COME-Invoicing: Digital Solution for Invoice Management in Small Businesses

BY

WONG SUM HUI

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF COMPUTER SCIENCE (HONOURS)

Faculty of Information and Communication Technology

(Kampar Campus)

FEBRUARY 2025

COPYRIGHT STATEMENT

This Final Year Project report is submitted in partial fulfillment of the requirements for the degree of Bachelor of Computer Science (Honours) at Universiti Tunku Abdul Rahman (UTAR). This Final Year Project report represents the work of the author, except where due acknowledgment has been made in the text. No part of this Final Year Project report may be reproduced, stored, or transmitted in any form or by any means, whether electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of the author or UTAR, in accordance with UTAR's Intellectual Property Policy.

ACKNOWLEDGEMENTS

I would like to express my heartfelt gratitude to my supervisor, Ts. Dr. Chai Meei Tyng, for providing me with this valuable opportunity and for her insightful guidance and support throughout this project. Her expertise and encouragement have been instrumental in my growth and learning.

Lastly, I extend my deepest thanks to my parents and family for their unwavering love, support, and encouragement, which have sustained me throughout this journey. Their belief in me has been my greatest motivation.

ABSTRACT

The COME-Invoicing application aims at helping hawkers and small businesses in Malaysia to survive and assimilate in the government adopted e-invoicing system. This mobile application solves problems like low levels of technology adoption, compliance with stringent e-invoicing rules, and the lack of automation in sales tracking. Thus, the application allows users with low IT literacy to better navigate the recording of daily sales, generate automatic reports for the month, and even receive real-time information about what measures they need to fulfil requirements imposed by the government. The app also features sophisticated functionalities such as Optical Character Recognition (OCR) which utilizes the Artificial Intelligence technology to digitize handwritten and the voice to text feature that enables capturing of sales data from spoken words. Combined with multilingual support and the ability to educate users of the application, these attributes make a product comprehensible and easy to navigate for small business owners and managers. The proposed solution not only improves the performance and compliance of businesses but also assists the users to transform and learn about Artificial Intelligence and digitization to succeed the existing digital economy of Malaysia. The use of the COME-Invoicing app itself also conforms with Agile development, enabling constant adjustments in the features and designs based on users' requirements and shifts in regulations. Therefore, the COME-Invoicing application which will be developed through this project will try to support small business and overcome the digital divide in Malaysia.

Area of Study: Software Development & Design
Keywords: Mobile Application, Sqlite3, Application Programming Interfaces, Google Firebase, E-Invoicing

TABLE OF CONTENTS

APPENDIX A

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AI	Artificial Intelligence
AI-OCR	Artificial Intelligence Optical Character Recognition
API	Application Programming Interface
CRM	Customer Relationship Management
ERP	Enterprise Resource Planning
HDD	Hard Disk Drive
ICTs	Information and Communication Technologies
IDE	Integrated Development Environment
IRBM	Inland Revenue Board of Malaysia
JIT/AOT	Just-In-Time / Ahead-Of-Time
LHDN	Lembaga Hasil Dalam Negeri (Malaysia's Inland Revenue Board)
LHDNM	Lembaga Hasil Dalam Negeri Malaysia
OCR	Optical Character Recognition
OS	Operating System
PDPA	Personal Data Protection Act
QR	Quick Response
SDRAM	Synchronous Dynamic Random-Access Memory
UML/ERD	Unified Modeling Language / Entity-Relationship Diagram
UOM	Unit of Measure
UX	User Experience

CHAPTER 1

Introduction

Historically, small businesses and hawkers in Malaysia did not have access to simple point of sale systems, they mostly used manual and traditional approaches to control their financial transactions [1]. Some of the most common types of paper-based record keeping include handwritten cash receipts, use of registers and journals among others. These methods are easy to use and do not need much preparation. However, they possess several weaknesses and vices such as errors, data loss, and lack of scalability and transparency to address new business expansion. Furthermore, the paperwork in traditional bookkeeping poses barriers to tax compliance and, in effect, could raise problems with the authorities or opportunities for deductions that may not be exploited.

Over the last few years, the Malaysian government has emphasized the improvement of the methods of tax reporting and collection, the result of which was the introduction of the Goods and Services Tax (GST) in 2015 [2]. While GST was later replaced by the Sales and Services Tax (SST) [3], such a move further demonstrated the government's sincerity in addressing the issue of simplification of tax reporting for companies small or big. It also forms part of this initiative to gradually roll out e-invoicing which is a technique of electronic generation and exchange of invoices in a structured format between buyers and sellers.

E-invoicing is advantageous in several ways than when it is done the conventional way. It helps in the ministry of tax requirements, revenue, minimize the chance of mistakes, standardizes and enhances the quality of information, and also assist in auditing and preparing of financial statements [4]. Another benefit that arises from digitization is the cutting of costs and time that may be used in data entry and paper works [4].

However, the problem arises in the case of small businesses like hawking businesses because large corporations and medium-scale businesses adapt well to this digitization due to the availability of resources, digital competency, and inclination towards digital approaches as opposed to manual ones. These challenges have however limited the implementation of e-invoicing particularly in the informal economy segments.

Therefore, COME-Invoicing addresses these challenges by providing a mobile-based solution tailored specifically for hawkers and small businesses in Malaysia. The app aims to bridge the gap between traditional bookkeeping practices and the new e-invoicing system.

COME-Invoicing is a mobile application interface predominantly developed for hawkers and small businesses in Malaysia. The introduced application will also help to overcome the existing issues related to the integration of the new e-invoicing system as it will include options to record daily sales in a very convenient way, as well as generate monthly summary reports which will be automatically generated and sent to the user. The application should also include information on the latest trends in the government e-invoicing system and its updates that appear from time to time. The application will also be intuitively navigable to make it available for the users with low levels of digital literacy and the application will also be in multiple languages. Other than that, the application also adopts the use of AI-powered OCR for scanning and converting handwritten to text. The customers can capture images of their handwritten writings or photo and through the use of AI, the sales will be sorted into the appropriate categories from this converted digital text. Finally, the application includes a voice recording feature that allows users to capture audio.

1.1 Problem Statement

Digital Literacy Gap Among Hawkers and Small Business Owners

Malaysian hawkers and small businesses face low levels of digital skills resulting from their low awareness of current technology. The new government regulation on e-invoicing presents unique problems for these people who wait for employment trough manual documentation and dealings in cash. Existing e-invoicing software can be complex work from home mothers might struggle to navigate them and this can lead to noncompliance with the regulations and business operations may also be slowed. But, small business owners want a tool that can assist them in migrating from traditional invoices to digital invoices while not disrupting their operations heavily.

Compliance Challenges with E-Invoicing Regulations

It is evident that with the application of e-invoice in Malaysian context, hawker and small business owners struggle meet the regulation. E-invoicing rules are quite fluid and can change often because they are governed by a range of authorities depending on the country and firms do not always receive the updated detailed guidelines on when to implement these changes. The failure to adhere to these standards will result in legal consequences which will put more pressure on these small businesses. The impact of poor compliance with e-invoicing rules requires an adequate solution, which will enable small business owners to quickly and easily follow all the necessary requirements while negating the need for highly technical knowledge or the input of large amounts of capital.

Inefficiency in Manual Sales Tracking and Reporting

Hawkers and small business owners in Malaysia usually record the daily sales manually and, thus, experience challenges such as inaccuracy, time-consuming, and the inability to integrate with contemporary systems. By the same token, these practices do not enable the creation of accurate records, monthly summaries or e-invoicing compliance. The lack of a well-connected and efficient solution that is completely manual in this case impacts their financial planning and utilization of technology within an invoice environment. The lack of a suitable application poses a pressing challenge to streamline the process of tracking all sales, automate the monthly report as well as connect it with the e-invoicing system to enhance organizational efficiency and business outcomes.

1.2 Motivation

The motive of developing the COME-Invoicing application is the urgent need to support hawkers and small business owners in Malaysia to transform into the digital economy. The Malaysia government seeks to advance the usage of e-invoicing systems sensitive the inadequate knowledge on e-invoicing among these groups. Most of the hawkers and the other small business people still use the traditional approach manually to record their sales which is captivating with several drawbacks. This is a traditional approach that has implementations which are hard to be followed in the new e-invoicing standards thus exposing such businesses to legal consequences of the fines or even financial losses. The absence of simple and efficient tools adapted for the aims of those who are not very familiar with the internet contributes to the strengthening of this issue. Existing solutions of electronic invoicing are cumbersome and expensive and mostly developed for large companies while small businesses lag far behind in getting the benefits. Such a state of affairs means that there is a danger to non-compliance of the law, which may lead to legal and financial consequences and bring additional pressure on these enterprises with weak economy.

To address these challenges, the COME-Invoicing application's goal is to develop an application to ensure that the shifting to digital invoicing was easily understandable by small business owners. Besides, the COME-Invoicing app is easy to use even when the users have a low level of digital literacy. There are different facilities, such as AI-OCR to capture handwritten, you can record data in voice mode. The monthly report is generated automatically. They also improve working activities while also focusing on government compliance, with little to no need for high technicality. Also, due to simplification and focus on the localization of the interface, the application strengthens people's digital literacy and makes them more proactive in front of technologies. This not only assists them to cross the digital divide but also assists them to have improved financial status to manage their businesses and compete at world market more effectively.

1.3 Project Objectives

To Promote Digital Literacy

- Education and awareness creation needs to be provided to the hawkers and the owners of the small businesses on how to use the application as well as understanding of its functionalities.
- Providing, in the app, educational content and guides that will help the users get the general picture of e-invoicing.

To Facilitate Compliance with E-Invoicing Regulations

- Implement a solution to send e-invoices friendly with the Malaysian government framework.
- Inform Malaysian government's e- invoicing current event/developments and offers updates about the e- invoicing.

To Simplify Sales Tracking

- Design a usable interface of the hawker's daily transaction to ensure the ease of data recording and collection.
- Allow for summery privileges that can generate the summaries on recorded sales on a monthly basis.

1.4 Project Scope and Direction

Manage Log In

a. The system shall give users sign in into their account, this will be done using their Google email and password.

b. The system shall warn the users when enters an incorrect Google password or email, the system will inform them.

Manage Register

a. The system shall allow users to register for an account by signing up using their Google email and password.

b. The system shall send a confirmation email to the user's Google email address to verify their registration.

Manage Stock

a. The system will provide functionalities to manage and track inventory, allowing users to input, update, and monitor stock levels efficiently.

Manage Customer

a. The system will provide functionalities to manage customer information, allowing users to input, update, and view customer details efficiently.
b. The system will allow users to add, edit or delete customer entries as needed to maintain an up-to-date database.

Manage Catalog &Cart

a. The system will allow users to add, remove, and adjust quantities of products in their shopping cart.
b. The cart will display item details, quantities, and prices for easy review before checkout.
c. Users can proceed to checkout from the cart, where the system will calculate the total cost, including any applicable taxes and discounts.
d. The system will provide a clear summary of the purchase before the final confirmation, ensuring users can make adjustments as needed.

Manage Receipt

a. The system will offer users the ability to create, manage, and store receipts, helping them maintain proper records of transactions.

Manage Sales Tracking

a. The system shall allow users to record daily sales transactions easily.
b. The system shall automatically generate monthly sales summaries for financial management.

Manage E-Invoicing

a. The system shall generate the QR code allow users to scan the QR code for e-invoice validation.

Manage Voice Recorder and Audio-to-Text Conversion

a. The application shall include a voice recording feature that allows users to capture audio.
b. The application shall have the capability to convert recorded audio into text format.
c. The audio-to-text conversion feature shall support multiple languages commonly used in Malaysia, including Malay, English, Chinese, and Tamil.
d. The converted text shall be editable within the application for accuracy and user customization.

Manage AI-Powered OCR for Handwritten

a. The application shall integrate AI-powered OCR technology to scan and convert handwritten into digital text.
b. Users shall be able to capture images of handwritten writings using their mobile devices.
c. The AI shall automatically sort converted text into appropriate sales categories based on recognized keywords and patterns.
d. The OCR feature shall support multiple languages commonly used in Malaysia, ensuring accuracy and usability across diverse user bases.

Manage Stock Aging

a. The system will allow users to view the stock aging
b. The system will allow users to download the stock aging report in excel format.

Manage Notification

a. The system will notify users if the stock in low quantity.

1.5 Contributions

The proposed application is urgent to bring significant changes and improvements to the lives of hawkers and owners of small businesses due to compliance with e-invoicing regulations, minimum penalties, and legal risks. It leads to efficiency in the sales function by automating it, cutting time and diminishing mistakes, ensuring that owners can concentrate on core, value adding jobs. Also, it enhances the record keeping of the money transactions that allows for better controls in sales and expenses which in turn enhances decision making and even planning. By enhancing digital literacy the app enables the users to engage oneself into technology, overcome the digital divide and secure the financial stability and thereby supporting the Malaysian economy in global market competition. Thanks to the application's targeted analysis, those who may have limited experience with using the Internet will be able to have full access to the resource and gain localized support, and receive actual information regarding government changes in legislation that contribute to the successful functioning in sustainable development of small businesses' activity.

1.6 Report Organization

The details of this project are shown in the following chapters. The details of this project are outlined in the following chapters. Chapter 2 presents a comprehensive overview of the e-invoice requirements, an analysis of similar applications, and a review of e-invoice regulations and rules. In Chapter 3, the system architecture is introduced, along with key design components such as the activity diagram, use case diagram, and both

software and hardware requirements necessary for implementing the system. Chapter 4 describes the preliminary work undertaken, including steps for setting up the program, implementing Firebase authentication, configuring voice recognition, and showcasing the user interface and completed sections of the application.

CHAPTER 2

Literature Reviews

2.1 Overview of E-invoice

An e-Invoice consists of transaction information between a supplier and a buyer and is electronic version of invoice, credit note and debit note [1]. These include the supplier and buyer details, item description, quantities, price before tax, taxes, and total amount which are crucial for the daily business operations. Thus, e-invoices play the role of facilitating documentation of transactions which is more efficient and transparent.

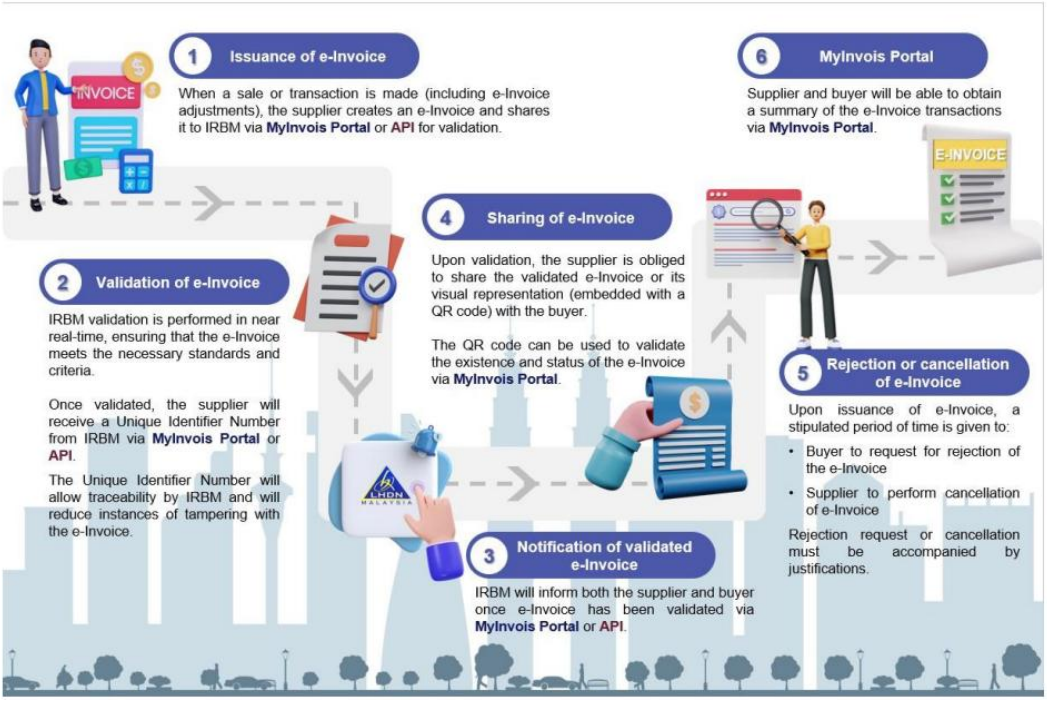


Figure 2.1 Workflow of e-Invoice. [5]

There are several benefits that e-Invoicing brings to businesses and tax authorities as indicated below. It helps to minimize manual work and mistakes and it improves tax filing system as well as other organizational activities. Thus, through digitalization of taxes and financial statements, e-Invoicing optimizes the processes, enhances the compliance with taxes, and enables the business to work on the core business [4]. There are four major types of e-invoice transactions these are invoices, credit notes, debit notes, and refund notes.

