

**Smart-Cost: Semantic Inventory Normalization Engine: A
Hybrid BERT + Graph-Augmented Fuzzy Resolution
Framework for Noisy Receipt Streams in Micro-SME
Supply Chains**

R26-IT-026

Project Proposal Report

Perera N.K.M – IT22151810

B.Sc. (Hons) Degree in Information Technology Specialized in
Information Technology

Department of Information Technology

Sri Lanka Institute of Information Technology Sri Lanka

DECLARATION

I declare that this is my own work, and this proposal does not knowingly incorporate any material previously submitted for a degree or diploma at any other university or higher education institution, nor does it contain any material that has been previously published or written by another person apart from where proper acknowledgment is given in the text.

Name	Student ID	Signature
Perera N.K.M	IT22151810	

The above candidate is conducting research for their undergraduate Dissertation under my supervision.

Name	Date	Signature
Mr. Deemantha Siriwardana	12 /03 /2026	

ABSTRACT

Small and medium-sized businesses such as bakeries frequently rely on paper-based receipts to record ingredient purchases. When these receipts are digitized using Optical Character Recognition (OCR), the extracted text often contains inconsistent ingredient names, abbreviations, and varying measurement units. For example, ingredient names such as “Ap Flr”, “Flour”, and “A.P. Flour” may refer to the same item but appear differently in receipt text. Existing receipt-processing systems typically store this information as raw text strings without interpreting the meaning of the ingredients, making it difficult to perform automated inventory tracking or cost analysis. This highlights a critical research gap in converting unstructured receipt text into structured and standardized data.

This research proposes a Semantic Inventory Normalization Engine designed to automatically transform inconsistent ingredient descriptions into a standardized inventory database. The proposed solution utilizes Natural Language Processing (NLP) techniques combined with a domain-specific bakery ingredient dictionary to recognize ingredient entities extracted from receipt text. To handle variations in spelling and abbreviations, fuzzy string-matching algorithms based on Levenstein Distance will be implemented to map approximate ingredient names to standardized database entries. In addition, a unit normalization module will convert different measurement units such as grams, kilograms, and pounds into a consistent metric representation. The processed information will then be stored in a structured inventory database to support automated ingredient tracking.

The system will be implemented using Python, Natural Language Processing techniques, fuzzy string-matching algorithms, and structured database technologies. The expected outcome of this research is an intelligent normalization framework capable of accurately interpreting noisy receipt text and converting it into structured ingredient records.

This research contributes to improving data accuracy, inventory visibility, and digital transformation for small businesses, enabling them to manage ingredient purchases more efficiently and reduce manual data processing.

ACKNOWLEDGMENT

I would like to take this opportunity to express my heartfelt gratitude to everyone who has contributed to the development of this project proposal report for the SmartCost Decision Support System for Micro-SMEs.

First and foremost, I would like to thank our supervisor, Mr. Deemantha Siriwardana, for suggesting and guiding us through this research problem. His expertise and valuable insights in the areas of software engineering, data processing, and research methodology have been instrumental in helping us shape the proposed solution. His continuous encouragement, constructive feedback, and guidance greatly helped us refine the research direction, particularly in the development of the Semantic Inventory Normalization component of the system.

Next, I would also like to express my sincere appreciation to our co-supervisor, Miss Ayesha Wijesooriya, for her continuous support and valuable advice throughout the development of this proposal. Her guidance in research planning, problem analysis, and system design has significantly contributed to improving the clarity and quality of this research work.

Then, I would like to thank my colleagues and appreciate the dedication and hard work they have contributed to this research project from the beginning. Their enthusiasm, collaboration, and willingness to share ideas have played a vital role in shaping overall system design. Working together as a team allowed us to explore different perspectives and develop innovative solutions for the research problem.

I would also like to acknowledge the availability of open-source technologies, academic publications, and publicly available datasets that supported the development of this proposal. These resources helped us understand the challenges related to receipt data processing, ingredient normalization, and intelligent inventory management systems.

Finally, I would like to extend my sincere appreciation to the Faculty of Computing and the Comprehensive Design and Analysis Project (CDAP) team at the Sri Lanka Institute of Information Technology (SLIIT) for providing us with the opportunity to conduct this research. Their continuous support, guidance, and academic resources have played an important role in helping us structure and complete this project proposal successfully.

TABLE OF CONTENTS

Declaration.....	Ii
Abstract.....	Iii
Acknowledgement.....	Iv
Table Of Contents	V
List Of Tables	Vi
List Of Figures	Vii
List Of Abbreviations	Viii
List Of Appendices	ix
1. Introduction	1
1.1 Background And Context	1
1.2 Literature Review	3
1.3 Research Gap	5
2. Objectives	7
2.1 Main Objective	7
2.2 Specific Objectives	7
3. Methodology	9
3.1 Research Approach	9
3.2 Technical Methods And Algorithms	10
3.3 Tools And Platforms	11
3.4 Data Collection And Ethical Considerations	12
3.5 Validation Strategy And Performance Metrics	13
4. High-Level System Architecture	14
4.1 Overview Of Proposed Solution	14
4.2 System Architecture Diagram	15
4.3 Component Interaction And Data Flow	16
4.4 Scalability And Security Considerations	17
5. User Requirements	18
5.1 Stakeholder Identification	18
5.2 Functional Requirements	19
5.3 Non-Functional Requirements	20

6. Commercialization Plan	21
6.1 Target Market And Customer Persona	21
6.2 Value Proposition	22
6.3 Revenue Model	23
6.4 Cost Estimation	24
6.5 Pricing Strategy	25
6.6 Competitive Advantage	26
6.7 Intellectual Property Considerations	27
7. Budget And Justification	28
7.1 Hardware Costs	28
7.2 Software Licenses	29
7.3 Cloud Services	30
7.4 Human Effort Cost	31
7.5 Expense Justification	32
8. Work Breakdown Structure (WBS)	33
9. Gantt Chart	35
10. References	37
11. Appendices	39
❖ Appendix A – Survey Instruments	
❖ Appendix B – System Diagrams	
❖ Appendix C – Ui Mockups	
❖ Appendix D – Risk Analysis	
❖ Appendix E – Ethical Documents	

LIST OF TABLES

Table 1: Historical Commodity Price Dataset Structure for Ingredient Price Forecasting

Table 2: Sample Dataset of Bakery Ingredient Prices Collected for Time-Series Analysis

Table 3: Data Preprocessing Steps for Cleaning and Preparing Time-Series Data

Table 4: Comparison of Forecasting Models (ARIMA vs Prophet) for Ingredient Price Prediction

Table 5: Forecast Evaluation Metrics Used for Model Validation (MAPE, RMSE, MAE)

Table 6: Example Output of Predicted Ingredient Price Trends

Table 7: Risk Factors Affecting Supply Chain Price Volatility in Micro-SME Bakeries

Table 8: System Integration Between Forecasting Module and Mobile Decision Support System

LIST OF FIGURES

Figure 1: Overall SmartCost Decision Support System Architecture

Figure 2: Data Collection and Preprocessing Workflow for Historical Ingredient Prices

Figure 3: Time-Series Forecasting Model Workflow for Ingredient Price Prediction

Figure 4: ARIMA Model Architecture for Commodity Price Forecasting

Figure 5: Facebook Prophet Forecasting Model with Trend and Seasonality Components

Figure 6: Historical Ingredient Price Trend Visualization

Figure 7: Forecasted Ingredient Price Trend with Confidence Intervals

Figure 8: Integration of Forecasting Engine with Mobile Decision Support System

Figure 9: Data Flow Between Receipt Processing, Ingredient Normalization, and Price Prediction Modules

Figure 10: Work Breakdown Structure (WBS) for SmartCost System Development

Figure 11: Gantt Chart for Project Timeline and Milestones

LIST OF ABBREVIATIONS

Abbreviation	Full Form
AI	Artificial Intelligence
API	Application Programming Interface
ARIMA	Autoregressive Integrated Moving Average
CSV	Comma Separated Values
DSS	Decision Support System
MAPE	Mean Absolute Percentage Error
MAE	Mean Absolute Error
RMSE	Root Mean Square Error
NLP	Natural Language Processing
OCR	Optical Character Recognition
SME	Small and Medium Enterprises
MSME	Micro, Small and Medium Enterprises
UI	User Interface
JSON	JavaScript Object Notation
SQL	Structured Query Language

Table 01

LIST OF APPENDICES

Appendix A: Historical Commodity Price Dataset Samples Used for Forecasting

Appendix B: Data Preprocessing Scripts and Cleaning Procedures for Time-Series Data

Appendix C: Implementation of ARIMA and Prophet Forecasting Models

Appendix D: Forecast Evaluation Results and Accuracy Analysis

Appendix E: Additional Price Trend Visualization Graphs

Appendix F: Risk Analysis of Supply Chain Volatility in Bakery Ingredient Markets

Appendix G: Ethical Considerations and Data Source Documentation

1. INTRODUCTION

1.1 Background and Context

Small and Medium Enterprises (SMEs), particularly bakeries and food production businesses, rely heavily on accurate ingredient tracking to manage operational costs and maintain profitability. In many small businesses, ingredient purchase records are maintained using paper-based receipts obtained from suppliers or retail stores. These receipts contain important information such as ingredient names, quantities, and prices. However, the data recorded in these receipts is often unstructured and inconsistent, making it difficult to automatically process and analyze.

