

**SALES FORECASTING APPLICATION
ON ANDROID USING ARIMA
AND RECURRENT NEURAL NETWORK MODELS**

Tugas Akhir

**Diajukan untuk Memenuhi Salah Satu Persyaratan Mencapai Derajat Sarjana
Komputer**



Dibuat Oleh:

I Nyoman Gede Giri Mulyawan

210711412

**PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ATMA JAYA YOGYAKARTA**

2025

HALAMAN PENGESAHAN

Tugas Akhir Berjudul

SALES FORECASTING APPLICATION ON ANDROID USING ARIMA AND RECURRENT
NEURAL NETWORK MODELS

yang disusun oleh

I Nyoman Gede Giri Mulyawan

210711412

dinyatakan telah memenuhi syarat pada tanggal 23 Juli 2025

Dosen Pembimbing 1	: Th. Adi Purnomo Sidhi, S.T., M.T.	Keterangan Telah Menyetujui
Tim Penguji Penguji 1	: Th. Adi Purnomo Sidhi, S.T., M.T.	Telah Menyetujui
Penguji 2	: Martinus Maslim, S.T., M.T.	Telah Menyetujui
Penguji 3	: Stephanie Pamela Adithama, S.T., M.T.	Telah Menyetujui

Yogyakarta, 23 Juli 2025

Universitas Atma Jaya Yogyakarta

Teknologi Industri

Dekan

ttd.

Dr. Ir. Parama Kartika Dewa SP., S.T., M.T.

Dokumen ini merupakan dokumen resmi UAJY yang tidak memerlukan tanda tangan karena dihasilkan secara elektronik oleh Sistem Bimbingan UAJY. UAJY bertanggung jawab penuh atas informasi yang tertera di dalam dokumen ini

Abstract

Accurate sales forecasting is essential for effective inventory management, pricing strategies, and operational planning in the retail sector. However, many small and medium-sized enterprises lack access to affordable and intelligent forecasting tools. This thesis presents the development of a mobile-based Android application for sales forecasting, integrating statistical and machine learning techniques using ARIMA (Autoregressive Integrated Moving Average) and Recurrent Neural Networks (RNN). Leveraging Firebase Firestore for cloud-based data storage and Chaquopy for Python-Java integration, the application enables users to input or upload historical sales data and generate 1-, 2-, and 3-month predictions. The app provides a user-friendly interface with visual representations of forecasts and supports algorithm selection between ARIMA and RNN. Empirical testing using real-world sales data demonstrates that ARIMA performs well on stable trends, while RNN offers advantages for fluctuating or nonlinear patterns. This project highlights the feasibility and benefits of embedding advanced predictive models into mobile platforms, enhancing accessibility for small-scale retail decision-makers.

Keyword: Android Application, ARIMA, Chaquopy, Firebase Firestore, Machine Learning, Recurrent Neural Network, Sales Forecasting.

Acknowledgments

First and foremost, I would like to express my deepest gratitude to Mr. Chang Heyou, my thesis supervisor, Mr. Li Qing and Mr. Wu Hai Yong for their continuous support, guidance, and encouragement throughout the development of this research. Their expertise, critical insights, and thoughtful feedback have been instrumental in shaping the direction and quality of this work.

I am also sincerely thankful to the faculty and staff of the Software Engineering, Nanjing Xiaozhuang University for providing a supportive academic environment and the necessary resources to conduct this research. The knowledge and skills I gained from my courses and academic mentors laid a strong foundation for this project.

My appreciation extends to my family and close friends for their unwavering moral support, patience, and understanding during this journey. Their belief in me was a source of motivation throughout the challenges of development and writing.

Special thanks also go to the developers and maintainers of open-source tools and libraries, particularly the contributors to Chaquopy, TensorFlow, statsmodels, and Firebase, without which this application would not have been possible. And also to Spotify, who faithfully accompanies me through countless sleepless night.

Lastly, I would like to acknowledge all individuals and small business owners who inspired this project and highlighted the real-world importance of accessible sales forecasting tools. It is my hope that this work contributes in some way to supporting their efforts in data-driven decision-making.

Table of Contents

Halaman Pengesahan	2
Abstract	3
Acknowledgements	4
Table of Contents	5
List of Figures	8
List of Tables	10
Chapter 1. Introduction	11
1.1 Background	11
1.2 Significance of the Study	12
1.3 Problem Statement	12
1.4 Research Objectives	13
1.5 Scope and Limitations	13
1.6 Methodology	14
1.7 Research Benefits	14
1.8 Thesis Organization	15
Chapter 2. System Analysis	16
2.1 System Requirements Analysis	16
2.1.1 Functional Requirements	16
2.1.2 Non-Functional Requirements	17
2.1.3 Technical Requirements	18
2.2 Forecasting Algorithms Analysis	19
2.3 Summary	23
Chapter 3. System Design	24
3.1 System Overview	24
3.2 System Business Process	24
3.2.1 Flowchart	24