To ease the transition to e-Invoicing, Malaysia offers two primary transmission mechanisms, allowing taxpayers to choose the one that best fits their business needs [6]:

1. **MyInvois Portal**: Hosted by the Inland Revenue Board of Malaysia (IRBM), the webpage is easily available and accessible to all these tax payers, especially those with inability to make an API link.
2. **Application Programming Interface (API)**: For instance, for large companies which engage in frequent transactions, an API ensures that the MyInvois system receives data from and transfers data to the taxpayers' systems. This method involves a one-time cost of technology and system adjustments hence suitable for those companies who wants a smooth integration. The rollout of e-Invoicing in Malaysia will be phased in, allowing businesses to gradually transition based on their revenue thresholds.

The rollout of e-Invoicing in Malaysia will be phased in, allowing businesses to gradually transition based on their revenue thresholds:

Table 2.1: Implementation Date of e-Invoicing. [7]

DATE	Annual Turnover or Revenue
01 August 2024 (Phase 1)	> RM 100 million
01 January 2025 (Phase 2)	between RM 25 million to RM 100 million
01 July 2025	between RM 500,000 to RM 25 million
01 January 2026	Up to RM500,000

2.2 Impacts of E-Invoicing

E-Invoicing mainly affects large business and SMEs mainly in operational efficiency, compliance, and data processing. The e-Invoicing is also beneficial to the large business organizations by helping them to automate their invoicing systems hence cutting down on their costs and eliminating any possibilities of making mistakes [8]. The automation also helps in improving the governance as it provides better control over invoicing, its compliance and also reduces the risks of frauds. Big companies are helped by faster payment cycles that enhance both cash flow and sellers' interactions, whereas accurate business data enhances decision-making and planning.

In the case of SMEs, the e-Invoicing has the ability to enhance tax compliance since the transactions are recorded promptly in real time hence minimizing reporting leakages and preventing taxes as a result of stray mistakes [8]. It also covers for shortcomings in certain tax structures such as SST or GST by guaranteeing the right computation of tax [8]. E-Invoicing also helps in reducing the manual work required in audits due to proper Data trail and also helps in proper collection of business data for better financial analysis [8]. It provides real time accounts to SMEs which do not require extra manpower and also minimizes the human intervention leading to decrease in manual errors and hence improved business image and better relations with the customers / suppliers and financial institutions [8].

2.3 Discussion on Government E-Invoicing Solutions

2.3.1 MyInvois Portal

MyInvois Portal is the part of the e-Invoicing system in the Malaysian government aimed to help businesses of any size to adhere to all the requirements [9]. This platform provides an accessible, user-friendly environment for generating, submitting, and managing e-Invoices, particularly for businesses without access to API integration [10].

Features and Capabilities

The MyInvois Portal offers multiple functionalities for taxpayers, allowing them to individually create e-Invoices through comprehensive forms or use batch processing by uploading a spreadsheet for multiple transactions. It is especially relevant for small businesses that are not capable of using API connection but since they are required to meet the e-Invoicing standards [5].

Accessible to all taxpayers, the portal allows users to:

- Generate, submit, view, cancel, or reject invoices.
- Perform a search of historical e-Invoices and analyse them.

Taxpayers must log in to the MyTax Portal to access the MyInvois Portal and fulfill their e-Invoice obligations according to the rules set by the Inland Revenue Board of Malaysia (IRBM).

Pre-Submission and Submission Process

Since there are pre submission activities that need to be accomplished before a tax invoice can be submitted electronically through the MyInvois Portal, activities such as retrieving of the taxpayer identification number through the MyTax Portal or the e-Daftar platform. Once authenticated the supplier can generate and send an e-Invoice to IRBM for checking or verification [5].

The MyInvois Portal offers two options for creating e-Invoices

- **Individual Creation**: By using a form, taxpayers will be able to enter all the required information needed.
- **Batch Upload**: Of course users are allowed to upload the data in a predefined format such as a Microsoft Excel spreadsheet and process many records at once.

Validation and Notification

Upon submission of an e-Invoice, IRBM verificate immediately nearly in real-time. If verified, the supplier receives a verified e-Invoice which has the IRBM Unique Identifier Number making it easy to track and also minimizing the chances of tampering. If unvalidated, an error message is produced and the supplier has to edit the error before sending a proper e-Invoice [5].

Once the e-Invoice is validated, both the supplier and the buyer get notified through the portal and the supplier is expected to make available to the buyer either the validated e-Invoice or in electronic form of the invoice. For the time being, suppliers can provide either format to the buyer in order to save paper [5].

Rejection and Cancellation

The system also encompasses the feature where an e-Invoice can be rejected or cancelled within 72 hours after validation. In the case of errors, the buyer can reject thru MyInvois Portal, while the supplier can cancel it. If no action is taken in 72 hours, correction is possible only through the issuance of a new credit note, debit note or refund note [5].

Storage and Reporting

However, it is important to observe that validated e-Invoice is archived in the database of IRBM, while any business has to maintain its records. This also includes the dashboard services whereby taxpayers can obtain the e-Invoice information such as the date of invoice, amount and status in formats entitled xml, json, pdf or meta data grid view. This strong system instills compliance and cross-checks on breaches, increases visibility and assists businesses in the e-Invoice requirements easily [5].

2.3.2 Application Programming Interface (API)

The Malaysian government's e-Invoicing system allows for seamless integration with taxpayers' Enterprise Resource Planning (ERP) systems through API. This enables businesses to submit e-Invoices directly to the Inland Revenue Board of Malaysia (IRBM). The system also supports submission via Peppol and non-Peppol technology providers, ensuring accessibility for a variety of businesses [5].

Transmission Methods

- **Direct Integration with MyInvois System**: It is also significant that the MyInvois System can be connected to a taxpayer's ERP system so that e-Invoice submission and management can be done automatically.
- **Peppol Service Providers**: Some companies and organizations would like to translate the e-Invoice in Peppol-formats and submit them through pre-approved service providers that guarantee the security and conformity with Peppol-standard.
- **Non-Peppol Technology Providers**: Peppol is not the only means of transmitting e-Invoice and other technology providers can enable e-Invoice transmission to be flexible among firms that are not using Peppol.

API Structure and Formats

- **Extensible Markup Language (XML)**: XML is broadly utilized for expressing structural information and has accurate syntactic rules.
- **JavaScript Object Notation (JSON)**: As a one among the data format family, JSON is lighter model for data interchange compared with the XML model and is easier to use. It is language independent and this means that businesses can use JSON in any programming language.

Required Data Fields

To issue an e-Invoice, 55 data fields are required, grouped into eight categories [5]:

Figure 2.2: 55 Data Fields for e-Invoice. [11]

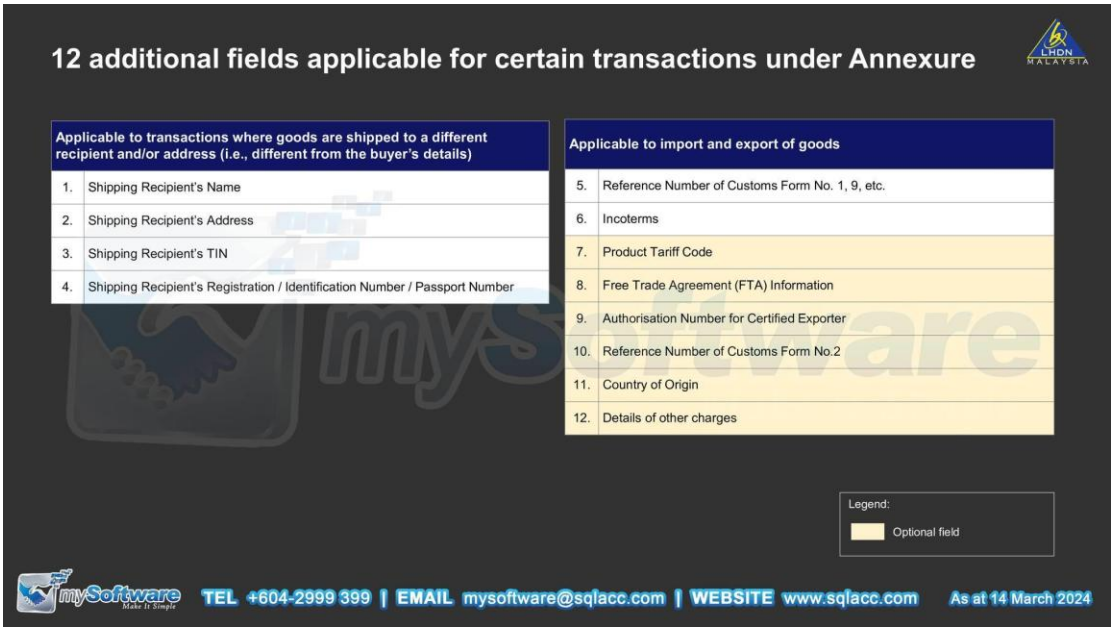


Figure 2.3: 12 Additional Data Fields for e-Invoice [11]

Pre-Submission Requirements

- **Digital Certificate**: A digital certificate (i.e. . cer or . pfx) of the issuing taxpayer has to be incorporated in every e-Invoice. This certificate is then used to establish a digital signature, as a means to validate the e-Invoice.
- **e-Invoice Preparation**: As a result of that, the taxpayers have to set up their systems in a way that will produce e-Invoices in XML or JSON format in further UBL2. 1 structure thus guaranteeing all the require fields in the table being featured.

To get an understanding of how to submit e-invoices, the following are steps involved [5]:

- **Submission**: After the transaction has been effectuated, the taxpayer or more particularly, the service provider can issue an e-Invoice in the format of XML/JSON as deemed appropriate. It is then sent to the IRBM for validation through API to provide an e-Invoice. Accuracy of the information provided rests on the part of the supplier.
- **Validation**: The MyInvois System certifies the e-Invoice in near real time. Upon successful validation, an API response is sent to the supplier or technology provider, containing: **IRBM Unique Identifier Number**, the **date and time validation** was conducted a track back **link to validate.** They also have a tag link for traceability. If the errors are present, an error message is given and the supplier has to reconcile and re-issuing the e-Invoice.

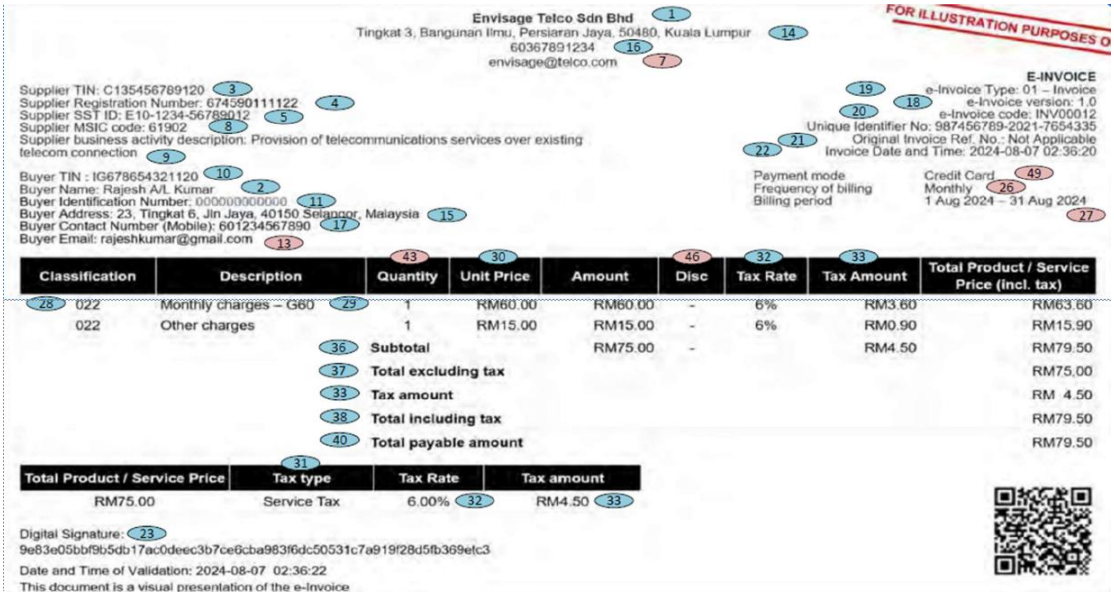


Figure 2.4: 55 Sample of e-Invoice. [11]

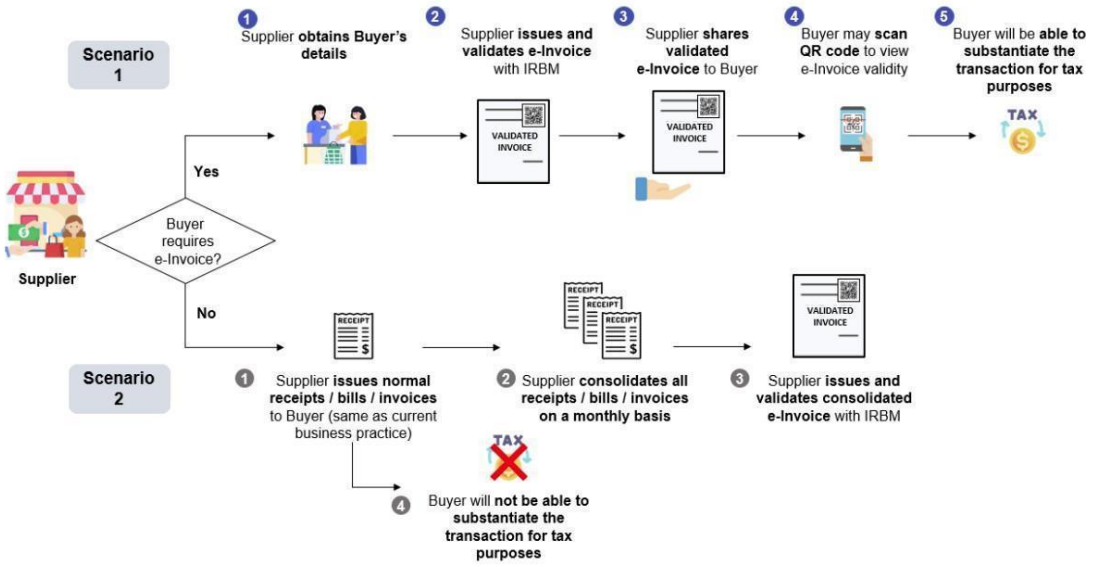


Figure 2.5: Process of e-Invoice. [11]

2.3.3 Receipt of E-invoice

The e-invoice system implemented by POPULAR enables customers to execute seamless and efficient requests for validated invoices through a single platform for all in-store and online acquisitions. Customers can reach the e-invoice request platform through the URL https://einvoice.popular.com.my. Customers must either scan the QR code located on their official store receipt for purchases or follow the link provided on their online delivery notice for e-invoicing purposes. Figure 2.6 show the sample store receipt of Popular Book Co. (Malaysia) Sdn. Bhd.