When receipt images are digitized using Optical Character Recognition (OCR) technologies, the extracted text frequently contains abbreviated ingredient names, spelling variations, and inconsistent measurement units. For example, ingredient names such as “Ap Flr”, “Flour”, and “A.P. Flour” may all refer to the same ingredient but appear differently across receipts. Similarly, ingredient quantities may be expressed using different units such as grams, kilograms, or pounds. These inconsistencies create significant challenges when attempting to store the extracted data in structured inventory databases.

This problem particularly affects small-scale bakery owners and food production businesses that depend on accurate ingredient records to control costs and manage inventory. Without a reliable mechanism to standardize receipt text, business owners must manually interpret ingredient names and quantities before entering them into inventory systems. This manual process is time-consuming, error-prone, and inefficient.

Globally, SMEs account for approximately 90% of businesses and contribute more than 50% of global employment, highlighting their critical role in economic development. Despite their importance, many SMEs still lack access to intelligent data processing tools that can automate routine tasks such as ingredient data organization and inventory tracking. In developing economies such as Sri Lanka, the reliance on manual record keeping further increases the difficulty of maintaining accurate operational data.

Existing receipt digitization systems primarily focus on extracting textual content but rarely provide mechanisms to interpret and normalize ingredient-related information. Most systems store the extracted text as raw strings, leaving users responsible for manually identifying ingredient names and converting measurement units. This limitation prevents automated inventory analysis and reduces the effectiveness of receipt digitization technologies.

To address this issue, this research proposes the development of a Semantic Inventory Normalization Engine designed to automatically convert unstructured ingredient descriptions extracted from receipts into standardized inventory records. The proposed system utilizes Natural Language Processing (NLP) techniques combined with a domain-specific bakery ingredient dictionary to recognize ingredient entities and map abbreviated or misspelled ingredient names to standardized entries. In addition, fuzzy

string matching algorithms will be used to identify similar ingredient names, while a unit conversion module will normalize different measurement units into a consistent metric format.

This research supports the United Nations Sustainable Development Goal (SDG) 9 Industry, Innovation, and Infrastructure, which emphasizes the use of technological innovation to improve industrial efficiency and economic growth. By enabling automated normalization of ingredient data, the proposed system contributes to improving digital infrastructure for small businesses and supporting their transition toward data-driven operational management.

Furthermore, this research aligns with the Technology Integration and Management research cluster, as it integrates Natural Language Processing methods, fuzzy string matching algorithms, and structured database systems to address a practical business problem related to ingredient data processing and inventory management.

1.2 Literature Review

Recent advancements in Natural Language Processing (NLP) and data extraction technologies have enabled automated processing of unstructured textual data across various application domains. Several studies have investigated methods for extracting meaningful information from noisy textual inputs such as scanned documents, receipts, invoices, and retail transaction records. These studies provide valuable insights into techniques that can be applied for semantic normalization of ingredient-related text extracted from receipts.

Optical Character Recognition (OCR) technology has been widely used to digitize printed documents and extract textual information from images. Smith [1] introduced the Tesseract OCR engine, which has become one of the most widely used open-source OCR systems for recognizing text in scanned documents. Although OCR technology effectively extracts textual content from images, it does not provide semantic understanding of the extracted text. Consequently, additional text processing techniques are required to interpret and structure the extracted data.

In order to improve the interpretation of extracted text, several researchers have explored the use of Natural Language Processing techniques for entity recognition and data normalization. Jurafsky and Martin [2] highlight the importance of Named Entity Recognition (NER) in identifying meaningful entities within unstructured text. NER techniques have been widely used in domains such as financial document processing, medical text analysis, and retail transaction systems. However, applying general-purpose NER models to domain-specific text such as ingredient names often results in reduced accuracy due to the presence of abbreviations and domain-specific terminology.

Fuzzy string matching techniques have also been widely used to handle variations in spelling and abbreviations. Navarro [3] provides a comprehensive survey of approximate string matching algorithms, including the widely used Levenshtein Distance method. These techniques measure the similarity between two strings and are frequently used in applications where exact text matching is not

feasible. In retail data processing systems, fuzzy matching has been used to match product names across different datasets that contain inconsistent naming conventions.

Several studies have also investigated data normalization techniques for structured data integration. Rahm and do [4] discuss schema matching and data cleaning techniques used in database integration systems. Their work highlights the importance of standardizing inconsistent data formats to enable reliable analysis and integration across different systems. Similar approaches can be applied to normalize ingredient names and measurement units extracted from receipt data.

In addition to academic research, several commercial applications have attempted to automate receipt processing and expense tracking. Applications such as **Expensify**, **Veryfi**, and **Receipt Bank** provide receipt scanning services that use OCR technology to extract information such as transaction dates, vendor names, and total purchase amounts. These systems are widely used for expense management in corporate environments. However, most of these platforms focus on financial transaction processing rather than detailed semantic interpretation of item-level product information.

Another widely used application, **Google Lens**, provides image recognition capabilities that can extract text from images and documents. While Google Lens can successfully extract receipt text, the system does not automatically interpret abbreviated product names or convert measurement units into standardized formats. As a result, users still need to manually interpret and organize the extracted information.

Despite the availability of these technologies, existing systems still face several limitations when applied to ingredient management in small-scale food businesses. First, most systems store extracted receipt text as raw strings without performing semantic normalization. Second, existing NLP models are typically trained on general text corpora and may not perform well when processing domain-specific ingredient terminology. Third, existing receipt-processing applications rarely include automated unit conversion mechanisms to standardize ingredient quantities.

These limitations highlight the need for a specialized solution that can automatically interpret ingredient-related text extracted from receipts. The proposed **Semantic Inventory Normalization Engine** aims to address these challenges by integrating domain-specific ingredient dictionaries, fuzzy string matching techniques, and unit normalization algorithms. By transforming inconsistent receipt text into structured ingredient records, the proposed system aims to improve the usability of digitized receipt data and support automated inventory management for small businesses.

1.3 Research Gap

Although many receipt scanning and expense management systems exist, most of them only extract text from receipts using OCR technology and store the extracted information as raw text. These systems do not interpret the meaning of the extracted data, especially ingredient names that appear in different formats or abbreviations. As a result, users still need to manually organize and standardize the information before it can be used for inventory management or analysis.

Additionally, existing Natural Language Processing solutions are usually trained on general text datasets and do not include domain-specific knowledge related to bakery or food ingredient terminology. Because of this limitation, many systems fail to correctly recognize ingredient names that appear in abbreviated or inconsistent forms such as “Ap Flr”, “Brn Sgr”, or “Coco Pwdr”.

Another limitation is that most current systems do not provide automatic measurement unit normalization. Ingredient quantities may appear in different units such as grams, kilograms, or pounds, which makes it difficult to maintain consistent inventory records.

Therefore, there is a clear need for a system that can semantically interpret receipt text and convert inconsistent ingredient descriptions into standardized inventory data. The proposed Semantic Inventory Normalization Engine addresses this gap by combining domain-specific ingredient dictionaries, fuzzy string matching algorithms, and automated unit conversion techniques to transform unstructured receipt text into structured inventory records.

2. OBJECTIVE

2.1 Main Objective

The main objective of this research component is to develop a semantic inventory normalization engine capable of converting unstructured receipt text into standardized and structured ingredient records, enabling accurate inventory management for small food-related businesses such as bakeries, cafés, and home-based food enterprises.

2.2 Specific Objectives

To achieve the main objective, the following specific objectives are defined:

Objective 1 – Develop a Domain-Specific Ingredient Dictionary

Creating a dictionary that contains common ingredient names used in bakeries and food businesses. This dictionary will also include common abbreviations found in receipts so the system can recognize different variations of ingredient names.

Objective 2 – Implement Fuzzy String Matching for Ingredient Normalization

To develop a method that can identify ingredient names even when they appear in different forms or spellings in receipts. Fuzzy string matching will be used to match these variations with the correct standardized ingredient name.

Objective 3 – Extract Ingredient Quantities and Units from Receipt Text

To develop a method that can identify and extract ingredient quantities and measurement units such as grams, kilograms, and liters from receipt text generated by the OCR system.

Objective 4 – Normalize Measurement Units for Inventory Consistency

To convert different measurement units found in receipts into a single standard format. This helps ensure that ingredient quantities are stored consistently in the inventory system.

Objective 5 – Design a Structured Inventory Database Schema

To design a database structure that can store normalized ingredient information such as ingredient name, quantity, measurement unit, and purchase details in an organized format.

Objective 6 – Evaluate the Performance of the Normalization Engine

To test the system using sample receipt data and measure how accurately the system identifies ingredient names and converts them into standardized inventory records.

3. METHODOLOGY

The methodology of this research describes the systematic approach used to design, develop, and evaluate the Semantic Inventory Normalization Engine, which is responsible for converting unstructured ingredient descriptions extracted from receipt data into standardized inventory records. This component focuses on interpreting inconsistent ingredient names, resolving abbreviations, and normalizing measurement units to ensure accurate and structured data storage. The methodology combines Natural Language Processing techniques, fuzzy string matching algorithms, and structured database management to address the challenges associated with processing noisy receipt text.

