting with deep learning: A survey. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 379(2194). https://doi.org/10.1098/rsta.2020.0209

Lindemann, B., Müller, T., Vietz, H., Jazdi, N., & Weyrich, M. (2021). A survey on long short-term memory networks for time series prediction. Procedia CIRP, 99(March), 650–655. https://doi.org/10.1016/j.procir.2021.03.088

Muhammad, R., & Nurhaida, I. (2025). Penerapan LSTM Dalam Deep Learning Untuk Prediksi Harga Kopi Jangka Pendek Dan Jangka Panjang. JIPI (Jurnal Ilmiah Penelitian Dan Pembelajaran Informatika), 10(1), 554–564.

https://doi.org/10.29100/jipi.v10i1.5904

Papaioannou, P. G., Talmon, R., Kevrekidis, I. G., & Siettos, C. (2022). Time-series forecasting using manifold learning, radial basis function interpolation, and geometric harmonics. Chaos, 32(8), 21–25. https://doi.org/10.1063/5.0094887

Putu, N., Ginanti, V., Wiedyaningsih, C., & Yuniarti, E. (2024). Jurnal Farmasi Sains dan Praktis Comparison Of Forecasting Drug Needs Using Time Series Methods In Healthcare Facilities : A. 10(2), 156–165.

Rahman, M. A., Bhowmik, J., Ahamed, M. S., & Rahman, R. (2024). Opportunities and Challenges in Data Analysis Using Sap: a Review of Erp Software Performance. Global Mainstream Journal, 1(4), 50–67. https://doi.org/10.62304/ijmisds.v1i04.192

Rahman, M., & David, J. (2022). Advantages and disadvantages of literature review. Howandwhat, 21(2). https://www.howandwhat.net/advantages-disadvantages-literature-review/

Salsabila, A., Anggraini Puspita Sari, & Achmad Junaidi. (2025). Predicting CO levels using LSTM and Rolling-Features. Journal of Artificial Intelligence and Engineering Applications (JAIEA), 4(3), 2178–2183. https://doi.org/10.59934/jaiea.v4i3.1126

Sarathi, P., Pedda, R., & Jangam, S. K. (2022). Predictive Analytics in Oracle Fusion Cloud ERP : Leveraging Historical Data for Business Forecasting. 3(4), 86–95.

Sari. (2017). Analisis kebangkrutan Lehman Brothers dengan menggunakan metode Z-Score Altman dan model logit (Skripsi, Universitas Gadjah Mada). UGM Repository. [Universitas Gajah Mada]. https://etd.repository.ugm.ac.id/penelitian/detail/143102

Shahriar, M. S., Hasan, K. B. M. R., Hossain, T., Beg, T. H., Islam, K. M. A., & Zayed, N. M. (2021). Financial Decision Making and Forecasting Techniques on Project Evaluation: a Planning, Development and Entrepreneurial Perspective. Academy of Entrepreneurship Journal, 27(4), 1–7.

Singh, V. (2025). AI-Driven ERP Evolution: Enhancing Supply Chain Resilience with Neural Networks and Predictive LSTM Models. European Journal of Advances in Engineering and Technology, 12(2), 47–52.

Sirisha, U. M., Belavagi, M. C., & Attigeri, G. (2022). Profit Prediction Using ARIMA, SARIMA and LSTM Models in Time Series Forecasting: A Comparison. IEEE Access, 10(November), 124715–124727. https://doi.org/10.1109/ACCESS.2022.3224938

Skogsvik, K., Skogsvik, S., & Andersson, H. (2023). Bankruptcy Risk in Discounted Cash Flow Equity Valuation. Journal of Risk and Financial Management, 16(11). https://doi.org/10.3390/jrfm16110476

Sofian, H., Yunus, N. A. M., & Ahmad, R. (2022). Systematic Mapping: Artificial Intelligence Techniques in Software Engineering. IEEE Access, 10, 51021–51040. https://doi.org/10.1109/ACCESS.2022.3174115

Sukaina Izzat, & Nada N. Saleem. (2023). Software Testing Techniques and Tools: A Review. Journal of Education and Science, 32(2), 30–44. https://doi.org/10.33899/edusj.2023.137480.1305

Surakhi, O., Zaidan, M. A., Fung, P. L., Motlagh, N. H., Serhan, S., Alkhanafseh, M., Ghoniem, R. M., & Hussein, T. (2021). Time-lag selection for time-series forecasting using neural network and heuristic algorithm. Electronics (Switzerland), 10(20), 1–22. https://doi.org/10.3390/electronics10202518

Tandean, V. A., Kusnadi, E., Anantadjaya, S. P., & Nawangwulan, I. M. (2024). Impact of Cash Flow Management on the Sustainability of Indonesian MSMEs. Community Development Journal: Jurnal Pengabdian Masyarakat, 5(4), 8237–8241. https://repository.ipmi.ac.id/2571/1/33286-Article Text-112284-1-10-20240830.pdf

Taslim, D. G., & Murwantara, I. M. (2024). Comparative analysis of ARIMA and LSTM for predicting fluctuating time series data. Bulletin of Electrical Engineering and Informatics, 13(3), 1943–1951. https://doi.org/10.11591/eei.v13i3.6034

Tatachar, A. V. (2021). Comparative Assessment of Regression Models Based On Model Evaluation Metrics. International Research Journal of Engineering and Technology, 8(9), 853–860. www.irjet.net

Yi, K., Zhang, Q., Fan, W., Cao, L., Wang, S., He, H., Long, G., Hu, L., Wen, Q., & Xiong, H. (2025). A Survey on Deep Learning based Time Series Analysis with Frequency Transformation. In Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (Vol. 2, Issue 1). arXiv. https://doi.org/10.1145/3711896.3736571