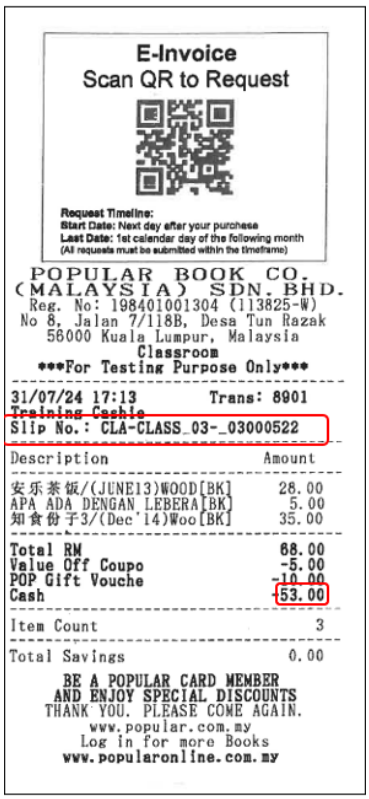

E-Invoice
Scan QR to Request
Request Timeline:
Start Date: Next day after your purchase
Last Date: 1st calendar day of the following month
(All requests must be submitted within the timeframe)
POPULAR BOOK CO.
(MALAYSIA) SDN. BHD.
Reg. No: 198401001304 (113825-W)
No 8, Jalan 7/118B, Desa Tun Razak
58000 Kuala Lumpur, Malaysia
Classroom
For Testing Purpose Only
31/07/24 17:13 Trans: 8901
Training Cashie
Slip No.: CLA-CLASS_03-_03000522
Description Amount
安乐茶饭/(JUNE13)WOOD[BK] 28.00
APA ADA DENGAN LEBERA[BK] 5.00
知食份子3/(Dec'14)Woo[BK] 35.00
Total RM 68.00
Value Off Coupo -5.00
POP Gift Vouche -10.00
Cash 53.00
Item Count 3
Total Savings 0.00
BE A POPULAR CARD MEMBER
AND ENJOY SPECIAL DISCOUNTS
THANK YOU. PLEASE COME AGAIN.
www.popular.com.my
Log in for more Books
www.popularonline.com.my

Figure 2.6 Sample Store Receipt of Popular Book Co. (Malaysia) Sdn. Bhd.

To ensure timely processing, customers should make e-invoice requests within the established time period. Clients must submit e-invoice requests starting at 10:00 AM the next day after buying but all requests must be submitted by 11:59 PM during the first calendar month after purchase. Customers who buy on August 1, 2024 can submit their invoice requests from August 2, 10:00 AM until September 1 at 11:59 PM. The August 31 purchase imposes time restrictions on requests which must occur on September 1 during the 10:00 AM to 11:59 PM period. The system rejects all requests

that arrive after the designated period since the September 2 12:05 AM inquiry falls outside the acceptance window.

To make use of POPULAR's E-Invoice Request Form users need to provide particular information. Every request for e-invoicing must contain the customer's or company's name together with their NRIC or BRN identification number followed by email address and phone number and Taxpayer Identification Number (TIN). All SST-registered companies need to submit SST numbers, but individual users can request SST numbers without providing them. Foreign customers need to select "non-Malaysian" before adding their valid passport number and General TIN "EI00000000020."

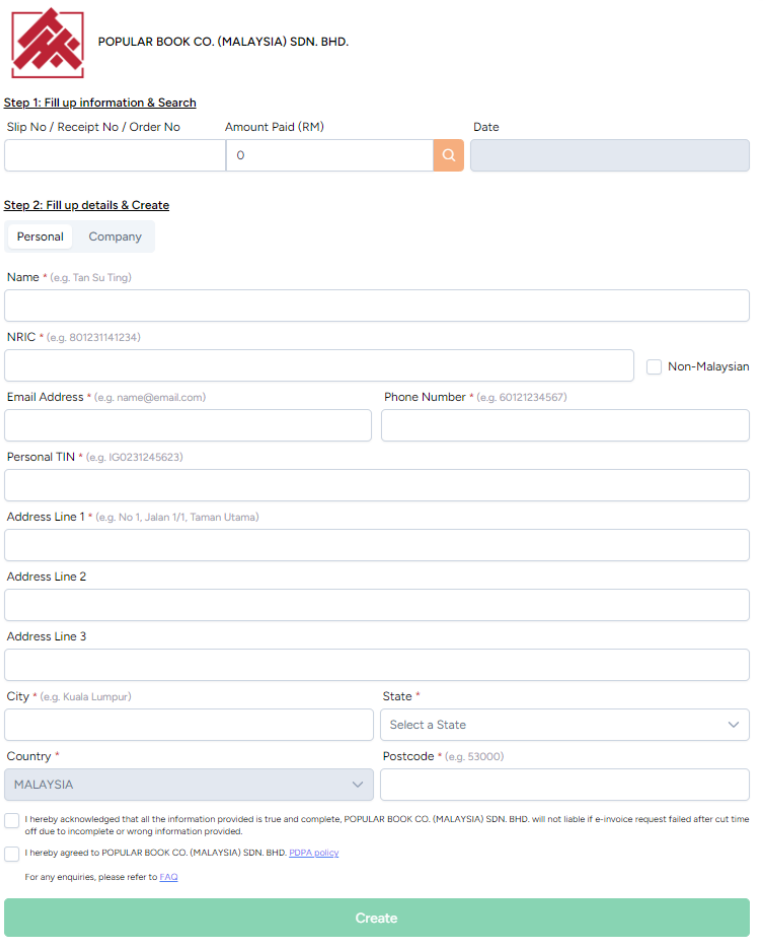


Figure 2.7 Request Form of Popular Book Co. (Malaysia) Sdn. Bhd.

Forms come in various formats which depend on what kind of purchase was made. Customers should enter their purchase slip details and purchase amounts that appear on

their receipt for purchases made in-store. Customers making online purchases must obtain their order number and amount from the “My Order” page found on POPULAR Online website. All store pick-up transactions require information extraction from the shipping notice as a part of the claims process.

POPULAR demonstrates its commitment to customer service through e-invoice implementation alongside regulatory standard compliance. The provided guidelines along with set timelines allow customers to easily access their e-invoices while benefiting from improved invoicing efficiency.

2.4 Review on Existing System

2.4.1 JomeInvoice

JomeInvoice is an effective e-invoicing platform designed for companies operating in Malaysia that serves to address the issues related to compliance with LHDN regulations as well as to promote the strategic upgrade of the overall financial management process [12]. It can be accessed as a Web application or through a downloadable iOS/Android application. JomeInvoice acts as a flexible software that easily connects with other current systems like ERP, accounts, CRM, and e-commerce by using-api links. By means of this integration capability, companies can uphold their established working models as the entirety of invoice data can be captured, converted, and submitted to the LHDN MyInvoice portal automatically. JomeInvoice allows businesses to gain the 100% compliance with local standards and enables them to free from the time-consuming routine, including the PDPA-compliant data protection mechanisms and digital signatures for the transactions' verifiability. Additionally, JomeInvoice improves business processes by providing user-friendly navigation, AI elements that help to reduce the burden from employees and improve productivity. Advanced reporting and business intelligence finally enable actual-time reporting and analytics that enhance companies' management by offering them correct information concerning their financial operations.

Nevertheless, some of the disadvantages of JomeInvoice are it requires perfect API management and compatibility control from the beginning. A potential disadvantage of the solution is that the users can have trouble learning and adjusting to new changes within the platform and the possible new flows it provides, which means that enough training must be provided to the clients.

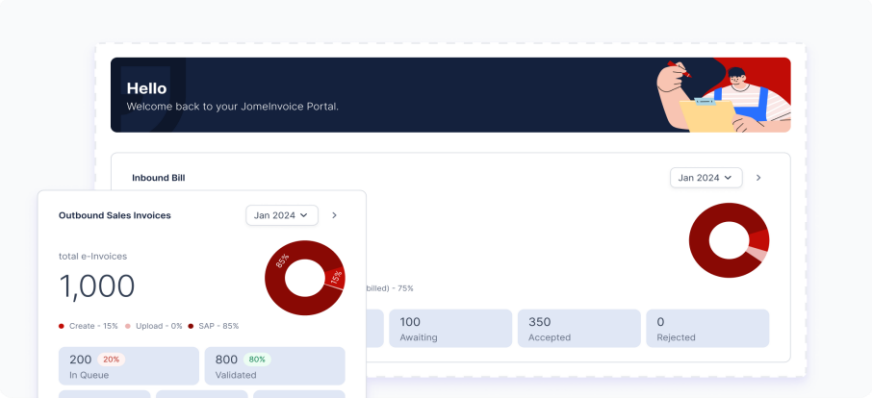


Figure 2.8: Interface of JomeInvoice. [12]

2.1.2 Refrens

Refrens, a strong e-invoicing software solution with targeted functionality specifically for the Malaysian business market with strong adherence to the IRBM/LHDN regulations and active use of multifunctional automation capabilities [13]. It presents an integrated set of tools whose main purpose is to support the management of e-invoices. It allows business people to produce professional invoices, credit and debit notes among other financial documents using custom designs in the shortest time possible. Among its functions, one can mention the possibility of the e-invoice being encoded with the relevant QR codes without the need for additional input from the clients to address the Malaysian tax requirements. These documents can also be forwarded by the user by electronic mail, WhatsApp, or as portable document format making communication with the clients quite easy. It has enhanced features of automation that cuts on time and work input since about 80% of the data input can be automated. It has smart data validation to check the data for truths and to correct the data to meet the standard. Security aspect takes into consideration when it comes to adding various levels of users, administrators and their privileges and roles concerning the information to be analyzed. Other key features include API connection, auto payment follow-ups, invoicing in multiple currencies, client and vendor management as well as per-client expenditure tracking to support all sorts of businesses.

Nevertheless, Users' have raised key concerns over international payment issues. Users noted that they encounter failure in payment services even when the transaction is effective, and prolonged releases of funds add to the problem [13]. Another more or less well-known problem concerns customer support which has been labeled as clearly inadequate and falling short of expectations. Additionally, it was noted that some features are missing or limited, for example, there is no AI mode and auto-responder, and the export/import invoicing feature is not very informative.

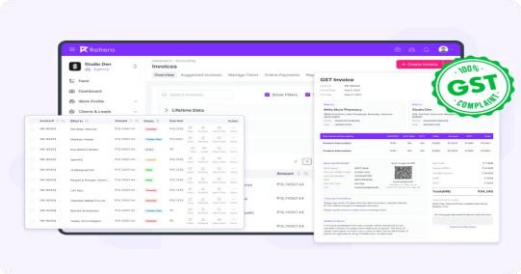


Figure 2.9: Interface of Refrens. [13]

2.1.3 Storecove

Storecove is an e-invoicing provider which has over 10 years of experience in providing the services focusing primarily in compliance with the regulations regarding the e-invoicing process [14]. They mostly concentrate on making sure that their customers are ready to meet every condition of the Malaysia's LHDN reporting that is set to start from August 2024 They provide services of sending compliant invoices, reporting to LHDNM, and delivery through the Peppol network. One of Storecove's clients' specific offerings consist of automatic accounts receivable procedures and one-stop-shop API for e-invoicing across the world, offering businesses a way to handle billings and compliance confidently. Many of them also assist software providers on how to implement Peppol functionalities and become resellers of sophisticated e-invoicing solutions of the providers.

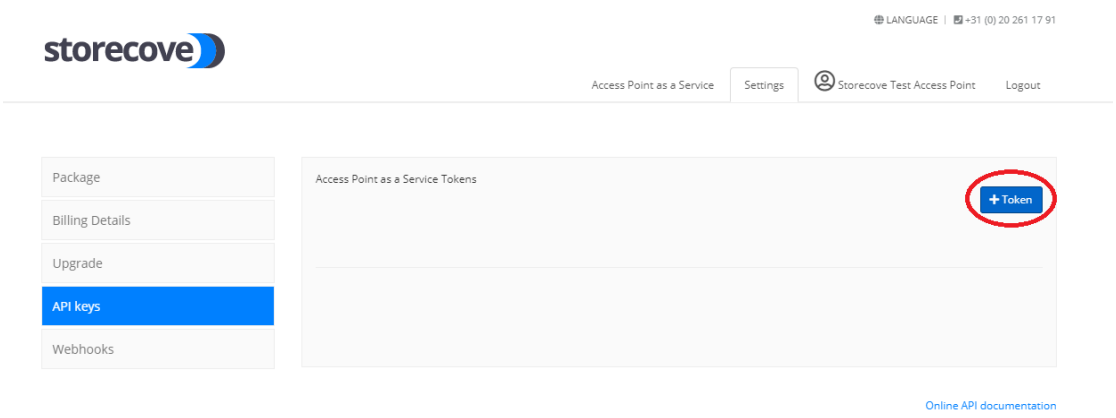


Figure 2.10: Interface of Storecove. [15]

2.5 Summary of the Existing System's Review

Table 2.2: Summary of the Existing System's Review.

Features	JomeInvoice	Refrens	Storecove	COME-Invoicing
Ease of Use	User friendly navigation but complex API management.	Intuitive interface.	Straightforward interface.	Simplify the transition from paper-based to e-invoicing.
Compliance with LHDN Regulations	100% compliance with Malaysian LHDN regulations.	Adheres to LHDN regulations but limitations in international transactions.	Meet all LHDN requirements.	Fully compliant with LHDN regulations, with automatic updates on legal changes.
AI Integration	Incorporates AI elements.	Lacks AI capabilities and advanced automation.	Lacks AI features.	Includes OCR with Google ML Kit
API Integration	Highly flexible with API links to integrate with ERP, CRM but complex from start.	Includes API connections but has reported issues with international payments.	Mainly focusing on compliance and Peppol network.	Mainly focusing on compliance and Peppol network.
Multi-Platform	Accessible via web and mobile apps (iOS/Android)	Web-based	Web-based	Both web and mobile accessibility
Data Security	PDPA-compliant data protection mechanisms and digital signatures.	Provides various levels of user access but lack the advanced security features.	Robust Security	Digital signatures and encrypted data transmission
Multi-Language	Bahasa Malaysia and English	Bahasa Malaysia and English	Global platform, possibly supporting multiple languages	Aims to offer support in multiple languages
Support for Small Businesses	Too complex	Limited technical skills quite hard to access it.	Not fully address the needs of small businesses	Simple, designed to support hawkers and small businesses

2.6 Limitation of Previous Studies

The limitations of the previous studies JomeInvoice, Refrens, and Storecove highlight several challenges faced by hawkers and small business owners in Malaysia. The combination of API management and compatibility control from the starting point puts **JomeInvoice** at risk of being time-consuming and asking for numerous resources. Further, the users may have difficulties when it comes to the new updates and features, and may even spend much time in familiarizing and training in order to be in a position to use them efficiently. **Refrens** has significant problems with the international payments, which is one of the key indicators that prevents Refrens from being a reliable business partner, users have reported that the payment services fail sometimes, and there are delays in the funds release, that makes the companies' operations suffer [15]. The customer support that Refrens provide is also considered very poor and does not meet user expectation. Besides, it has some disadvantages like the absence of AI mode, auto-responder, and there is no developed export/import invoicing system. Currently, the platform most associated with **Storecove** is designed mainly to address compliance with e-invoicing legislation and, in particular, the use of the Peppol network, which may not fully meet the interests of entrepreneurs in search of more diverse invoicing possibilities. This could limit its usefulness for some corporation as it focused on compliance solutions may not fully meet their needs. Also, it is deeply integrated with the use of the Peppol network, which cannot be entirely suitable for various business models.