3.1 Research Approach

This research adopts a Design Science Research (DSR) approach, which focuses on developing and evaluating an innovative technological artifact that addresses a real-world problem. Design Science is particularly suitable for this study because the research aims to design a functional system component capable of transforming unstructured textual data into structured inventory information.

The research process follows several stages. First, the problem domain is analyzed by examining how small and medium-sized food businesses currently record ingredient purchases using receipts. This stage involves identifying the challenges associated with inconsistent ingredient naming conventions, abbreviations, and varying measurement units.

Second, the system artifact is designed by proposing a Semantic Inventory Normalization Engine that integrates Natural Language Processing techniques with domain-specific knowledge. The system architecture includes modules for ingredient recognition, fuzzy matching, unit extraction, and data normalization.

Third, the proposed system is implemented using suitable programming languages and text-processing libraries. The implementation focuses on converting raw OCR-extracted text into structured ingredient data.

Finally, the artifact is evaluated using experimental validation techniques to measure the accuracy and reliability of the normalization process. Performance evaluation will determine how effectively the system converts unstructured receipt text into standardized inventory records.

3.2 Technical Methods and Algorithms

The proposed system utilizes several computational techniques to process receipt text and normalize ingredient data. These techniques include entity recognition, fuzzy string matching, unit extraction, and quantity normalization.

First, ingredient entity recognition is performed to identify ingredient names within the raw text extracted from receipts. This process uses a domain-specific ingredient dictionary containing common bakery ingredient names such as flour, sugar, cocoa powder, butter, and baking powder. The dictionary enables the system to recognize ingredient-related words within unstructured text.

Second, fuzzy string matching algorithms are used to resolve variations in ingredient names. Receipt text often contains abbreviations and spelling variations such as “Ap Flr”, “Brn Sgr”, or “Coco Pwdr”. To address this issue, the system calculates similar scores between extracted words and entries in the ingredient dictionary using the Levenshtein Distance algorithm. The ingredient name with the highest similarity score above a predefined threshold is selected as the standardized ingredient label.

Third, regular expression (Regex) techniques are used to extract numerical values and measurement units from receipt text. This enables the system to identify ingredient quantities such as “500g”, “1kg”, or “2 lb”.

Fourth, a unit conversion module is implemented to normalize different measurement units into a standard metric format. For example, quantities expressed in grams or pounds are converted into kilograms to ensure consistent inventory representation.

Finally, the normalized ingredient name, quantity, unit, and price information are stored in a structured relational database, allowing the system to maintain organized inventory records that can be used by other modules of the SmartCost system.

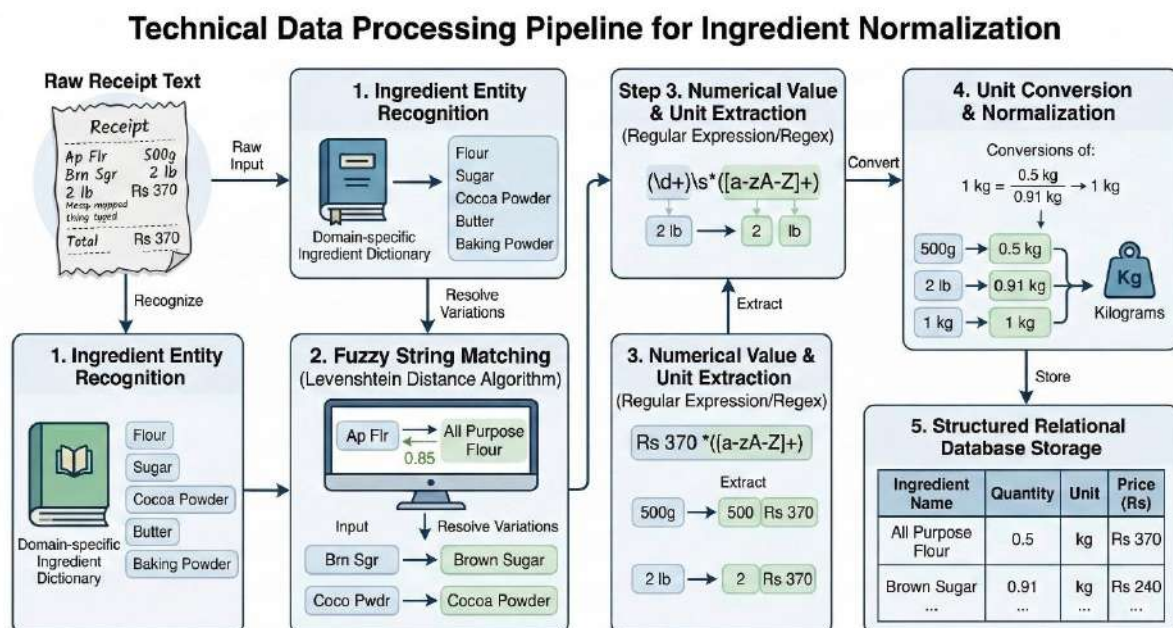


Figure 01

3.3 Tools and Platforms

The development of the Semantic Inventory Normalization Engine requires several programming tools and software platforms. The system will be implemented primarily using the Python programming language, which provides extensive support for text processing and Natural Language Processing techniques.

Several specialized libraries will be used to implement the normalization algorithms. The RapidFuzz or FuzzyWuzzy library will be used to perform fuzzy string matching operations for ingredient name normalization. The re (Regular Expressions) library will be used to identify numerical values and measurement units within receipt text. Additionally, Pandas may be used for managing intermediate data processing tasks and organizing ingredient datasets.

The normalized data will be stored in a relational database system such as MySQL or PostgreSQL, enabling structured storage and retrieval of ingredient records. This database will allow other system components to access standardized ingredient data for inventory tracking and cost analysis.

The development and testing environment will include tools such as Visual Studio Code or PyCharm, which provide integrated development support for Python-based applications. Version control systems such as Git may also be used to manage the source code and track development progress.

3.4 Data Collection and Ethical Considerations

The research requires datasets that represent real-world receipt text containing ingredient purchase information. Data collection will involve gathering sample receipts from grocery stores, supermarkets, and bakery supply vendors. These receipts will be used to create a dataset of ingredient descriptions that include various abbreviations, spelling variations, and measurement units.

In addition to physical receipt data, supplementary ingredient information may be collected from publicly available grocery product catalogs and online retail datasets. These datasets will help construct the domain-specific ingredient dictionary used for normalization.

To protect privacy and ensure ethical compliance, all receipt data used in the research will be anonymized before processing. Sensitive information such as business names, addresses, tax identification numbers, and payment details will be removed or redacted before the data is included in the experimental dataset.

The collected data will be used exclusively for academic research purposes, and the study will adhere to institutional ethical guidelines for data handling and privacy protection.

Category	Description
Data Source	Receipt samples collected from small bakeries, cafés, home-based food businesses, and grocery stores. Additional ingredient information will be gathered from publicly available grocery product datasets and supplier catalogs.
Type of Data Collected	Ingredient names, abbreviations used in receipts, quantities, measurement units, and purchase details extracted from receipt text.
Data Collection Method	Receipt images will be captured or obtained from participating businesses. OCR technology will extract text from these receipts, which will then be processed by the semantic normalization engine.
Purpose of Data Collection	The collected data will be used to build the ingredient dictionary, train and test fuzzy matching algorithms, and evaluate the accuracy of the ingredient normalization process.
Data Storage	All processed ingredient data will be stored in a structured database designed for inventory management and research analysis.
Ethical Considerations	Business-related information such as company names, supplier details, or payment information will be anonymized to protect privacy.
Consent from Participants	Participants providing receipt data will be informed about the purpose of the research and their consent will be obtained before using their data.
Data Security	Access to collected data will be restricted to authorized researchers, and appropriate data protection measures will be implemented to prevent unauthorized access.
Data Usage Limitation	The collected data will only be used for academic research purposes related to the development and evaluation of the Semantic Inventory Normalization Engine.

Table 02

3.5 Validation Strategy and Performance Metrics

The performance of the Semantic Inventory Normalization Engine will be evaluated through experimental testing using the collected receipt dataset. The validation process will measure the accuracy of ingredient recognition, name normalization, and quantity conversion.

Several evaluation metrics will be used to assess system performance. Normalization Accuracy will measure the percentage of ingredient names correctly mapped to standardized dictionary entries. This metric will indicate how effectively the fuzzy matching algorithm resolves variations in ingredient naming.

Entity Recognition Accuracy will measure how accurately the system identifies ingredient-related words within the extracted receipt text.

Additionally, unit normalization accuracy will evaluate the correctness of measurement conversions when transforming quantities from different units into standardized metric values.

Experimental testing will be conducted using a set of labeled receipt samples where the correct ingredient names and quantities are known in advance. The results produced by the system will be compared against these ground truth values to determine the overall accuracy of the normalization process.