3.2.2 Use Case Diagram	25
3.2.3 Data Flow Diagram	30
3.3 Architectural Design	31
3.3.1 Presentation Layer (Android App UI)	32
3.3.2 Back End Layer (Chaquopy Python Integration)	32
3.3.3 Data Layer (Firebase Firestore)	32
3.4 Deployment Diagram	32
3.5 Database Diagram	37
3.5.1 Overview of Firebase Firestore	38
3.5.2 Collection 1: User	38
3.5.3 Collection 2: Sales_Prediction	39
3.5.4 Subcollection: Sales	40
3.5.5 Subcollection: Prediction	41
3.6 User Interface Design Principles	42
3.6.1 Design Goals and User Considerations	42
3.6.2 Color Theme and Visual Aesthetics	43
3.6.3 Layout Structure and Navigation Flow	43
3.6.4 Form Layout and Input Design	44
3.6.5 Data Presentation and Visual Feedback	44
3.6.6 Accessibility and Responsiveness	45
3.7 Forecasting Algorithm Comparison and Evaluation	45
3.7.1 Conceptual Comparison of ARIMA and RNN	46
3.7.2 Public Dataset Evaluation: Shampoo Sales	46
3.8 Summary	47
Chapter 4. Implementation	48
4.1 Overview	48
4.2 Development Environment and Tools	48
4.3 Firebase Integration	49
4.4 User Interface Implementation	50
4.4.1 Fragment-Based Navigation	51
4.4.2 HomeFragment – Forecast Construction Display	53
4.4.3 SalesFragment – View and Edit Historical Sales	55
4.4.4 Upload Fragment – Manual Sales Entry and Firestore Write Logic	57

4.5 Forecasting Algorithm Implementation	59
4.5.1 Chaquopy Integration	59
4.5.2 Input Data Preparation	60
4.5.3 ARIMA Forecasting Implementation	61
4.5.4 Recurrent Neural Network (RNN) Implementation	62
4.5.5 Writing Prediction to Firestore	64
4.6 Displaying Prediction and Result Visualization	66
4.6.1 Retrieving Prediction Result from Firestore	66
4.6.2 Visualizing Predictions in HomeFragment	67
4.6.3 Extensibility and Multi-Model Support	68
4.7 Error Handling and Input Validation	68
4.7.1 Input Validation	69
4.7.2 Prediction Error Handling	69
4.7.3 Firestore Write Failures	70
4.8 Summary	70
Chapter 5. Testing	72
5.1 Testing Objective	72
5.2 Testing Environment	72
5.3 Module Testing Overview	72
5.3.1 User Login and Session Handling	73
5.3.2 Sales Data and Edit Module	74
5.3.3 Editing Existing Sales Data	75
5.3.4 Generating Forecast	76
5.4 Model Validation on Public Dataset	76
5.4.1 Dataset Detail	77
5.4.2 Forecast Output	78
5.5 Possible Issue Testing	79
5.6 Summary	80
Chapter 6. Conclusion and Future Work	81
6.1 Conclusion	81
6.2 Future Work	82
References	83

List of Figures

Figures 2 – 1 ARIMA Structure Diagram	20
Figures 2 – 2 RNN Structure Diagram	22
Figures 3 – 1 System Flowchart	25
Figures 3 – 2 Use Case Diagram	26
Figures 3 – 3 Data Flow Diagram	30
Figures 3 – 4 Architectural Design	31
Figures 3 – 5 Home Fragment	34
Figures 3 – 6 Sales Fragments	35
Figures 3 – 7 Upload Fragment	36
Figures 3 – 8 Main Activity	37
Figures 3 – 9 User Class Diagram and Collection	39
Figures 3 – 10 Product Class Diagram and Collection	40
Figures 3 – 11 Sales Data Firestore 1	41
Figures 3 – 12 Sales Data Firestore 2	41
Figures 3 – 13 Sales Data Firestore 3	42
Figures 4 – 1 Navigation Design	53
Figures 4 – 2 Home Fragment Design	55
Figures 4 – 3 Sales Data Yearly and Monthly	57
Figures 4 – 4 Upload Sales Data Design	59
Figures 4 – 5 Prediction Data	66
Figures 5 – 1 User Login	73

Figures 5 – 2 Upload Sales Data	74
Figures 5 – 3 Edit Sales Data	75
Figures 5 – 4 Forecast Before and After	76
Figures 5 – 5 Public Dataset Forecast Output	78

List of Tables

Table 2 – 1 Comparison of Forecasting Algorithms	19
Table 3 – 1 Use Case 1: Input Product Name	26
Table 3 – 2 Use Case 2: Input Sales Data	27
Table 3 – 3 Use Case 3: Generate Prediction	27
Table 3 – 4 Use Case 4: View Prediction	28
Table 3 – 5 Use Case Description	29
Table 3 – 6 Algorithms Comparison	46
Table 3 – 7 Shampoo Dataset Result	47
Table 4 – 1 Development Tools	48
Table 5 – 1 Testing Environment	72
Table 5 – 2 Test Case 1	73
Table 5 – 3 Test Case 2	74
Table 5 – 4 Test Case 3	75
Table 5 – 5 Test Case 4	76
Table 5 – 6 Dataset Sample	77
Table 5 – 7 Public Dataset Forecast Output	78
Table 5 – 8 Evaluation Results	78
Table 5 – 9 Possible Issue Testing	79