2.7 Proposed Solutions

With regard to the drawbacks of the current e-invoicing solutions and the special requirements of hawkers, and small businesses in the context of Malaysia, the features of the **COME-Invoicing** application have been suggested as the effective solution for the seamless realization of e-invoicing. The application is aimed at the existing lack of digital skills, legislative requirements, and organizational problems that are facing these business owners.

Hence, the challenges faced by users who have little skills in handling digital applications indirectly point to the fact that COME-Invoicing differs in being the application that has an easy to use interface that does not pose difficulty to such users. The application optimizes the shift from paper-based to e-invoicing through the inclusion of Advanced OCR functionalities as part of AI. This enables users to scan handwritten receipts and sales records, the application subsequently tagging them for easier record keeping and compliance.

Also, the app has a voice recording feature, which lets the users record and report the sales information instead of typing them down. The application is also embedded with features like auto-population of monthly reports and its submission to the relevant authorities locally, with total compliance to the Malaysian LHDN requirements. COME-Invoicing offers users updates on alterations to the e-invoicing law and means that all the users are updated and in conformity with the law without much strive. The provision of the support in multiple languages would allow users with different language proficiency levels to use the application effectively and thus the continuity of the efforts to narrow down the gap between people with and without access to ICTs.

CHAPTER 3

Proposed Method/Approach

There are different software development models such as the Waterfall model, the Prototyping model, and Agile development model. Table 3.1 shows a comparison of these methodologies:

Table 3.1: Strengths and Weaknesses of the Reviewed Methodologies.

Methodology	**Waterfall**	**Prototyping**	**Agile**
Development Type	Sequential	Iterative	Iterative
Suited For	Well-defined and structured applications	Small to medium-sized applications	Small to medium applications with evolving requirements
Project Planning	Highly structured and detailed	Flexible with evolving designs	Adaptive and continuous planning
Development Speed	Slower due to rigid structure	Moderate, depending on number of iterations	Faster with frequent releases and updates
Documentation	Extensive documentation at each phase	Basic documentation, mostly focused on prototype revisions	Minimal documentation, prioritizing working software

From the comparison above, the Waterfall and Prototyping models are not ideal for this project due to its small scale. Agile emphasizes dividing the project into features delivered in multiple sprints.

After reviewing the methodologies, Agile Development is recommended for this project. Agile promotes iterative development throughout the project lifecycle,

allowing developers to review the framework after each iteration, ensuring the project stays on track or adapting to new requirements as needed. Even though there are no external customers for this project, if any dissatisfaction arises during development, requirements can be adjusted accordingly. Additionally, unlike the Waterfall model, Agile allows for simultaneous development and testing, which supports rapid development—a key factor since this project is expected to be completed within 6-7 months.

The development of this application will involve four phases: Planning, Analysis, Design, and Implementation.

Planning Phase

This phase will involve creating a rough idea that will be taken to the supervisor of more details on the target users and the mobile platform. After approval, the problem statement to be addressed and the project title to be used will be identified. The project scope and the objectives of the current project will then be set before constructing a Gantt chart that suits the project and likely to be completed at a given date.

Analysis Phase

During the analysis phase, similar existing systems will be reviewed to analyze their features, strengths, weaknesses, and potential solutions. This analysis will guide the integration of features to enhance user satisfaction. The chosen development methodology will also be analyzed, and the expectations for the project will be established. Finally, a development tool for the application will be selected.

Design Phase

During the design process, UML/ERD, use case and activity diagrams will be used as well as data dictionaries. In addition, to ease visualization a user interface and a prototype will be developed. If the prototype developed does not satisfy the need of the developer then the design has to be carried out repeatedly until an acceptable solution is arrived at. This phase is quite generic and as I move through the coding phase then enhancements as will be seen take time to implement.

Implementation Phase

In the last phase of the research known as the implementation phase, coding will get on. After this, the front end shall be designed, and the APIs that were identified in the previous phase shall be shown. An effective means for reducing such errors and bugs will always be incorporated by going through the code severally, coding and testing, then go back to coding and testing once again until all the problems are corrected. When the user and structural interface will work, a backend to handle data and servers will be developed next. The application will subsequently be posted on a server, and subsequently released to users. It will be during the maintenance stage where feedback will be taken in a bid to improve on the model. The last result of the work will be the prepared and completed report.

3.1 System Requirement

3.1.1 Hardware

The Asus ZenBook 13 UX331 is a compact and efficient laptop equipped with an Intel Core i5-8250U processor, making it suitable for various computing tasks. It runs on Windows 11, ensuring access to the latest features and updates. The laptop features an NVIDIA GeForce graphics card, providing enhanced visual performance for tasks such as video editing and light gaming. With 8GB of DDR3 SDRAM, it offers smooth multitasking capabilities, and its 256GB SATA3 HDD provides ample storage for essential files and applications. Overall, the Asus ZenBook 13 UX331 is a well-rounded laptop for everyday use.

Table 3.2 Specifications of laptop.

Description	**Specifications**
Model	Asus ZenBook 13 UX331
Processor	Intel Core i5-8250U
Operating System	Windows 11
Graphic	NVIDIA GeForce
Memory	8GB DDR3 SDRAM
Storage	256GB SATA3 HDD

3.1.2 Software

For the operating system of the application, it will not be iOS or Windows but Android OS. The programming language that is to be used will be Java. SQLite3 will be used for performing the database services. Finally, the integrated development environment specifically used for the development of the application shall be Android Studio.

Table 3.3 Software Requirements.

Description	**Minimum Requirement**
OS	Android
Android Version	11.0 above
Programming Language	Java
Database	SQLite3
IDE	Android Studio

3.2 System Design Diagram

3.2.1 Development Tools

Android Studio

Android Studio is the official Integrated Development Environment for Android developed by Google to offer developers a complete Integrated Development Environment with the IntelliJ IDEA as the basis, offering a graphical layout editor and a fast emulator [16]. Besides, it supports C++, Java programming languages and, by default, Google Cloud Platform.

Google Cloud Speech API

It enables developers transcribe audio to text in over 120 languages in several platforms such as Android. It also accept audio files uploaded in the request and provide text results in few seconds [17].

Google ML Kit

Google ML Kit operates as a mobile machine learning SDK which provides native ML features through its SDK for Android and iOS applications. Through ML Kit developers can add machine learning features because they do not need extensive expertise in ML programming [18].

SQLite 3

SQLite 3 is a lightweight, serverless, self-contained SQL database engine. It is commonly used in mobile and desktop applications to store local data due to its simplicity, small footprint, and ease of integration into applications.

Firebase

Firebase is an end-to-end solution by Google to develop, optimize and monetize mobile and web applications. It brings out features such as real-time database, authentication, cloud storage, analytics, and hosting all in one package in order to foster proper app integration. Firebase is most useful for applications where data is being edited frequently across devices, the Real-time database offers updates in real time.

Java

Java is a widely-used, high-level, object-oriented programming language that is known for its portability across platforms. It follows the "Write Once, Run Anywhere" philosophy, which means that Java code can run on any device or operating system that has the Java Virtual Machine (JVM) installed. Java is commonly used for building web applications, mobile apps (especially Android apps), desktop software, and large-scale enterprise systems.

3.3 System Architecture

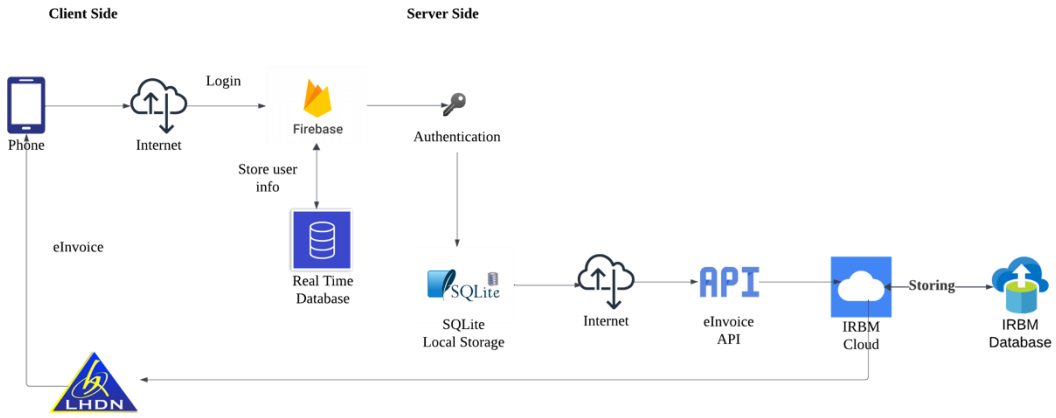


Figure 3.1 System Architecture of COME-Invoicing.

The COME-Invoicing system provides hawkers and small businesses with a streamlined way to comply with Malaysia's e-invoicing regulations through a robust, mobile-first design. Users access the system through an Android application, requiring an internet connection for initial registration and login via Firebase Authentication. The real time database of Firebase securely saves a user's encrypted email and password login credentials. The app uses SQLite3 once you log in to do offline data storage which to help users do daily sales records intuitive, generate invoice, manage inventory without internet connection which is very important for users in an area with bad connectivity.

After a user completes an order and is ready to submit an invoice, they reconnect to Internet to send the invoice data using the government provided eInvoice API. They then route the invoice info to the Inland Revenue Board of Malaysia (IRBM) cloud and IRBM storage that is interoperated with the LHDN (Inland Revenue Board of Malaysia) system. All that LHDN is guaranteed is the security of this transmission for the sake of regulatory compliance, to then have the right to log the e-invoice for future auditing and tax reporting purposes.

Once the invoice is submitted to LHDN, the user will receive an email notification to complete the invoicing procedure as and when LHDN requires additional details that are not reflected on the invoice. This notification contains a request for mandatory fields such as the Tax Identification Number (TIN) or Identity Card (IC) number or SST

(Sales and Services Tax) registration number. In this case when user fills in these details, system will take this information and will give a complete formal e-invoice to the user at his or her registered email id. This is an e-invoice that counts as a document of a fact of which users have a verified record of account transactions.

The COME-Invoicing application facilitates the switch to digital invoicing for small business owners, by combining offline functionality with compliance processes in the cloud. Firebase and local SQLite3 storage are also used in the architecture — meaning this is a flexible and easy to use tool that brings the digital divide to the forefront for low tech users in the evolving digital economy that is Malaysia.

3.4 Use Case Diagram

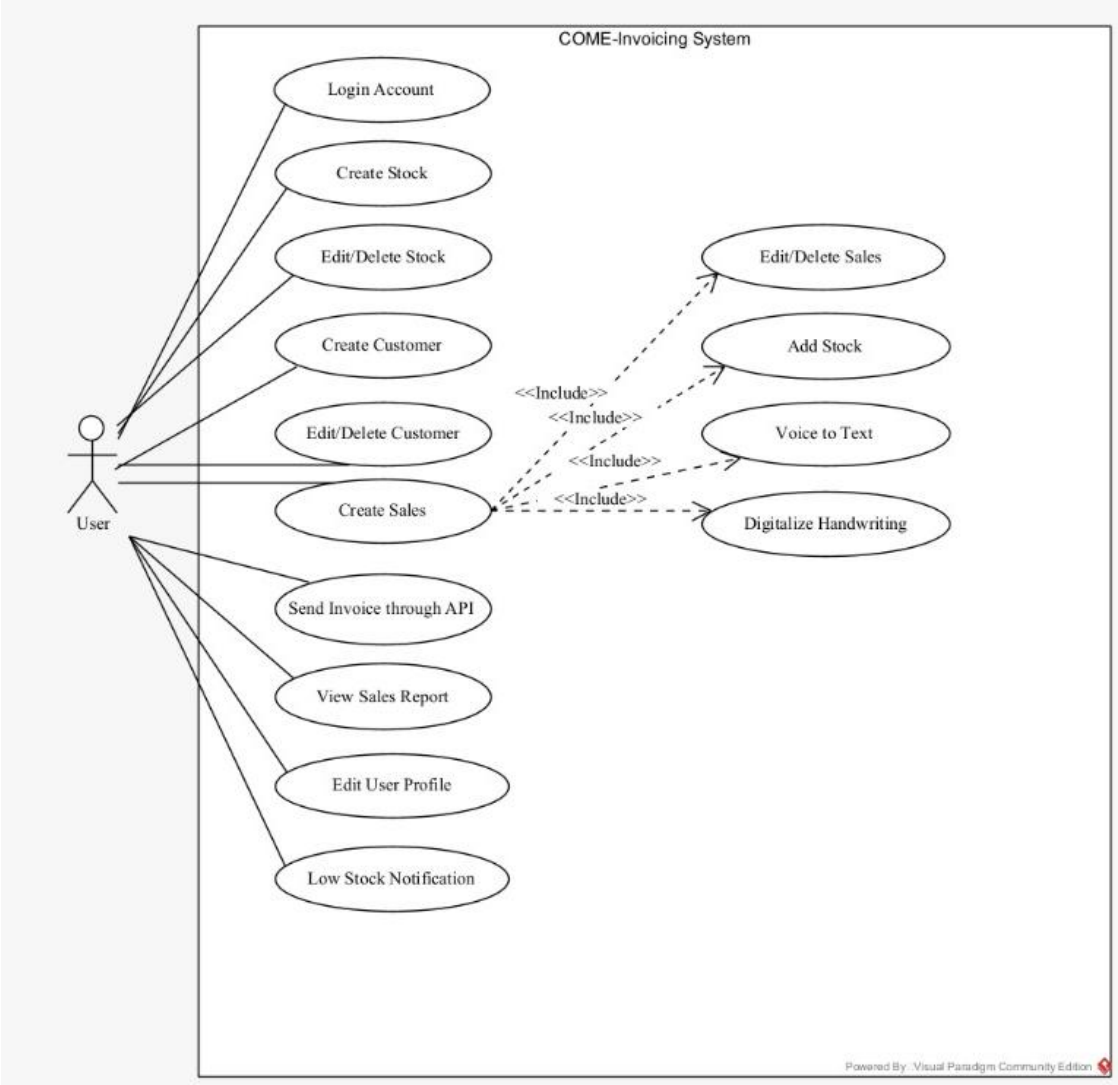


Figure 3.2: Use Case Diagram.

The use case diagram displays those activities that define the "COME-Invoicing System" and the way the user observes specific facets of this system. The system can be utilised to do several operations that include; account, stock, customers, and sales management. In particular, the user is able to log in to an account, add and modify or remove stock, add customers and modify or remove customer data. The user can also create sales, which is another capability of the system that is made up of other sub-tasks which are 'include' relationship, for instance, modifying/editing or deleting sales, ability to add stock, speech-to-text which can enable the voice to be converted to text at a click of a button or digitize handwriting. Moreover, the system contains features for generating invoices by API, reviewing the Sales Report, and editing the user's profile, and low stock notification. These actions characterise the activities that can be

performed at the user interact interface with the system and each is shown as a use case linked to the user actor. The given diagram also illustrates all the major aspects of this particular invoicing system while stressing the customer-oriented approach and flexibility when it comes to managing the financial activities and the customer information.