The evaluation results will provide quantitative evidence of the effectiveness of the proposed normalization engine and demonstrate its ability to transform noisy receipt text into structured inventory records suitable for automated analysis.

4. HIGH-LEVEL SYSTEM ARCHITECTURE

The proposed SmartCost system is designed as a modular decision support platform that enables small and medium-sized food businesses to digitize receipt data and convert it into structured ingredient information for inventory analysis. The architecture follows a layered structure where raw receipt data is captured, processed through text extraction modules, normalized into structured inventory records, and stored in a centralized database for further analysis.

Within this architecture, the Semantic Inventory Normalization Engine (Member 2 component) plays a critical role in transforming unstructured receipt text into standardized ingredient data. After receipt images are processed through the text extraction stage, the extracted text is passed to the normalization engine, which identifies ingredient entities, resolves abbreviations, and standardizes measurement units. The processed data is then stored in the inventory database where it can be used for cost tracking, inventory monitoring, and further analytical processing.

The system architecture ensures modular separation of responsibilities so that each component can function independently while contributing to the overall system functionality.

4.1 System Overview

The SmartCost system architecture consists of several interconnected modules designed to process receipt data and convert it into meaningful business information. The process begins with the capture of receipt images from users through the mobile application interface. These images are then processed by the text extraction module, which converts the visual information into machine-readable text.

Once the textual data is extracted, the Semantic Inventory Normalization Engine processes the text to identify ingredient names, quantities, and measurement units. The normalization module applies fuzzy matching techniques and domain-specific dictionaries to resolve inconsistencies in ingredient naming conventions. It also converts various measurement units into a standardized metric format to ensure consistency across inventory records.

After normalization, the processed data is stored in a structured database that organizes ingredient information in a consistent format. This structured dataset enables the system to support inventory monitoring and cost analysis functionalities.

The modular structure of the system ensures that improvements to individual components can be implemented without affecting the entire system architecture.

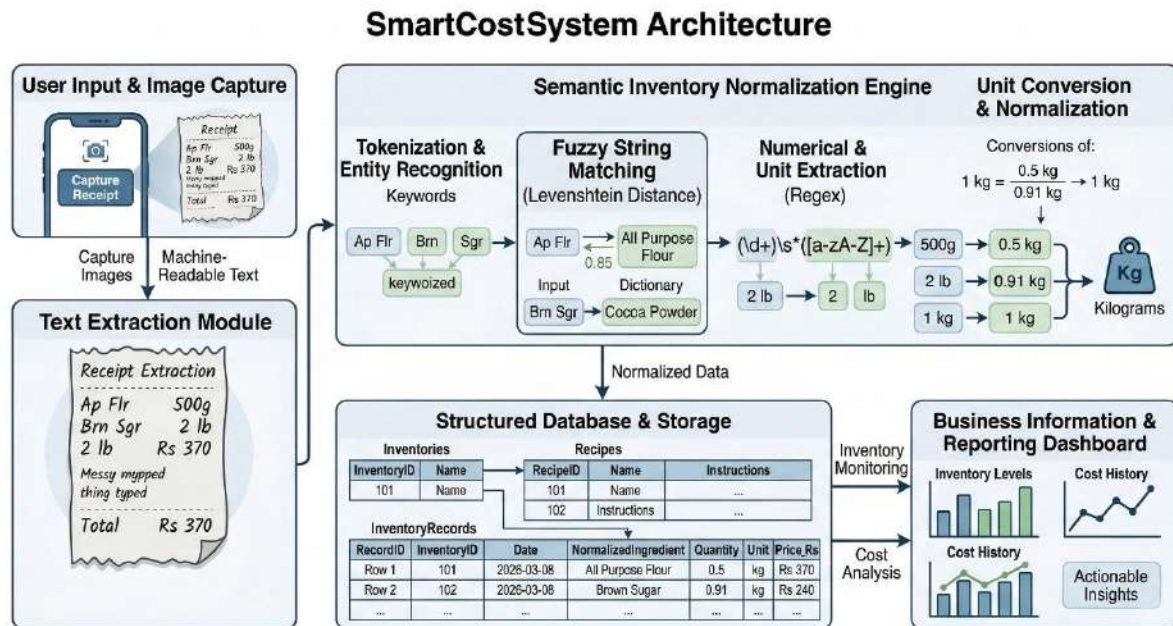


Figure 02

4.2 System Component Diagram

The high-level system architecture consists of the following key components:

1. Receipt Image Capture Module

This module allows users to capture receipt images using a mobile device or upload receipt files directly to the system. The captured images serve as the input data for the processing pipeline.

2. Text Extraction Module

This component uses OCR technology to convert receipt images into raw textual data. The output generated at this stage typically contains unstructured text that includes ingredient descriptions, quantities, and pricing information.

3. Semantic Inventory Normalization Engine

This module processes the extracted text and converts it into structured ingredient data. It performs ingredient recognition, fuzzy string matching, and measurement unit normalization. The module ensures that different textual variations of ingredient names are mapped to standardized inventory labels.

4. Ingredient Database

A structured database stores normalized ingredient information, including ingredient names, quantities, units, and price details. The database enables efficient storage and retrieval of inventory records.

5. Inventory Management Interface

This interface allows users to view and manage ingredient records, track purchases, and analyze inventory usage.

The system component diagram illustrates how these modules interact to transform raw receipt data into structured inventory information.

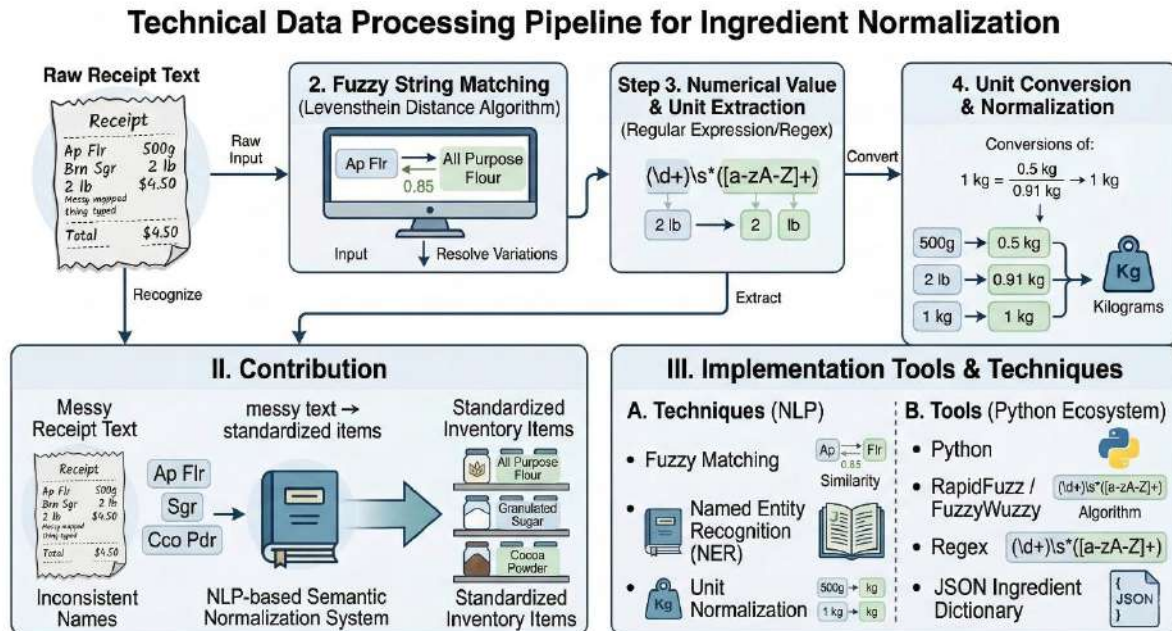


Figure 03

4.3 Component Interactions

The interaction between system components follows a sequential processing pipeline. The process begins when the user uploads or captures a receipt image using the mobile application interface. The image is transmitted to the text extraction module, which converts the visual receipt content into raw textual data.

The extracted text is then forwarded to the Semantic Inventory Normalization Engine. This component analyzes the textual input to identify ingredient-related information. Using fuzzy string matching algorithms and a domain-specific ingredient dictionary, the module maps inconsistent ingredient descriptions to standardized inventory terms.

Once ingredient entities are identified, the normalization engine extracts the corresponding quantities and measurement units. These units are converted into standardized metric values using the unit conversion module. The normalized ingredient data is then packaged into structured records and stored in the ingredient database.

Finally, the structured data becomes available to the inventory management interface, allowing users to view and analyze ingredient purchases efficiently.

4.4 Data Flow Explanation

The system processes data through a structured pipeline designed to ensure accurate transformation of raw receipt text into usable inventory information.

1. Input Stage

Receipt images are captured or uploaded through the mobile application interface.

2. Text Extraction Stage

The receipt image is processed using OCR technology to extract textual information from the image.

3. Normalization Stage

The extracted text is analyzed by the Semantic Inventory Normalization Engine. Ingredient names are recognized using the ingredient dictionary, and fuzzy matching techniques are applied to resolve spelling variations and abbreviations.

4. Unit Normalization Stage

Measurement units associated with ingredient quantities are identified and converted into standardized units.

5. Database Storage Stage

The normalized ingredient information is stored in the structured ingredient database.

6. Output Stage

Users can access the normalized data through the system interface for inventory tracking and business analysis.

This data flow ensures that raw receipt data is systematically processed and converted into structured inventory records.

4.5 Scalability and Security Considerations

Scalability is an important factor in the design of the SmartCost system, as the platform must be capable of handling increasing volumes of receipt data as the number of users grows. The modular architecture allows each system component to scale independently. For example, the normalization engine can be deployed as a separate processing service that can handle multiple text processing requests simultaneously.

The system database can also be expanded to support larger datasets as ingredient records accumulate over time. Cloud-based database solutions may be utilized to ensure efficient data storage and retrieval while maintaining system performance.

Security considerations are equally important, particularly because receipt data may contain sensitive business information. To protect user privacy, the system will implement data anonymization techniques during the preprocessing stage to remove identifiable information from receipts before processing.

Secure communication protocols will be used to ensure that data transmitted between system components remains protected. Access control mechanisms will also be implemented to restrict unauthorized access to the ingredient database and ensure that only authorized users can view or modify inventory records.

By incorporating scalability and security considerations into the system architecture, the SmartCost platform aims to provide a reliable and secure environment for managing ingredient data and supporting small business operations.

5. USER REQUIREMENTS

The SmartCost system is designed to assist small and medium-sized food businesses in managing ingredient purchases and inventory data more efficiently. In particular, the Semantic Inventory Normalization Engine aims to transform unstructured receipt text into standardized ingredient records that can be stored and analyzed within the system.

User requirements define the needs and expectations of the stakeholders who interact with the system. These requirements ensure that the system functions effectively in real-world environments and supports the operational workflows of small businesses. The requirements are categorized into stakeholder identification, functional requirements, and non-functional requirements.

5.1 Stakeholder Identification

The SmartCost system involves several stakeholders who interact with the system in different ways. Understanding these stakeholders is important to ensure that the system design addresses their specific needs.

1. Small Business Owners (Primary Users)

Small business owners such as bakery operators, café managers, and food production entrepreneurs represent the primary users of the system. These users rely on receipt records to track ingredient purchases and manage inventory costs. The system must provide them with accurate and standardized ingredient data extracted from receipts so they can monitor purchases and manage operational expenses more effectively.

2. Store Staff or Employees

Employees who are responsible for purchasing ingredients or recording business transactions may interact with the system by scanning receipts or uploading receipt images. These users require a system that is easy to use and capable of automatically interpreting receipt data without requiring extensive manual input.

3. System Administrators

System administrators manage system configuration, database maintenance, and technical monitoring. They ensure that the ingredient dictionary, normalization rules, and system databases remain accurate and up to date.

4. Software Developers and Researchers

Developers and researchers involved in maintaining or improving the system may use the platform to test algorithms, analyze system performance, and enhance the normalization process. Their feedback helps improve the accuracy and efficiency of the semantic normalization engine.

5.2 Functional Requirements

Functional requirements describe the specific operations and services that the system must provide to users. These requirements define how the system processes receipt text and converts it into structured ingredient records.

FR1 – Receipt Text Input Processing

The system shall accept receipt text extracted from OCR systems and process it as input data for the normalization engine.

FR2 – Ingredient Entity Recognition

The system shall identify ingredient names within the extracted receipt text using a domain-specific ingredient dictionary.

FR3 – Fuzzy Matching for Ingredient Normalization

The system shall apply fuzzy string matching algorithms to resolve variations in ingredient names and map them to standardized ingredient labels.

FR4 – Quantity Extraction

The system shall extract numerical quantities associated with ingredient descriptions within receipt text.

FR5 – Measurement Unit Identification

The system shall detect measurement units such as grams, kilograms, liters, and pounds within receipt text.

FR6 – Unit Conversion and Normalization

The system shall convert different measurement units into a standardized metric format to ensure consistent inventory records.

FR7 – Structured Data Storage

The system shall store normalized ingredient information in a structured database including ingredient name, quantity, unit, and purchase details.

FR8 – Ingredient Dictionary Management

The system shall allow administrators to update or expand the ingredient dictionary to improve normalization accuracy.

FR9 – Data Retrieval and Visualization

The system allow users to retrieve and view normalized ingredient data through the system interface.

5.3 Non-Functional Requirements

Non-functional requirements describe the performance, usability, security, and reliability characteristics of the system. These requirements ensure that the system operates efficiently and securely.

NFR1 – Accuracy

The semantic normalization engine should achieve high accuracy in identifying ingredient names and converting them into standardized inventory records.

NFR2 – Performance Efficiency

The system should process receipt text within a reasonable time frame to ensure that users receive normalized results quickly.

NFR3 – Usability

The system interface should be user-friendly and accessible to small business owners who may not have advanced technical knowledge.

NFR4 – Scalability

The system should be capable of handling increasing volumes of receipt data as the number of users grows.

NFR5 – Security

The system should ensure secure handling of receipt data and protect sensitive business information from unauthorized access.

NFR6 – Reliability

The system should maintain stable performance and provide consistent results during receipt processing and ingredient normalization.

NFR7 – Maintainability

The system architecture should allow developers to update the ingredient dictionary, improve normalization algorithms, and expand system functionality with minimal disruption.

6. COMMERCIALIZATION PLAN

The SmartCost system aims to provide a digital solution for small and medium-sized food businesses to efficiently manage ingredient purchases and inventory records. The Semantic Inventory Normalization Engine, developed as part of this research, plays a key role in enabling automated interpretation of receipt text and transforming it into structured inventory data. By reducing manual data entry and improving accuracy in ingredient tracking, the system provides significant operational value for small businesses that rely heavily on ingredient-based production processes.

The commercialization plan outlines the potential market opportunities, value proposition, revenue model, and pricing strategy for deploying the SmartCost system as a commercial product. The plan also highlights the competitive advantages and intellectual property considerations associated with the proposed solution.

6.1 Target Market and Customer Persona

The primary target market for the SmartCost system consists of small and medium-sized food-related businesses, particularly those that rely on ingredient-based production processes. These include bakeries, cafés, restaurants, catering services, and small food manufacturing enterprises. Such businesses frequently purchase ingredients from suppliers and maintain receipt-based records of these transactions. However, many of these businesses lack digital systems capable of automatically interpreting and organizing receipt data.

The typical customer persona for the system is a small business owner or operations manager responsible for managing ingredient purchases and monitoring business expenses. These individuals often have limited time and technical expertise to manually record and organize receipt information. As a result, they require an automated system that simplifies the process of tracking ingredient purchases and maintaining accurate inventory records.

In addition to individual businesses, the system may also be useful for small business cooperatives, franchise food outlets, and accounting service providers that manage financial records for multiple clients.

Small Bakeries



Mr. Alagaratnam Rasiah

Small Bakery Owner - The Royal Bakery

About

Owner of The Royal Bakery in Colombo, Sri Lanka.

Over 20 years in the bakery business.

Handles ingredient purchasing and daily operations.

Relies on paper receipts for records.

Age: 48

Location: Colombo, Sri Lanka

Experience: 20+ Years

Status: Married

Tech Level: Beginner (2/5)

Goals

- Track ingredient purchases
- Organize inventory records
- Reduce manual bookkeeping
- Improve cost control

Motivations

"Managing ingredient costs is essential for bakery profitability."

"I want a simple system to organize my purchase records."

Concerns

- Complex software is hard to use
- Sorting through paper receipts
- Inconsistent ingredient names

Technology

- Uses a basic smartphone
- WhatsApp for messaging
- Not tech-savvy

Needs

- Easy-to-Use System
- Standardize Ingredient Names
- Digital Inventory Tracking
- Quick Receipt Scanning

Cafés and restaurants



Mr. Kaluhandilage Hemachandra Silva

Café Owner - Butter Boutique

About

Mr. Silva manages Butter Boutique, a café that sells pastries, desserts, and beverages.

The café purchases ingredients from several suppliers, resulting in many receipts with different naming formats for ingredients.

Organizing these records manually is inefficient and error-prone.

Age: 41

Location: Colombo, Sri Lanka

Experience: 12 Years

Status: Married

Tech Level: Intermediate (3/5)

Cafe

Goals

- Monitor ingredient purchases from multiple suppliers
- Maintain consistent ingredient records
- Improve operational efficiency

Motivations

"Better ingredient tracking helps improve cost management."

"A smart system could reduce my administrative workload."

Concerns

- Supplier receipts use inconsistent ingredient naming
- Manual data entry consumes too much time

Technology Usage

- Uses a laptop and smartphone for business
- Familiar with spreadsheets
- Needs simple, automated solutions

Needs

- Automated Ingredient Categorization
- Consistent Record Keeping
- Inventory & Cost Management
- Effortless Receipt Processing

Home-based food businesses



Mr. Wickramasinghe

Home-Based Business Owner - A&M Cupcakes

About

Mr. Wickramasinghe operates A&M Cupcakes, a small home-based bakery specializing in customized cupcakes for events and online orders.

Since he purchases ingredients frequently from supermarkets and suppliers, he struggles to organize purchase records and track ingredient usage efficiently.