3.5 Activity Diagram

3.5.1 Login Function

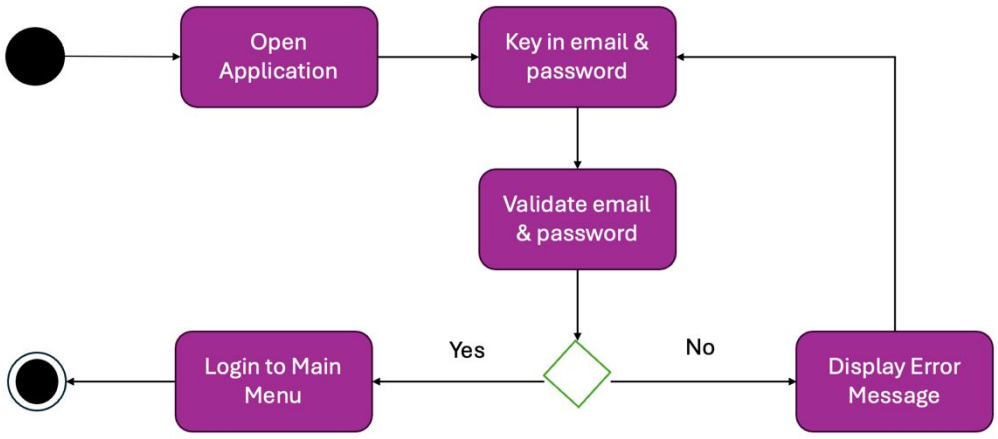


Figure 3.3: Activity Diagram of Login Function.

The activity diagram illustrated a login process within an application. The sequence was initiated when the user opened the application, which triggered the login process. Next, the user was prompted to enter their email and password as part of the authentication procedure. Once these credentials were provided, the system proceeded to validate them, checking if the email and password matched the stored records in the authentication system.

The flow diverged at this stage depending on the validation result. If the credentials turned out to be correct, the user was allowed to enter and took them to the main menu of the application. If the validation failed, meaning that the email or password didn't match, then an error message was displayed to let the user know they failed at log in. It sent this message to the user telling them to try again, if they fail it prompts the user to reenter their credentials if they are “having problems”.

3.5.2 Register Function

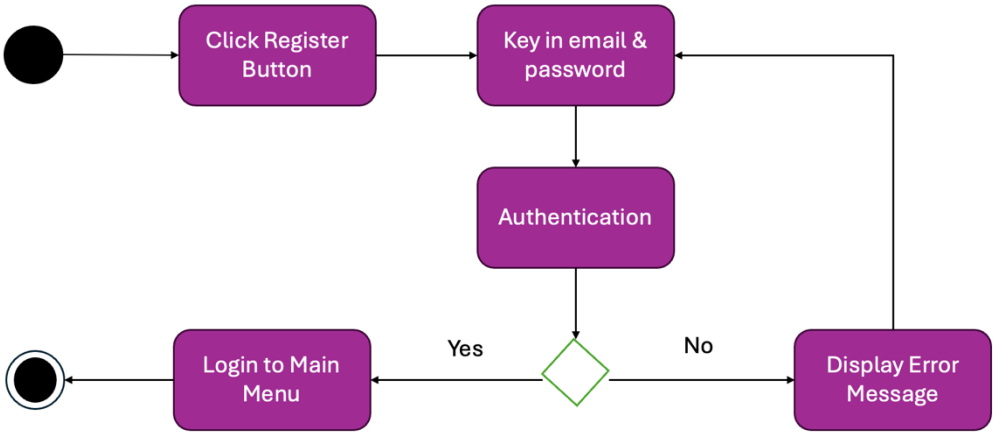


Figure 3.4: Activity Diagram of Register Function.

The process began when the user clicked the "Register" button, which initiated the registration workflow. After clicking this button, the user was prompted to enter their email and password as credentials for creating an account.

Once the email and password were entered, the system proceeded to authenticate the credentials. If the credentials met the required standards and passed the authentication check, the user was successfully registered and subsequently logged in, leading them to the main menu of the application. However, if the authentication failed, it indicating an issue such as an invalid or previously registered email or password not meeting complexity requirements—an error message was displayed. This message informed the user that the user were unable to register it and invited user to try again with the required information.

3.5.3 Create Stock Function

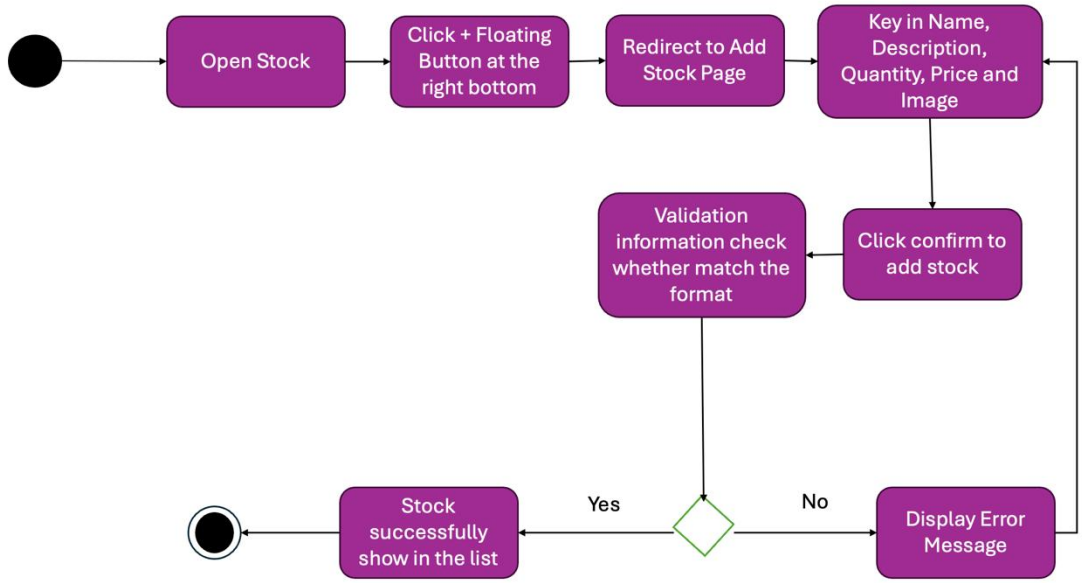


Figure 3.5: Activity Diagram of Create Stock Function.

The activity diagram above describes the process of adding a new stock item in the application. The process began with the user opening the "Stock" section, where they could view existing inventory items. To add a new stock item, the user clicked on the floating "+" button located at the bottom right of the screen. This action redirected them to the "Add Stock" page, specifically designed for entering new stock details.

On the "Add Stock" page, the user was prompted to enter various details, including the item's name, description, quantity, price, and an image representing the product. Once all required fields were completed, the user clicked the "Confirm" button to proceed with adding the stock item.

Following this, a validation check was performed by the system to ensure that the entered information met the necessary format and data requirements. If the validation was successful, the stock item was added, and it appeared in the stock list, signifying a successful addition. However, if the data did not meet the required format or if any field was missing or incorrect, an error message was displayed, informing the user of the issue and prompting them to correct the data before attempting to add the stock again.

3.5.4 Edit Stock

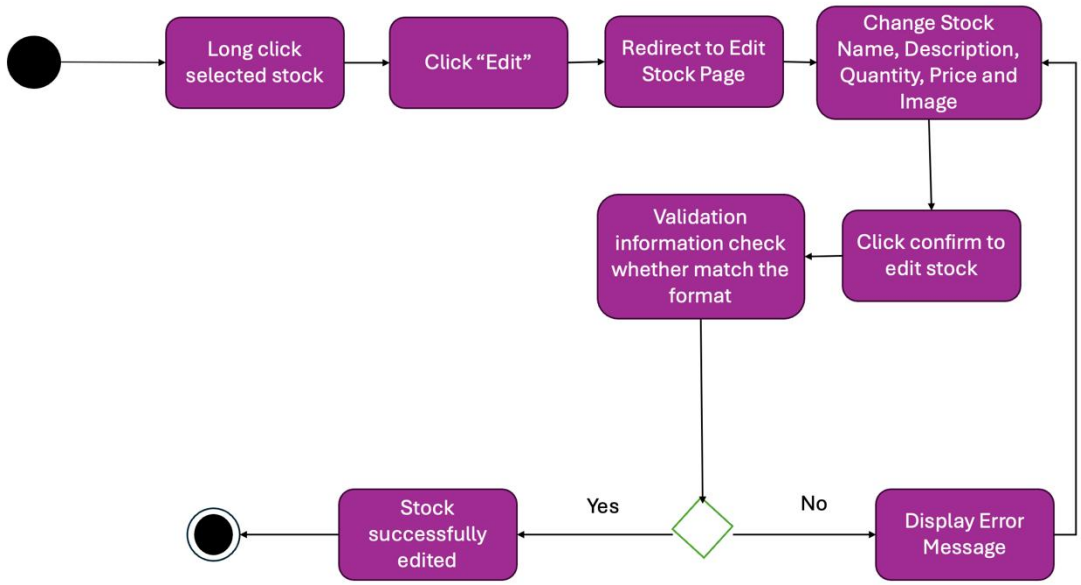


Figure 3.6: Activity Diagram of Edit Stock Function.

When the user long clicks on a stock item, the process starts in order to edit the stock. The user started this by selecting the item and clicking "Edit" and being taken to the "Edit Stock" page. Here, the stock name, description, quantity, price, and image were updated according to need. After the change was made, the "Confirm" button was clicked to save the change. It was then validated to make sure that information is of the right format. If the stock was able to be edited successfully with the meeting validation criteria, a message was provided informing the user that the stock was successfully edited. The editing process would be stopped, until corrections were made if the information did not pass validation and an error message would be shown.

3.5.5 Delete Stock Function

Figure 3.7: Activity Diagram of Delete Stock Function.

Here is the activity diagram of the process that takes place when user delete the stock in inventory management system. Here's a detailed explanation:

When a long-clicked stock item was selected, this initiated the process. After the selection, the user clicked on the "Delete" option to carry forward the deletion. When triggered by the 'Delete' option, the stock was successfully removed from the system after deleting. That said the stock record was permanently edited by deletion, and the flow ended indicating that enough.

3.5.6 Create Customer Function

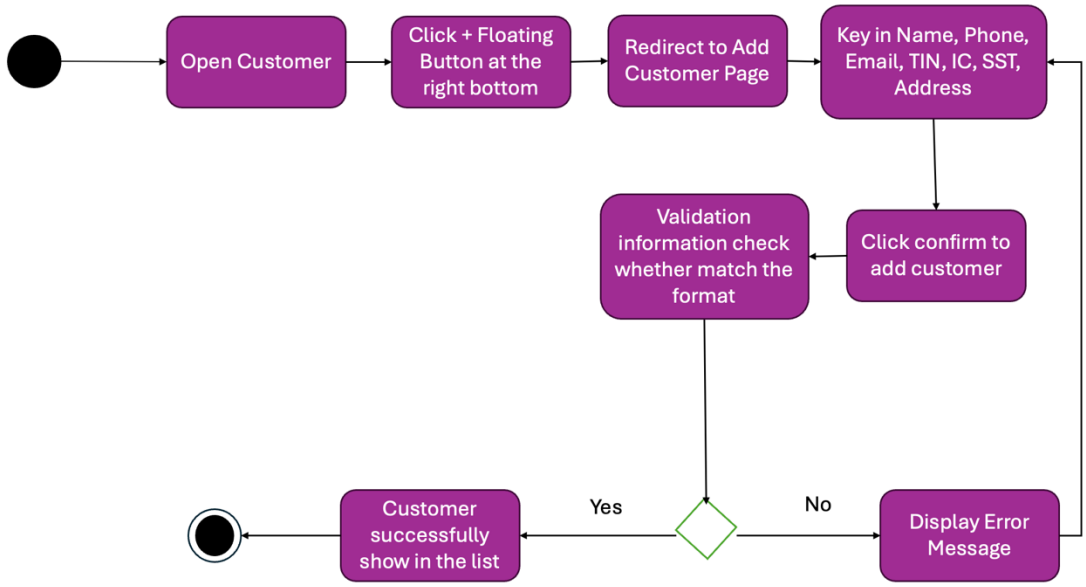


Figure 3.8: Activity Diagram of Create Customer Function.

It starts when the "Customer" section is opened. Then the user clicked on the button of “add customer” which is floating button that is located at the bottom right corner of the page and then he is directed to a “add customer” page. For the purposes of this page, the details of the customer is entered based on name, phone number, email, Tax Identification Number (TIN), Identity Card (IC) number, Sales and Service Tax (SST) number and address. Once you have entered the information the user click 'Confirm' submit the details. Then a validation check is performed to make sure all information is formatted in accordance. The customer is added if the data is valid, and they are now shown up in the customer list. An error message is also displayed if there is an error in the data format, and the user is then asked to fix this information before following on.

3.5.7 Edit Customer

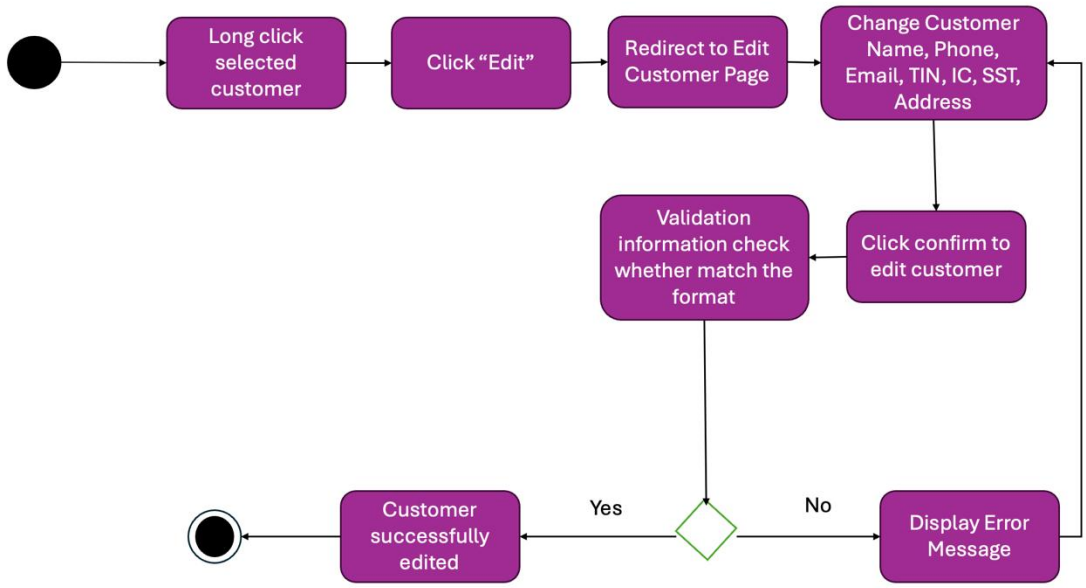


Figure 3.9: Activity Diagram of Edit Customer Function.

The process began with a selected customer being long-clicked. After this action, the "Edit" option was clicked, which redirected the user to the "Edit Customer Page." On this page, various customer details, including Name, Phone, Email, TIN, IC, SST, and Address, were modified as required. Once the necessary changes were made, the "Confirm" button was clicked to proceed with editing the customer information.

Following this, the system checked if the entered information matched the required format. If the information passed the validation, the customer details were successfully updated, and a confirmation was displayed. However, if the information did not meet the validation criteria, an error message was displayed, prompting the user to make the necessary corrections before retrying.

3.5.8 Delete Customer Function

Figure 3.10: Activity Diagram of Delete Customer Function.

The process was initiated when a customer was long-clicked to select them for deletion. After the customer was selected, the "Delete" option was clicked by the user to proceed with the deletion. Once the "Delete" option was activated, the customer’s information was successfully removed from the system, completing the deletion process. The flow then concluded, indicating that the customer record was permanently removed.