Age: 34

Location: Kandy, Sri Lanka

Experience: 5 Years

Status: Married

Tech Level: Intermediate (3/5)

Business Type: Home-Based Bakery

Goals

- Track ingredient purchases easily
- Calculate product costs accurately
- Maintain digital records of receipts
- Reduce manual accounting work

Motivations

"Running a home bakery requires careful cost management."

"A system that reads my receipts automatically would save a lot of time."

Concerns / Pain Points

- Receipts often contain unclear abbreviations
- Difficulty tracking ingredient costs across multiple orders
- Lack of affordable digital inventory tools

Technology Usage

- Uses laptop and smartphone for business
- Relies on email and social media for orders
- Familiar with basic apps, hesitant with complex systems

Needs

- A cost-effective app
- Efficient Digital Receipt Capture
- Ingredient Cost Analysis Tools
- User-friendly Accounting Features

Food manufacturing micro-enterprises



Mrs. Fathima Ayesha

Production Manager - Green Cabin

About

Mrs. Fathima Ayesha works as a production manager at Green Cabin, a small food manufacturing micro-enterprise. She oversees ingredient procurement, production scheduling, and inventory tracking.

Accurate ingredient data is critical for maintaining consistent product quality.

Age: 38

Location: Colombo, Sri Lanka

Experience: 10 Years

Status: Married

Tech Level: Advanced (4/5)

Business Type: Food Manufacturing

Goals

- Maintain precise ingredient inventory records
- Improve procurement efficiency
- Reduce documentation errors

Motivations

"Accurate ingredient data helps maintain consistent product quality."

"Automated data extraction would improve production planning."

Concerns / Pain Points

- Supplier receipts contain inconsistent ingredient labels
- Time-consuming manual verification of ingredient quantities

Technology Usage

- Uses a ERP system for procurement & inventory
- Prefers digital solutions to reduce manual work
- Values accurate, automated tools

Needs

- Efficient Receipt Scanning
- Standardized Ingredient Management
- Integrated Procurement Workflow
- Automated Inventory Accuracy

6.2 Value Proposition

The primary value proposition of the SmartCost system lies in its ability to transform unstructured receipt data into structured and meaningful inventory information. The Semantic Inventory Normalization Engine enables the system to automatically recognize ingredient names, resolve inconsistencies in text data, and convert measurement units into standardized formats.

This functionality eliminates the need for manual data entry and significantly reduces the time required to organize ingredient purchase records. By providing accurate and structured inventory data, the system allows business owners to monitor ingredient usage more effectively, identify purchasing trends, and improve operational efficiency.

Furthermore, the automation provided by the normalization engine helps reduce human errors that commonly occur when manually interpreting receipt information. As a result, businesses can maintain more reliable inventory records and make better-informed decisions regarding ingredient procurement and production planning.

6.3 Revenue Model

The SmartCost system can be commercialized using a Software-as-a-Service (SaaS) subscription model. Under this model, businesses pay a recurring subscription fee to access the system through a mobile or web-based platform.

A tiered subscription model can be implemented to accommodate businesses of different sizes and operational needs. For example, a basic subscription plan may provide limited receipt processing and inventory tracking features suitable for small businesses with low transaction volumes. More advanced subscription plans may offer additional capabilities such as detailed analytics, automated reporting, and extended data storage.

Another potential revenue stream involves offering enterprise integration services for larger organizations that require customized system features or integration with existing accounting software. This approach allows the system to generate revenue through both subscription-based services and customized deployment solutions.

6.4 Cost Estimation

The development and deployment of the SmartCost system involve several cost factors, including software development, system infrastructure, and operational maintenance.

Development costs include expenses associated with software engineering, system testing, and research activities related to the Semantic Inventory Normalization Engine. Additional costs may arise from acquiring cloud infrastructure services required to host the application and store receipt and inventory data.

Operational costs include server maintenance, data storage services, system updates, and technical support for users. These expenses are necessary to ensure that the platform remains reliable, secure, and scalable as the number of users increases.

Despite these costs, the use of open-source technologies and cloud-based infrastructure can significantly reduce initial development and deployment expenses.

6.5 Pricing Strategy

The pricing strategy for the SmartCost system is designed to remain affordable for small and medium-sized businesses while ensuring sustainable system operation. A freemium model may be adopted in which basic system functionality is offered at no cost, allowing small businesses to experiment with the platform before committing to a paid subscription.

Paid subscription tiers may offer enhanced features such as higher receipt processing limits, advanced analytics, and extended inventory management capabilities. This pricing structure encourages user adoption while generating revenue from businesses that require more advanced functionality.

The pricing model will be designed to remain competitive with existing expense management and receipt-processing applications while offering unique capabilities tailored to ingredient-based businesses.

6.6 Competitive Advantage

The SmartCost system provides several competitive advantages compared to existing receipt processing and expense management applications. Most existing systems focus primarily on financial transaction tracking and do not provide specialized tools for managing ingredient-based inventory.

The Semantic Inventory Normalization Engine differentiates the proposed system by introducing domain-specific data interpretation capabilities. Instead of simply extracting receipt text, the system interprets ingredient names, resolves abbreviations, and standardizes measurement units automatically.

This domain-focused approach makes the system particularly valuable for food businesses that rely on ingredient tracking as part of their daily operations. By providing structured ingredient data rather than raw receipt text, the system enables more advanced inventory monitoring and business analytics capabilities.

6.7 Intellectual Property Considerations

The intellectual property associated with the SmartCost system primarily involves the design and implementation of the Semantic Inventory Normalization Engine and its associated algorithms. The integration of domain-specific ingredient dictionaries, fuzzy string matching techniques, and automated unit normalization methods represents a unique technical solution for processing receipt-based ingredient data.

To protect this innovation, the system architecture, algorithms, and software implementation may be documented and registered under appropriate intellectual property protection mechanisms such as software copyright or patent protection, depending on the novelty of the implemented techniques.

Additionally, the system's source code and proprietary datasets used for training and optimization may be protected through licensing agreements and access control mechanisms to prevent unauthorized distribution or replication.

By implementing appropriate intellectual property protection strategies, the SmartCost system can maintain its competitive advantage while supporting potential commercialization opportunities in the software market.

7. BUDGET AND JUSTIFICATION

The development of the Semantic Inventory Normalization Engine requires a combination of hardware resources, software tools, cloud infrastructure, and human effort. Although the proposed system primarily relies on software-based technologies, several resources are necessary to support system development, testing, and deployment. This section outlines the estimated budget required for implementing the Member 2 component and provides justification for each cost category.

The budget estimation focuses on resources required for data processing, algorithm implementation, development environments, database management, and system deployment. Wherever possible, open-source technologies will be utilized to minimize development costs while maintaining system reliability and performance.

7.1 Hardware Costs

Hardware resources are necessary to support system development, data processing, and testing of the Semantic Inventory Normalization Engine. Development tasks such as text processing, dataset preparation, and algorithm evaluation require computing devices with adequate processing capabilities.

The primary hardware requirement for this research includes a development workstation or laptop capable of running programming environments and data processing tools. A typical development system with sufficient processing power, memory, and storage capacity is required to perform text normalization experiments and manage datasets.

In addition to the development workstation, a mobile device may be required for testing system integration with receipt capture applications. The mobile device will allow researchers to simulate real-world receipt scanning scenarios and verify the effectiveness of the normalization engine when processing receipt text.

Since most modern development environments can run on standard computing devices, the hardware requirements remain relatively modest compared to hardware-intensive research projects. Existing personal computing devices may also be utilized to reduce overall hardware expenses.

Item	Quantity	Estimated Cost	Purpose
Smartphone (Android device) test	1	Rs.78,000	Testing receipt capture and mobile application functionality
Laptop / Development Computer	1	Rs.280,000	Software development, model implementation, and system testing
External Storage Device	1	Rs.28,000	Backup of receipt image datasets and experimental results
Internet Router / Connectivity	1	Rs.24,000	Cloud access and API communication testing

Table 03

7.2 Software Licenses

The proposed system primarily relies on open-source software tools and libraries, which significantly reduces the need for expensive proprietary software licenses. The Semantic Inventory Normalization Engine will be developed using the Python programming language due to its extensive support for Natural Language Processing and data processing tasks.

Several open-source libraries will be used during development. These include libraries for fuzzy string matching, text processing, and data manipulation. Because these libraries are freely available, the software licensing cost is expected to remain minimal.

However, optional software tools such as integrated development environments (IDEs) or advanced data analysis platforms may be used to enhance developer productivity. Examples include professional versions of development tools that offer debugging support, version control integration, and performance monitoring features.

Despite these optional tools, most of the system implementation can be completed using open-source software, which helps keep the overall software budget low.

Software Tool	License Type	Estimated Cost	Purpose
Python Programming Language	Open Source	LKR 0	Main programming language used for backend processing
Pandas Library	Open Source	LKR 0	Data preprocessing and receipt text processing
RapidFuzz / FuzzyWuzzy	Open Source	LKR 0	Fuzzy string matching to recognize ingredient abbreviations
Regular Expressions (Regex)	Open Source	LKR 0	Extract ingredient quantities and measurement units
FastAPI Framework	Open Source	LKR 0	Backend API development
React Native	Open Source	LKR 0	Mobile application interface for scanning receipts
Firebase	Freemium / Cloud Service	LKR 0 (Free Tier)	Cloud database and backend services for storing and syncing data

Table 04

7.3 Cloud Services

Cloud services may be required to support system deployment, data storage, and processing scalability. Since the SmartCost system is intended to function as a digital platform accessible to multiple users, cloud infrastructure can provide reliable hosting and storage capabilities.

Cloud-based databases may be used to store normalized ingredient records and receipt datasets. These services allow the system to scale efficiently as the volume of receipt data increases. Cloud storage solutions also provide backup and recovery mechanisms that help ensure data integrity.

Additionally, cloud computing resources may be used for processing large datasets during algorithm evaluation and system testing. Cloud platforms provide flexible computing environments where processing capacity can be increased as needed.

Although cloud services introduce operational costs, many providers offer free or low-cost development tiers that can support research-level deployments during the initial stages of the project.

Cloud Service	Estimated Cost (LKR)	Purpose
Virtual Server Hosting	LKR 15,000	Hosting the forecasting module backend API
Cloud Storage	LKR 8,000	Storage of forecasting datasets and prediction results
API Hosting Infrastructure	LKR 10,000	Deployment of the prediction service API
Network Bandwidth	LKR 7,000	Data transfer and API communication
Estimated Cloud Cost: LKR 40,000		

Table 05

7.4 Human Effort Cost

Human effort represents one of the most significant resources required for developing the Semantic Inventory Normalization Engine. The research project requires time and expertise from researchers, developers, and technical contributors involved in designing, implementing, and evaluating the system.

Development tasks include designing the normalization algorithms, constructing the ingredient dictionary, implementing fuzzy matching techniques, and testing the system using sample receipt datasets. Additional effort is required for debugging the system, refining algorithm performance, and preparing documentation for research reporting.

Although the research is conducted as part of an academic project, it is important to recognize the value of the time and expertise invested by the researchers. Estimating the human effort cost provides an understanding of the overall development investment required to build the system.

The total effort includes activities such as literature review, algorithm development, data preparation, experimental evaluation, and documentation of research findings.

8. WORK BREAKDOWN STRUCTURE (WBS)

Task ID	Task Description	Tools / Technology	Duration
T1	Research on ingredient normalization techniques, NLP methods, and receipt text processing.	Research Papers, IEEE Xplore, Google Scholar	2 Weeks
T2	Collect sample receipt datasets from bakeries, cafés, and food businesses.	Python, OCR tools, Web Data Sources	1 Week
T3	Data cleaning and preprocessing of OCR-generated receipt text.	Python, Pandas, NumPy, Regex	2 Weeks
T4	Develop domain-specific ingredient dictionary containing standardized ingredient names and abbreviations.	Python, Pandas, CSV datasets	2 Weeks
T5	Implement fuzzy string matching algorithm to normalize ingredient names.	Python, RapidFuzz / FuzzyWuzzy, Levenshtein Distance	2 Weeks
T6	Extract ingredient quantities and measurement units from receipt text.	Python, Regex, NLP Techniques	1 Week
T7	Implement measurement unit normalization module to standardize units (grams, kilograms, liters).	Python Conversion Functions, Pandas	2 Weeks
T8	Design database schema to store normalized ingredient records.	PostgreSQL / MySQL / SQLite, ER Diagram Tools	2 Weeks
T9	Testing and validation of normalization engine using receipt datasets.	Python Testing Tools, Pandas	1 Week

Table 06

9. GANTT CHART

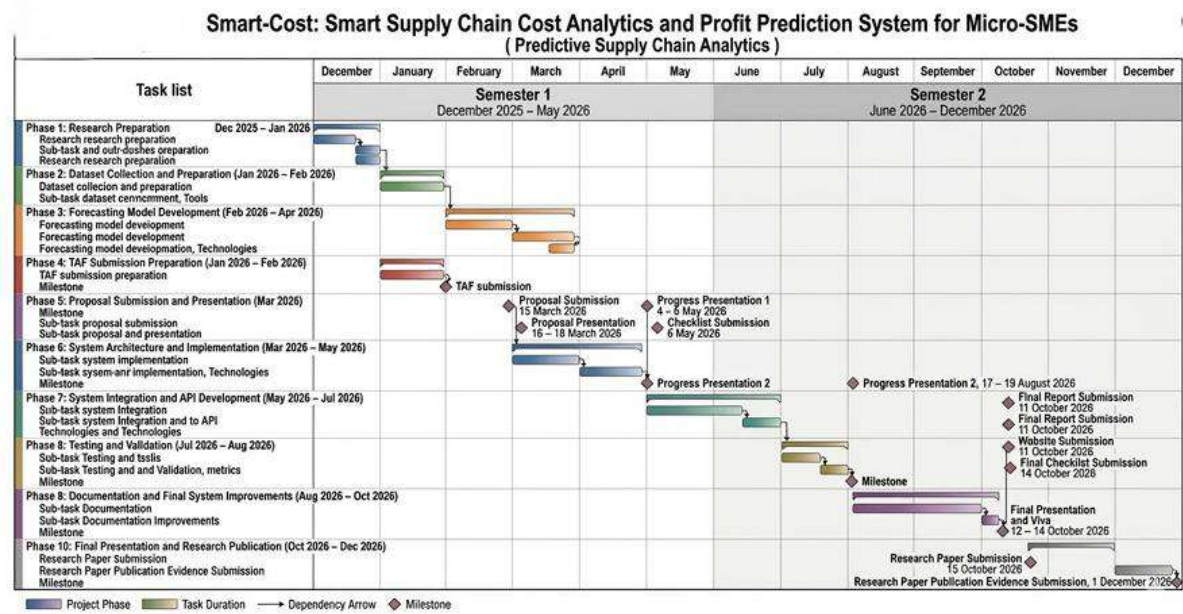


Figure 04

10.REFERENCES LIST

- 1) R. Smith, “An overview of the Tesseract OCR engine,” in Proceedings of the International Conference on Document Analysis and Recognition (ICDAR), Washington, DC, USA, 2007, pp. 629–633.
- 2) D. Jurafsky and J. H. Martin, Speech and Language Processing, 3rd ed. Upper Saddle River, NJ, USA: Pearson, 2020.
- 3) G. Navarro, “A guided tour to approximate string matching,” ACM Computing Surveys, vol. 33, no. 1, pp. 31–88, 2001.
- 4) E. Rahm and H. H. Do, “Data cleaning: Problems and current approaches,” IEEE Data Engineering Bulletin, vol. 23, no. 4, pp. 3–13, 2000.
- 5) C. D. Manning, P. Raghavan, and H. Schütze, Introduction to Information Retrieval. Cambridge, U.K.: Cambridge University Press, 2008.
- 6) H. Cunningham, D. Maynard, K. Bontcheva, and V. Tablan, “GATE: A framework and graphical development environment for robust NLP tools and applications,” in Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, 2002.
- 7) S. J. Russell and P. Norvig, Artificial Intelligence: A Modern Approach, 4th ed. Boston, MA, USA: Pearson, 2021.
- 8) J. Han, M. Kamber, and J. Pei, Data Mining: Concepts and Techniques, 3rd ed. Burlington, MA, USA: Morgan Kaufmann, 2012.
- 9) S. J. Taylor and B. Letham, “Forecasting at scale,” The American Statistician, vol. 72, no. 1, pp. 37–45, 2018.
- 10) S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.

11.APPENDICES

The appendices provide supplementary materials that support the development and evaluation of the SmartCost system. These materials include survey instruments used for stakeholder data collection, technical diagrams illustrating system processes, preliminary user interface designs, risk analysis related to system implementation, and ethical documentation regarding data usage. The appendices serve as additional evidence to demonstrate the research planning, system design considerations, and ethical compliance of the project.

11.1 Appendix A – Survey Instruments

To better understand the operational challenges faced by small food businesses, a preliminary survey will be conducted among bakery owners, café managers, and home-based food business operators. The purpose of the survey is to gather insights into how these businesses currently manage ingredient purchases, store receipt data, and track inventory information.

The survey will consist of structured and semi-structured questions designed to identify common challenges related to receipt management and inventory tracking. The responses collected from this survey will help guide the development of the Semantic Inventory Normalization Engine by identifying the most common ingredient naming conventions, abbreviations, and measurement formats used in supplier receipts.

Example survey questions include:

- How do you currently record ingredient purchases for your business?
- Do you store supplier receipts digitally or manually?
- What difficulties do you face when organizing ingredient purchase records?
- Do supplier receipts often contain abbreviated or inconsistent ingredient names?
- Would an automated system that converts receipt text into structured inventory records be useful for your business?

The survey results will be used to refine the ingredient dictionary and improve the system's ability to interpret real-world receipt data.

11.2 Appendix B – Detailed Diagrams

Several technical diagrams will be used to illustrate the design and operation of the Semantic Inventory Normalization Engine. These diagrams provide a visual representation of the system architecture, processing pipeline, and data flow between different system components.

Key diagrams include:

- System Architecture Diagram – illustrates how the OCR module, normalization engine, database, and mobile application interact within the SmartCost system.