3.5.9 Create Order Function

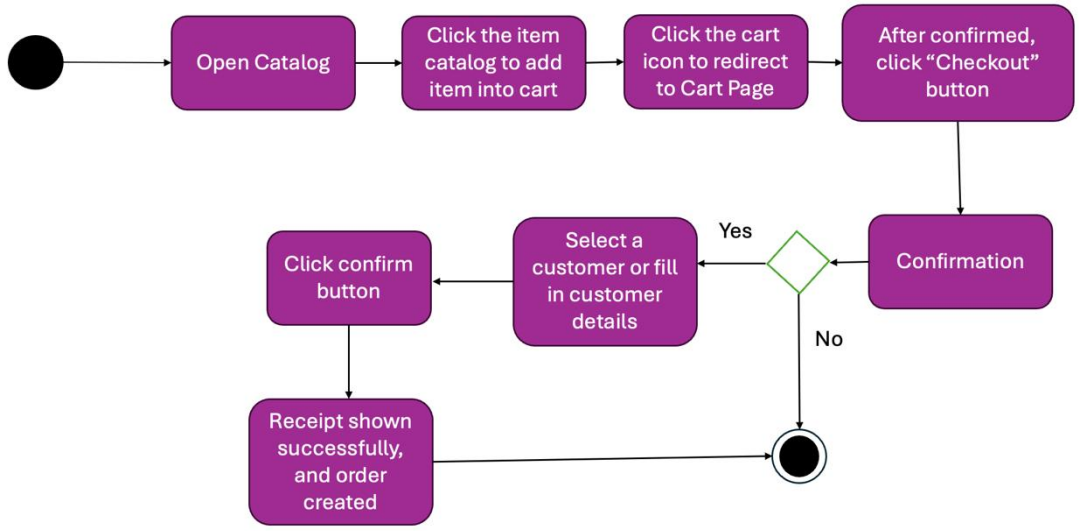


Figure 3.11: Activity Diagram of Create Order Function.

The process began with the catalog being opened. An item was then selected from the catalog to be added to the cart. After adding items, the cart icon was clicked, which redirected the user to the Cart Page. Once the items were confirmed, the "Checkout" button was clicked to proceed.

Next, the user was prompted to either select an existing customer or fill in the required customer details. If customer information was provided, a confirmation was displayed. Then, the "Confirm" button was clicked, after which a receipt was shown successfully, and the order was created. If no customer details were provided, the process was terminated.

3.5.10 Delete Order Function

Figure 3.12: Activity Diagram of Delete Order Function.

It all began with a long clicking of the order which resulted in the order being selected for deletion. Once the user selected the order, the user then clicked the "Delete" option to delete the order. The moment I clicked the "Delete" button; the order got successfully taken out from the system; the process of deletion completed. It was then flowed with the flow tells that the order record was permanently deleted.

3.5.11 E-Invoice Validation Function (Expected)

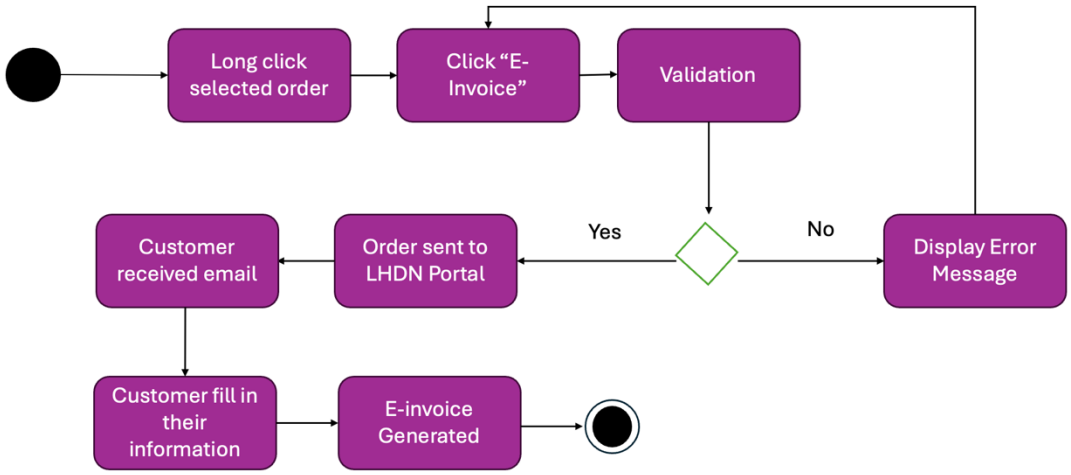


Figure 3.13: Activity Diagram of E-Invoice Validation Function.

When sending a "long click" order in this e-invoice process, the user chooses an order they want to invoice, and this is when things start. When this is selected, the user just clicks the 'E-Invoice' button, which means they want to create an electronic invoice for that order. This action prompts the system to do a check to see if the generation of an e-invoice is allowed since the system needs to verify if the e-invoice is produced according to certain requirements or not.

If the validation succeeds, order is given to the LHDN (Lembaga Hasil Dalam Negeri, or Inland Revenue Board) portal. As the second step, it fists tabulates the order information and forwards it to the tax authority system to record or process an invoice according to the requirements of the regulatory. When submitted successfully, an automatic email is sent to the customer regarding a new e invoice. You can either add more information for them in the email or send them an email with instructions of how they can add more information. The process of generating the e-invoice comes to an end the moment the customer fills their details, and the e-invoice is officially generated. But as soon as any validation check fails at any point, an error message is shown to the user indicating what is missing or incorrect so that the user can revisit and repair if needed and try again to generate the e-invoice. The e-invoice prevents the process from taking a step further to ensure it complies with the necessary standards so that no errors can go unnoticed.

3.5.12 Alternative Way E-Invoice Submission (Real Implementation)

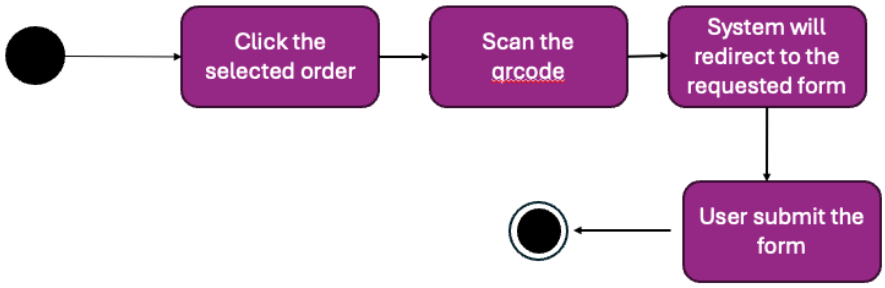


Figure 3.14: Activity Diagram of Alternative Way E-Invoice Validation Function.

The alternative e-invoice submission process begins by letting the user choose an order from their system. Users must choose an order before they proceed to scan its provided QR code located within the order details or receipt. The system conducts QR code processing after which the user obtains automatic access to the requested e-invoice submission form. Users must fill in required information upon form redirection before they submit their e-invoices to the system. After the system validates the submitted data it proceeds to verify the successful electronic invoice submission.

3.5.13 View Analysis Function

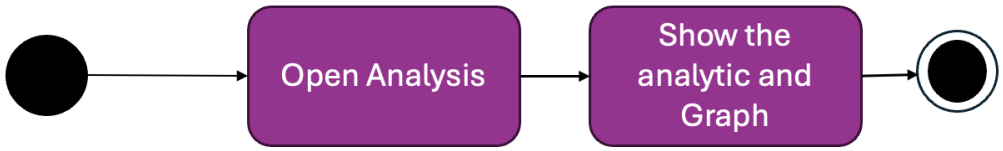


Figure 3.15: Activity Diagram of View Analysis Function.

The process began with the "Open Analysis" option being selected, initiating the analysis activity. When the analysis view was opened, data about the subject matter was shown in detail, such as a sales, inventory, or customer data.

Then, the system graphically displayed the data by producing a bar chart that compared different data points and facilitated easier analysis of trends and differences. It also included a pie chart which displayed the data as a color proportional of segments. The combination of detailed data and bar and pie charts gave a clear view of the information which helped bring some real insight into the data.

3.5.14 Edit Profile Function

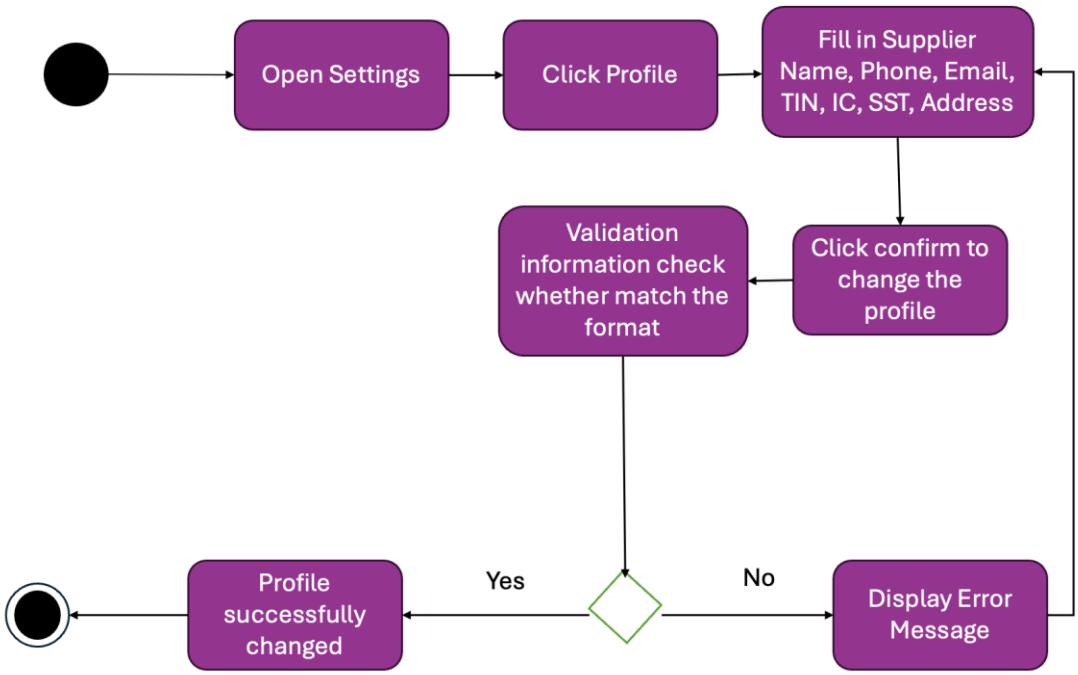


Figure 3.16: Activity Diagram of Edit Profile Function.

Once the user decides to edit the profile, he/she starts from the settings section by choosing the option: Profile. Here they can view and update all the personal details needed for their profile like the supplier's name, contact number, email, TIN, IC, SST number and address. After these fields are filled in, the user just clicks the "Confirm" button confirming the changes.

Upon confirmation, the system performs a validation check to ensure that all entered information matches the required format and complies with data entry standards. If the information passes validation, the profile is successfully updated, and the user sees a confirmation that their profile has been changed. However, if any of the information fails the validation check—due to missing, incorrect, or improperly formatted data—the system displays an error message. This feedback allows the user to correct the issues and try again, ensuring that only valid and complete information is saved in the profile.

3.5.15 Stock Aging Function

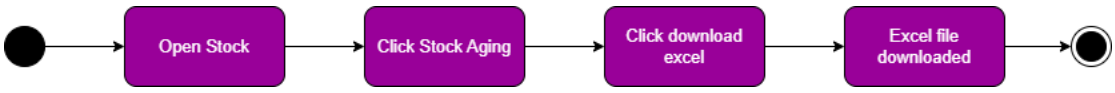


Figure 3.17: Activity Diagram of Stock Aging Function.

As the Stock Aging Function activity diagram reveals the system procedures to create stock aging reports. The system opens after a user selects the stock module to view inventory details. The system begins stock aging analysis after the user chooses the Stock Aging option from the available menu. After the user selects “Download Excel” from the system generates an Excel report which shows stock aging information. The completion of the process involves downloading the Excel file. This tool allows organizations to monitor inventory times to create better stock decisions while avoiding excess storage of old products.

3.5.16 Receipt Compile PDF Function

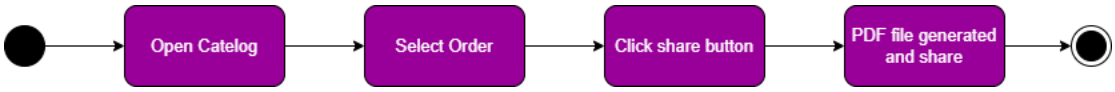


Figure 3.18: Activity Diagram of Receipt Compile PDF Function.

The Receipt Compile PDF Function activity diagram shows how system handles the operation of merging several order receipts into one consolidated PDF document. The catalog module order storage opens when the user initiates the process. The user chooses multiple orders from the catalog to prepare for compilation into receipt documents. The user selects multiple orders and then accesses the "Share" button which activates the system to create a PDF file displaying chosen receipts. The system concludes PDF sharing after finishing compilation and creates an accessible PDF file for users to store or distribute. The receipt management process becomes simpler because the system consolidates multiple receipts into a single PDF file which provides improved record-keeping and sharing options.

3.5.17 System Notification

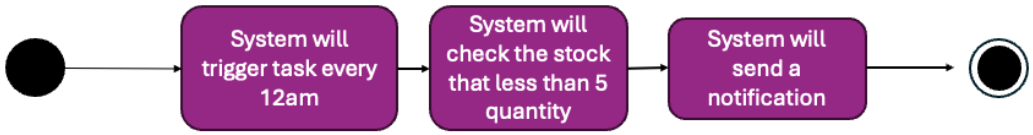


Figure 3.19: Activity Diagram of System Notification.

The daily automated stock monitoring procedure executes at midnight through the activity diagram. The system executes its first automatic task exactly at midnight. The system activates a check of inventory to locate all products with stock levels below five units. The system communicates directly to appropriate staff including store managers and procurement staff when detecting lower-than-five-unit items in stock. The system executes this process at midnight to ensure prompt restocking before stockouts occur. When the notification reach completion the system will proceed with its scheduled next run.

3.5.18 Handwriting to Text

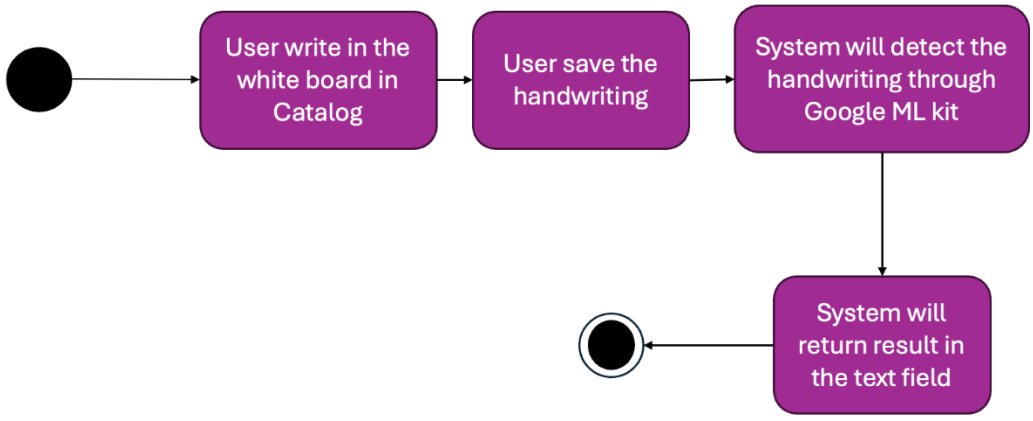


Figure 3.20: Activity Diagram of Handwriting to Text.

The activity diagram demonstrates how Google ML Kit handles handwriting recognition operations inside a catalog system. The handwriting interaction starts with users writing on digital whiteboard interfaces in the catalog. After users finish their writing session they must save the new piece of handwriting. After writing on the digital whiteboard users can save their handwriting before the system utilizes Google ML Kit's handwriting recognition solutions to process the notes. The user sees extracted text

results displayed within a text field after the recognition step of the system finishes. Within the application users can convert their handwritten notes into digital text through this easy workflow process.

3.5.19 OCR Digitalization

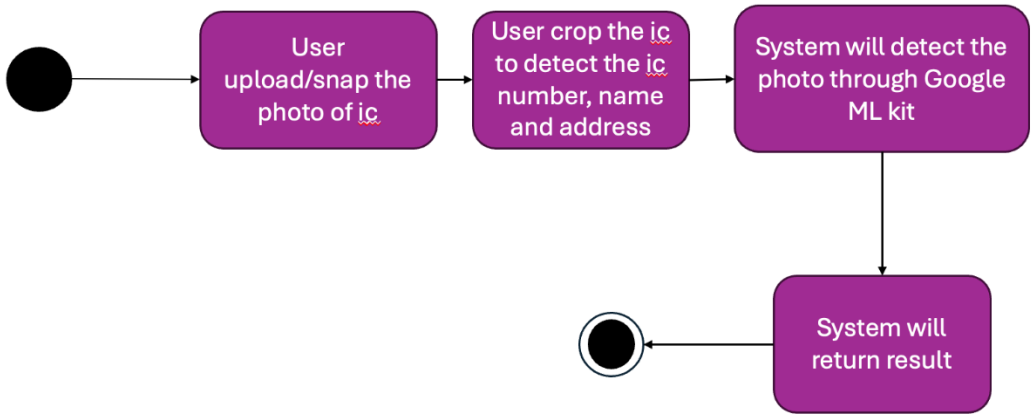


Figure 3.21: Activity Diagram of OCR Digitalization.

The activity diagram displays the workflow that extracts information from identity cards using Google ML Kit. A user starts the process by uploading or taking a picture of their identity card through the application interface. The user uses image cropping functionality to select important IC elements which include IC number, name, and address. Once the user prepares the cropped image the system starts text recognition with Google ML Kit to extract important information from the photo. After processing the information the system provides the recognized details in structured format to the user. The system uses this workflow to automate IC data retrieval operations which then improves both efficiency and data entry accuracy.

3.6 Timeline

3.6.1 FYP2

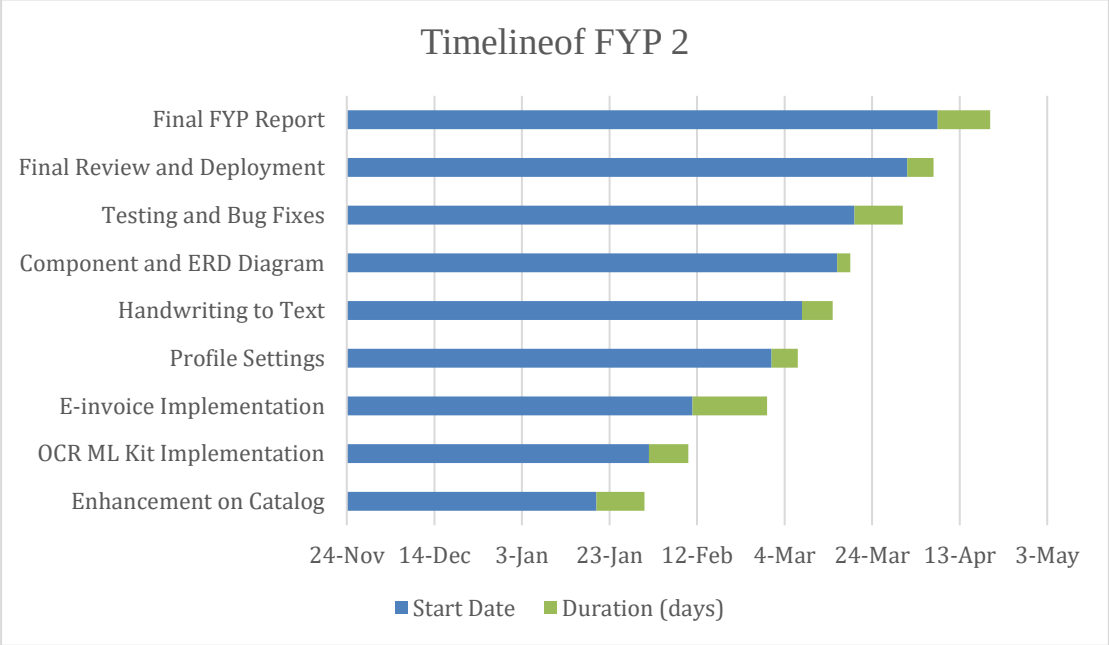


Figure 3.22: Timeline of FYP2.

CHAPTER 4

System Design

4.1 System Block Diagram

4.1.1 Firebase Authentication Sign In & Sign Up

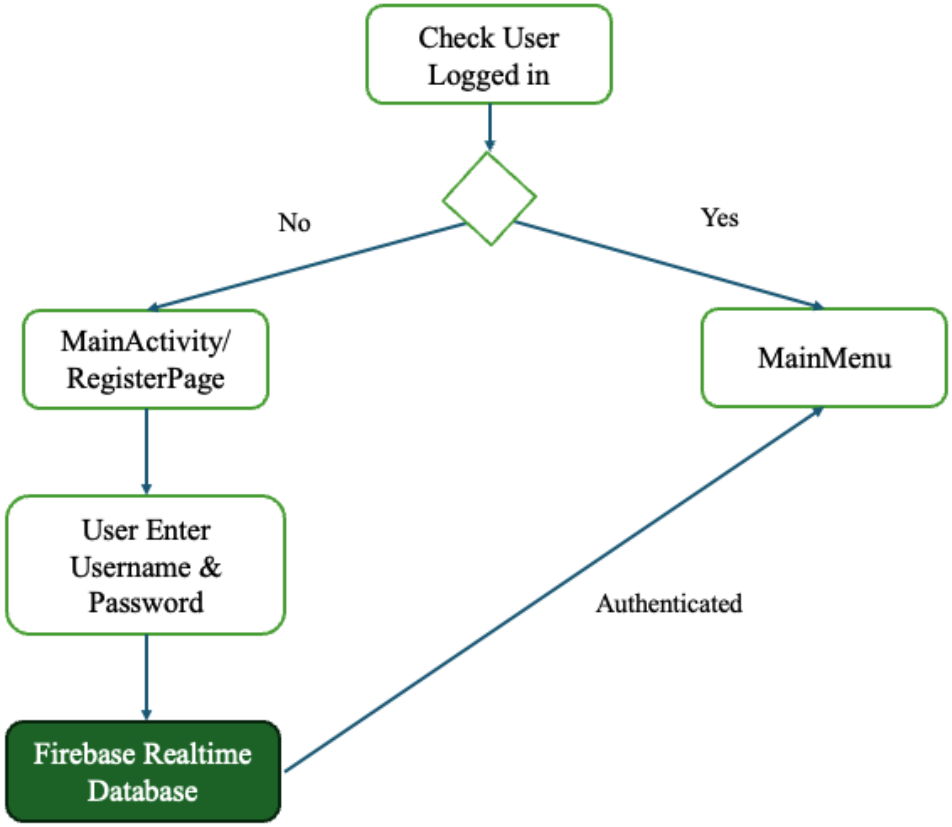


Figure 4.1: System Block Diagram of Firebase Authentication Sign In & Sign Up.

The block diagram represents the user authentication process for the e-invoice system, ensuring secure access and seamless navigation. When the application starts, it first checks whether the user is logged in. If the user is authenticated, they are directly navigated to the MainMenu, granting immediate access to the system's main functionalities. On the other hand, if the user is not logged in, they are redirected to the MainActivity or RegisterPage, where they must enter their username and password. These credentials are securely validated against the Firebase Realtime Database, which ensures real-time authentication and secure data synchronization. Firebase's robust infrastructure provides data integrity and protection, allowing the system to handle

multiple user sessions securely. This setup guarantees a smooth login experience and maintains secure user data storage.

4.1.2 Stock Management

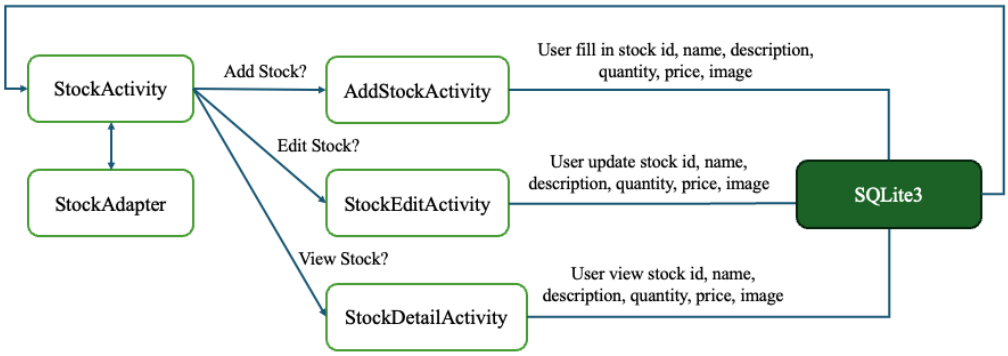


Figure 4.2: System Block Diagram of Stock Management.

The second block diagram delves into the stock management system, which plays a critical role in the e-invoice project's inventory control. The StockActivity serves as the central point for managing stock operations and seamlessly connects to various features for adding, editing, and viewing stock information. When users want to add a new stock item, they are redirected to the AddStockActivity, where they fill in essential details such as stock ID, name, description, quantity, price, and image. These details are then stored in the SQLite3 local database, ensuring fast and reliable data retrieval even without an internet connection. If users need to modify existing stock data, they can do so through the StockEditActivity, which allows them to update stock attributes efficiently. Additionally, the StockDetailActivity offers a comprehensive view of each stock item, enabling users to access detailed information at a glance. This structured and well-integrated stock management system ensures accurate inventory tracking, reduces errors, and facilitates informed decision-making in the sales process.

4.1.3 Stock Aging Module

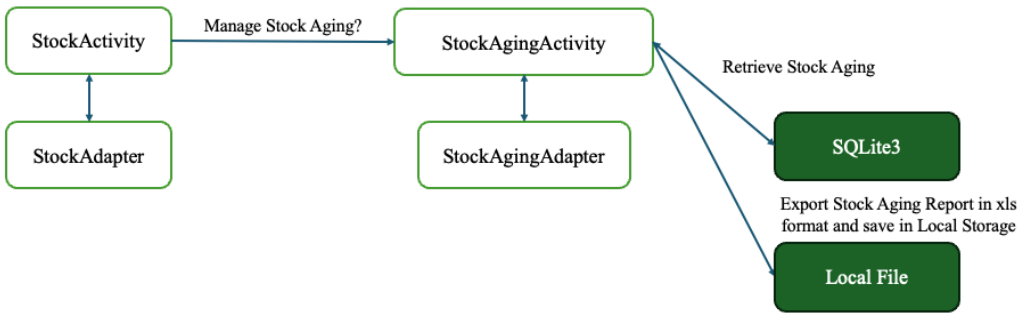


Figure 4.3: System Block Diagram of Stock Aging Module.

The third block diagram emphasizes the stock aging management feature, which is crucial for tracking the shelf life and turnover of inventory. From the StockActivity, users can navigate to the StockAgingActivity, which connects to the SQLite3 database to retrieve stock aging information. This feature allows users to monitor how long stock items have been held in inventory, providing insights that are essential for making timely sales decisions and preventing stock obsolescence. Moreover, the system includes functionality to export stock aging reports in XLS format, which are then saved in the local storage. This export capability allows users to access, share, and analyze stock aging data outside the application, promoting better reporting and analysis practices. By integrating stock aging management with export features, the system empowers businesses to maintain optimal inventory levels, improve stock turnover, and enhance operational efficiency. Overall, these three interconnected components work together to create a comprehensive and efficient e-invoice system tailored for modern business operations.

4.1.4 Customer Management

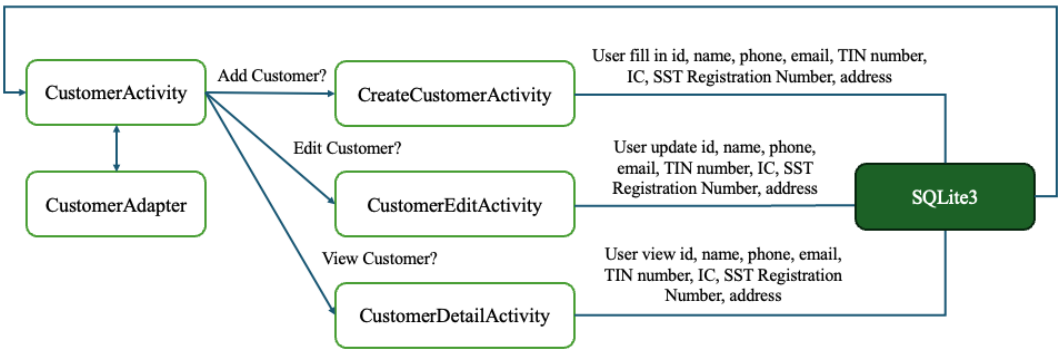


Figure 4.4: System Block Diagram of Customer Management.

The main component CustomerActivity functions as the central interface which allows users to view customer lists together with data interaction capabilities. The application uses CustomerAdapter to fetch SQLite3 customer data while ensuring correct visual display through the user interface.

Users access CreateCustomerActivity to input ID and nine various customer data points such as name and phone number before the details get stored into SQLite3. The data enters SQLite3 storage after users finish their submission. Users can edit existing customer information from CustomerEditActivity to change necessary details into SQLite3. The CustomerDetailActivity displays customer information retrieved from SQLite3 while preventing any modifications.

4.1.5 Catalog Module

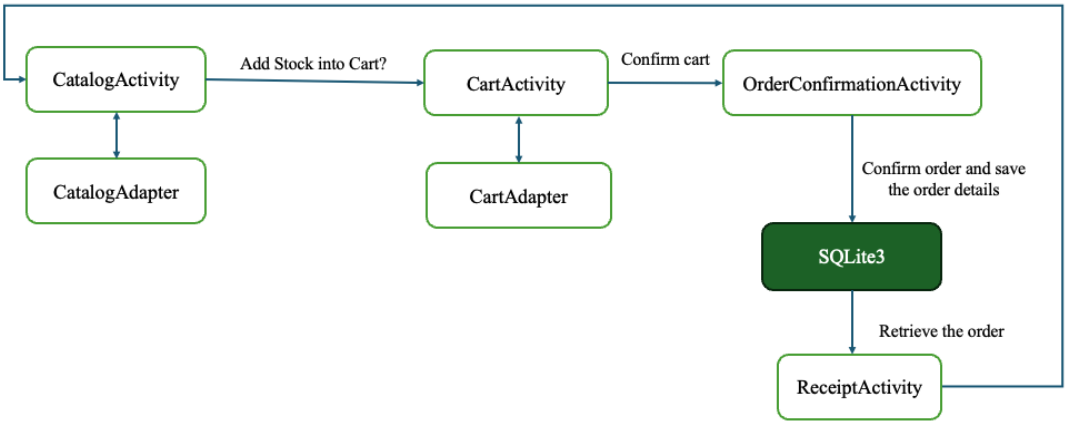


Figure 4.5: System Block Diagram of Catalog Module.