- Data Processing Pipeline Diagram – shows the step-by-step process of converting OCR text into structured ingredient records.
- Algorithm Workflow Diagram – explains the sequence of operations used to detect ingredient names, apply fuzzy matching, extract quantities, and normalize measurement units.
- Database Schema Diagram – illustrates the structure of the inventory database and relationships between stored data fields.

These diagrams help demonstrate how the normalization engine integrates with other components of the system and how data flows throughout the system architecture.

11.3 Appendix C – Preliminary UI Mockups

Preliminary user interface mockups will be developed to demonstrate how users interact with the SmartCost system through a mobile application. These mockups provide an early visualization of system functionality before full implementation.

The UI mockups will illustrate several key screens of the application, including:

- Receipt Scan Screen – allows users to capture images of receipts using the mobile device camera.
- Receipt Processing Screen – displays extracted receipt text and normalized ingredient information.
- Inventory Dashboard – provides an overview of ingredient purchases and inventory records.
- Ingredient Detail View – allows users to review standardized ingredient entries generated by the normalization engine.

These mockups help validate system usability and ensure that the application design aligns with the needs of small business users.

11.4 Appendix D – Risk Analysis

The implementation of the SmartCost system may involve several potential risks related to technical performance, data quality, and system integration. Identifying these risks early allows the research team to develop appropriate mitigation strategies.

Potential risks include:

- Data Quality Issues
 - OCR systems may produce incorrect text outputs if receipt images are unclear or poorly captured. This may affect the accuracy of the normalization engine.
- Inconsistent Ingredient Terminology
 - Supplier receipts may contain unique abbreviations or uncommon ingredient names that are not initially included in the ingredient dictionary.
- Algorithm Accuracy Limitations
 - Fuzzy matching algorithms may occasionally produce incorrect ingredient mappings if similar thresholds are not properly configured.

- System Integration Challenges

Integration between the OCR module, normalization engine, and mobile application may require careful API design to ensure reliable communication between system components.

Mitigation strategies include expanding the ingredient dictionary, adjusting fuzzy matching thresholds, and implementing validation checks within the normalization process.

.

The SmartCost system processes receipt data collected from participating small businesses. Although the system does not handle sensitive personal data, it may process business-related information such as supplier names and purchase records. Therefore, ethical considerations must be addressed during data collection and system testing.

To ensure ethical compliance, the following procedures will be implemented:

- Informed Consent

Business owners who participate in testing or data collection will be informed about the purpose of the research and how their data will be used.

- Data Anonymization

Any identifying information such as business names, supplier identifiers, or transaction references will be removed or anonymized before being used for research purposes.

- Data Usage Limitation

Collected data will be used strictly for academic research and system development purposes.

- Data Security

Appropriate security measures will be implemented to protect stored receipt data and prevent unauthorized access.

These ethical safeguards ensure that the research adheres to responsible data management practices and respects the privacy of participating businesses.

11.5 Ethical Clearance Documents

SLIIT FACULTY OF COMPUTING
Research Ethics Review Form

All students undertaking research in the Faculty of Computing must fill in and submit this form to the respective coordinator / supervisor as instructed by the program head.

SECTION 1

1. Project/research type

☒ B.Sc. ☐ M.Sc. ☐ MPhil/PhD

2. Project/research details

Project ID (if any): R25-IT-026
Research Title: Smart Cost : Smart Supply Chain Cost Analytics and Profit Prediction System for Micro-SMEs

3. Student details (if the project is a group project, insert the details of group leader first)

Student ID	Name	Email	Phone
IT22544872	Kodithuwakku K.A.C.H	IT22544872@my.sliit.lk	0760152847
IT22151810	Perera N.K.M	IT22151810@my.sliit.lk	0769884100
IT22162250	Jayasingha D.D.M	IT22162250@my.sliit.lk	0773055133
IT22152664	Dasanayaka M.P.J.R	IT22152664@my.sliit.lk	071200554

4. Supervisor details

Supervisor Name: Mr. Deemantika Siriwardana
Co-supervisor Name (if any): Ms. Aysha Wijesooriya
External supervisor (if any):

SECTION 2: DATA COLLECTION AND STORAGE

Read the following table before answering the questions.

Data Source	Description	Data Collection Methods
Primary Data	Primary data collection happens when researchers obtain information directly from the original sources.	Survey/Interviews/ Observation/ Focus groups/ New Images etc.
Secondary Data	Datasets already gathered from previous research. This data is usually already analyzed and put into context. Sources of secondary data can include:	Books / Past research papers / Annual reports / Newspapers / Reports / Websites / Online Databases etc.

Page 01

Provide details of all data required for your project. (Indicate what data will be required for each component, where the data is available, and how you plan to collect that data.)

Research Component	Data required. (e.g. Images of leaves)	Amount of Data Required	Data Source (Primary /Secondary)	Collection Method
Adaptive Thermal Thermal Restoration for OCR Based Inventory Digitization	Thermal receipt images from Supermarkets, and small food businesses	2000-3000 receipt images (Open, Valid, damaged)	Primary Data	Direct collection from shops, mobile camera, capture
Semantic Inventory Normalization Engine	Ingredient names, abbreviations, quantities, and measurement units from receipts	2000-3000 receipt images (Open, Valid, damaged)	Primary + Secondary Data	Extracted from collected receipts, manual dataset creation
Hybrid Predictive Supply Chain Volatility Forecasting	Historical ingredient price data (Rice, sugar, oil, onion etc.)	2-3 years historical price records	Secondary Data	Government statistics websites, Kaggle datasets, other datasets
Real-time Receipts & Dynamic Forecasting System	Serialized sales transactions and ingredient purchase records	200-300 simulated business transactions	Primary Data	Generated test dataset and manual data entry

5. DATA COLLECTION: If you require primary data, do you intend to collect data from following groups?

☐ Children or young people
☐ Pregnant women
☐ Individuals with cognitive disabilities
☐ People suffering from specific diseases
☐ Animals
☒ Certain group of people involved in an organization

6. DATA COLLECTION: If you are collecting information related to people, in what form will be the data collected?

☒ Individually identifiable organization
☒ Re-identifiable* Data indirectly leads to the identification of a person or organization (even through direct identifiers are removed)
☐ Non-Identifiable Data does not lead to the identification of a specific individual
*Anonymized data from which the individual could be discovered

7. DATA COLLECTION: Does your project possibly cause/invoke any of the following?

☐ Physical stress/distress, or discomfort to participants
☐ Psychological/mental stress/distress, or discomfort to participants
☒ Collection of sensitive/confidential data of individuals
☒ Collection of confidential data of organizations
☐ Development of a product harmful for a particular group or society

Page 02

8. DATA COLLECTION: Explain the procedures you will be following to ensure privacy of participants or organizations during data collection, data analysis and publishing of results.

☒ Before the data collection, participants will be given an information sheet to create awareness
☒ Data will be collected only from the participants who have given consent through a consent form
☒ Personally identifiable data will not be collected
☒ If identifiable data is present in the data set, data will be either anonymized or coded

Additional Measures:
The research will primarily collect receipt images and publicly available price datasets, which do not contain personal information. If any business-related information appears on receipts (such as shop names or addresses), it will be removed or anonymized during preprocessing. All collected data will be used strictly for academic research purposes.

9. DATA STORAGE: If you are collecting information related to people, in what form will the data be stored?

☐ Individually identifiable ☐ Re-identifiable ☒ Non-Identifiable

10. DATA STORAGE: Where will data be stored and what measures will be taken to ensure security of data?

☐ Paper-based data stored in locked filing cabinets within SLIIT.
☒ Data stored in hard-drives and computers will be protected by password and encryption.
☒ Data will be anonymized and will not be re-identifiable.
☐ After the completion of the project, all data will be permanently deleted from all student devices. Only a single copy will be securely stored within a repository in the institution.

Additional measures:
All research data including receipt images, datasets, and experimental results will be securely stored in password-protected computers and cloud storage. Access to the data will be limited only to the research team members and supervisors. The datasets will be anonymized and used strictly for academic research purposes.

11. How would you rate the overall risk associated with the research? (These risks may occur when the collected data is exposed to unauthorized parties, etc.)

☐ High Risk ☒ Low Risk ☐ Negligible Risk

Page 03

12. Declaration of the participants.

I/we certify that the details given above are accurate and agree to adhere to the conditions of approval specified by the panel. Furthermore, I/we will inform the FoC Research Project coordinator of any deviation from the specified details.

Student ID	Name	Signature
IT22544872	Kodithuwakku K.A.C.H	
IT22151810	Perera N.K.M	
IT22162250	Jayasingha D.D.M	
IT22152664	Dasanayaka M.P.J.R	

13. Approval of the Supervisor

Supervisor Recommendation (please select one option): ☐ Ethics committee Approval NOT Required ☐ Ethics committee Approval Required

If approval required, clearly specify the reasons:

Supervisor Name: Supervisor Signature: Date:

**** Please note that if the supervisor has recommended that ethics committee approval required, student should fill the 'Ethics Review Form' of the SLIIT ERC and submit it to the faculty Ethics Sub-committee.**

Page 04