Users can access available stock items through CatalogActivity before starting their selection process. Products appear correctly because of the CatalogAdapter functionality. The user will move to CartActivity after choosing products before CartAdapter displays their selections.

A user who confirms their cart will move to OrderConfirmationActivity before finalizing their order during review. The system stores order information in SQLite3 databases after users make their confirmation. Through ReceiptActivity users can access stored orders by fetching information from SQLite3.

4.1.6 Analysis Module

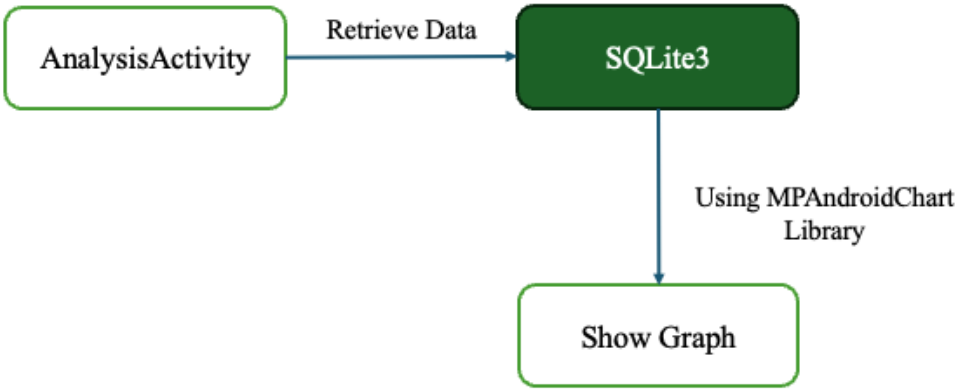


Figure 4.6: System Block Diagram of Analysis Module.

AnalysisActivity begins the process by retrieving information from the SQLite3 local database that maintains sales or performance records. The data fetch process generates visual representations after the SQL database retrieves the desired information. The MPAndroidChart library enables users to view data through graphical charts that enhance their ability to understand the information. The Show Graph element displays final results for efficient analysis of trends and insights.

4.1.7 Edit Profile

Figure 4.7: System Block Diagram of Edit Profile.

The block diagram demonstrates the relationship between SettingsActivity and EditCompanyProfileActivity components in an application. The SettingsActivity gives users access to navigate between itself and EditCompanyProfileActivity which allows users to edit company profile details. The data retrieval process from the system and essential updates take place simultaneously. The flow enables the EditCompanyProfileActivity to show updated company profile information to users when they access their settings. The application utilizes this structure to maintain efficient organization when users update their company profiles.

4.1.8 Voice Recognition

Figure 4.8: System Block Diagram of Voice Recognition.

The block diagram displays the communication between CatalogActivity and startVoiceRecognition() which functions as part of an application. The CatalogActivity allows users to begin voice recognition through Google Voice Recognition while they navigate the application screen. The startVoiceRecognition() function gets called after which it handles voice input through user processing for catalog item search and command execution. The application's flow design allows users to conduct their interactions without manual hands thus improving accessibility alongside user convenience during catalog sessions.

4.1.9 OCR Digitalization

Figure 4.9: System Block Diagram of OCR Digitalization.

The block diagram illustrates how Google ML Kit through OCR technology enables the digitalization of documents within a customer enrollment procedure. The CreateCustomerActivity serves as a section or display area in a program which enables new customer entry. The processImage() function accesses an image (such as an ID card or document) through this activity before using Google ML Kit for text recognition. ProcessImage function retrieves vital text from images before converting them into digital format which can be used for form autofilling or database customer information storage. The system increases operational speed through automated data input procedures which lower human errors.

4.2 ERD Diagram

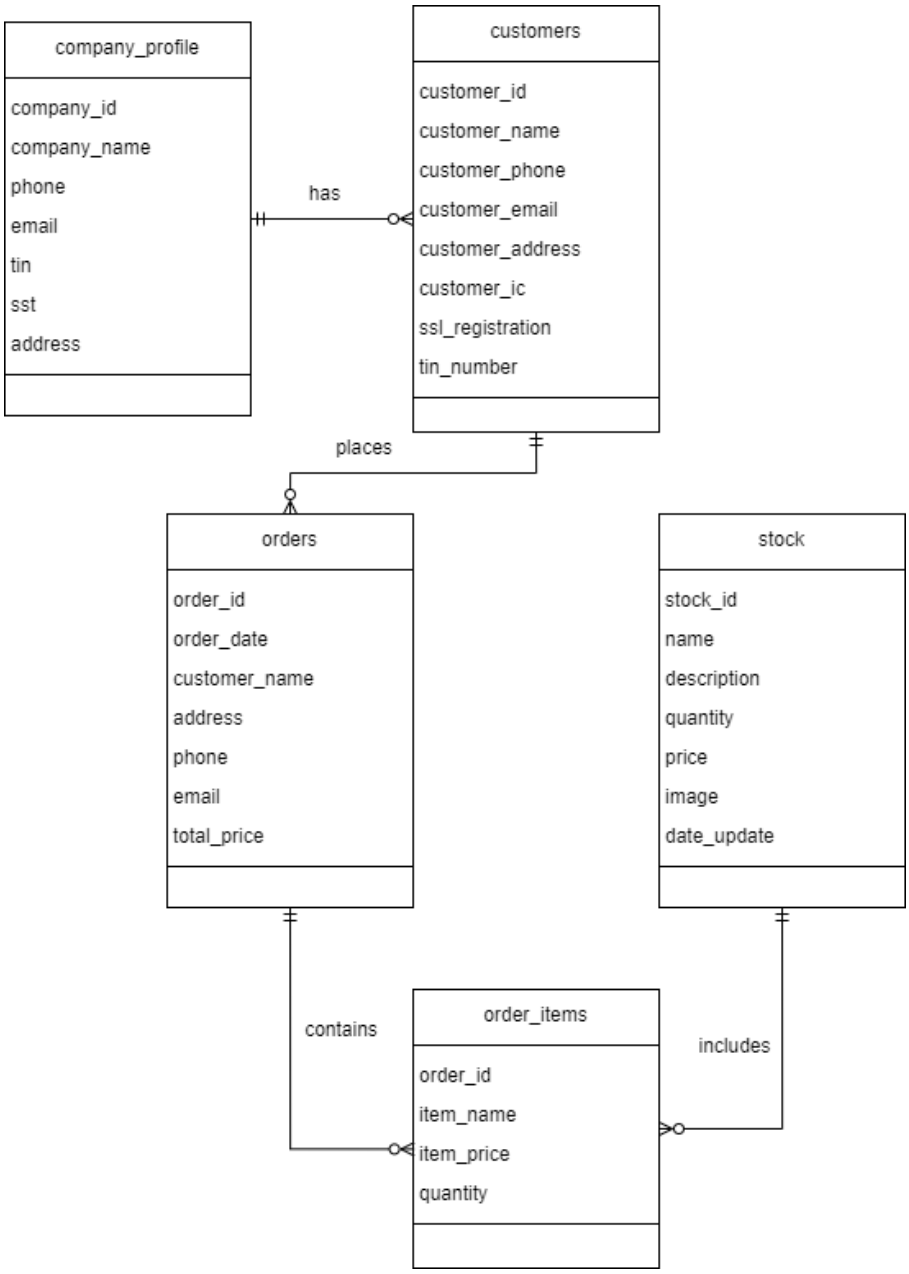


Figure 4.10: ERD Diagram.

The Entity-Relationship Diagram represents the core database structure for an order and invoicing management system, consisting of five primary entities: company_profile, customers, orders, stock, and order_items. The designed entities serve business needs by handling crucial operational features such as company data, customer account management and stock control, ordering procedures and billing management. The business operations run smoothly because these entities maintain perfect data communication.

The company_profile entity functions as the core business information storage point where the business name and contact information and TIN and SST details go along with the address. The customers table maintains a one-to-many association with this entity because one business organization can register multiple customers. The customers entity contains contact details which include customer names and phone numbers along with email addresses and addresses and identification card numbers (IC) as well as tax registration data. Customers can submit numerous orders resulting in a many-to-one relation with the orders database.

The orders entity contains essential transaction details by storing order ID with the order date and customer name while keeping address information and phone number and email and the complete order price demonstrated in each order record. Many customers can make orders so the entity maintains an essential function for order history management. The one-to-many relationship between order_items and orders exists since each order contains multiple items. Each order contains its individual items which the order_items entity stores using item name and both cost and quantity data. The system entity links order numbers to their corresponding products which have been bought.

The stock entity contains inventory records which include stock identification, item title, product description, remaining stocks number, selling cost, product image, and data updating timestamp. Each ordered item finds connection to existing stock entries via the many-to-one relationship with the order_items entity. Through this connection between the system the stock usage data can be tracked and inventory levels can be updated which prevents item overselling.

CHAPTER 5

System Implementation

5.1 Setting up

5.1.1 Software

Before the development of the COME-Invoicing application, four essential software components were installed and downloaded onto the laptop:

1. **Android Studio** – The latest version (Android Studio Ladybug) was downloaded from the Android Studio website, and the installation instructions were followed as outlined.
2. **Java Development Kit (JDK)** – The JDK 23, necessary for building Android applications, was downloaded from the Oracle website and installed, with the system's PATH variable configured correctly.
3. **Firebase SDK** – To utilize Firebase services (e.g., for authentication), the Firebase SDK was integrated into Android Studio by adding the necessary dependencies through **Tools > Firebase** and following the setup assistant instructions.
4. **SQLite 3** – For local database functionality, SQLite 3 was installed separately, as needed, if not already embedded within Android Studio, and was downloaded from the SQLite website.

With these software tools installed, the foundational setup for the COME-Invoicing development on the laptop was established.

5.1.2 Firebase Authentication

To set up Firebase Authentication for the COME-Invoicing app, Firebase was integrated into the Android project. Initially, the project was connected to Firebase from the Firebase console, and email and password authentication was enabled by opening the **Auth** section, selecting the **Sign-in Method** tab, and enabling the **Email/Password** option. This setup allowed users to register and log in using email credentials.

In the app's Gradle file (build.gradle.kts or build.gradle at the app module level), the dependency for Firebase Authentication was added. The Firebase Android BoM (Bill of Materials) was used to automatically select compatible library versions:

```
dependencies {
    // Add the Firebase Authentication library dependency
    Implementation(libs.firebase.auth)
}
```

Once these steps were completed, Firebase Authentication was ready to be used within the project.

To implement registration, an instance of the FirebaseAuth object was initialized in the app's sign-up activity using FirebaseAuth.getInstance(). On the user signup, there email and password from the user are passed as parameters to the createUserWithEmailAndPassword method. When the user user, successfully registered, the user was signed in automatically, and securely saved the user information to Firebase. In the sign in activity, similarly, signInWithEmailAndPassword, was set to let users in by their credentials.

To manage user sessions effectively in an Android app using Firebase Authentication, it's important to verify if a user is signed in whenever an Activity is initialized. This step ensures that users don't have to re login each time they open app, making for a better user experience. This could be done in Android by overriding the `onStart()` method in the Activity's lifecycle. An Activity becomes visible and will trigger the onStart() method, so that is a natural place to verify user login status.

The current user info was taken in the overridden `onStart()` method by invoking `mAuth.getCurrentUser()`. If the user is signed in this returns a FirebaseUser object if not it returns null. The returned value could be checked to determine the user's login state. It was confirmed that a user session existed if the `FirebaseUser` was not null. And in this case, one can simply call the reload() function to update the UI (user specific data), skip the login screens. This approach improves user experience since it reduces access to the app's features by preforming an implicit app store check for all returning users.

For those cases where a user wants to log out, Firebase provides a simple signOut() method. It can be called from any part of the app, from the log out button on the profile or settings screen. The calling of FirebaseAuth.getInstance().signOut() clears out the user's session and GetCurrentUser() returns null when checking in the future. This is particularly useful for apps that need secure access, or when switching between different user accounts — once the user signs out they'll have to sign back in again.

Once Firebase Authentication was set up for everything, it handled user log data and disconnected with the real time database to help with the apps extra functions.

5.1.3 Voice Recognition

The voice recognition was implemented into it using a method called startVoiceRecognition() to tackle voice input and process it when recognized. This method is to trigger the voice recognition functionality and to prompt user talking. The steps to set up this functionality are as follows:

1. **Initialize the Intent**:

   In the startVoiceRecognition() method, an Intent object was created that said the app can understand speech input. LANGUAGE_MODEL_FREE_FORM was specified so that the language model would accept natural language and Locale.getDefault() so that the user's default language would be picked. A prompt, "Say something...," was displayed to guide the user.

```
private void startVoiceRecognition() {
    Intent intent = new
Intent(RecognizerIntent.ACTION_RECOGNIZE_SPEECH);
    intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE_MODEL,
RecognizerIntent.LANGUAGE_MODEL_FREE_FORM);
    intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE,
Locale.getDefault());
    intent.putExtra(RecognizerIntent.EXTRA_PROMPT, "Say
something...");

    try {
        startActivityForResult(intent,
REQUEST_CODE_SPEECH_INPUT);
    } catch (Exception e) {
        Toast.makeText(this, "Voice recognition is not
available on this device", Toast.LENGTH_SHORT).show();
    }
}
```

2. **Handle the Start of Voice Recognition**:

   Within a try-catch block, the startActivityForResult() method was called to start the activity and wait for the user's voice input. If voice recognition was not available on the device, a Toast message would be displayed to inform the user.

```
  try {
    startActivityForResult(intent,
REQUEST_CODE_SPEECH_INPUT);
} catch (Exception e) {
    Toast.makeText(this, "Voice recognition is not available
on this device", Toast.LENGTH_SHORT).show();
}
```

3. **Process the Recognized Text**:

   The onActivityResult() method was overridden to process the recognized text after the user's input. When the request code and result code matched, the recognized speech results were retrieved using RecognizerIntent.EXTRA_RESULTS. If any results were returned, the first recognized phrase was set in the search bar (or another appropriate text field).

```
@Override
protected void onActivityResult(int requestCode, int
resultCode, Intent data) {
    super.onActivityResult(requestCode, resultCode, data);
    if (requestCode == REQUEST_CODE_SPEECH_INPUT &&
resultCode == RESULT_OK && data != null) {
        ArrayList<String> result =
data.getStringArrayListExtra(RecognizerIntent.EXTRA_RESULTS);
        if (result != null && !result.isEmpty()) {
            searchBar.setText(result.get(0)); // Set the
recognized text to the search bar
        }
    }
}
```

   By following these steps, voice recognition was successfully integrated, allowing the app to capture spoken input, convert it into text, and display it in the app's UI, specifically in the designated searchBar or any other text field required.

5.1.4 OCR Digitalization

The project builds a digital image-based MyKad information extraction system through an Android application where Google ML Kit's Text Recognition API handles the photo digitalization operations. Image selection follows cropping then text recognition before extracting data from the selected items. By pressing the "Upload" button the system launches an intent that makes the user choose an image from their gallery. The application selects an image through the "Upload" button before it gets cropped to a 1:1 aspect ratio with UCrop and finally gets converted to the Bitmap format. ML Kit's Text Recognition API receives the processed image to perform a text scan and extraction. The analysis within parseMyKadData() method examines extracted text to identify essential information including IC number and name and address sections. The system uses a specific pattern (XXXXXX-XX-XXXX) to identify the IC number followed by uppercase word recognition to find the name section and location keywords

(Jalan, Taman, Lorong and postal codes) for extracting the address. The system displays the extracted information in designated input forms of the user interface. StockDatabaseHelper facilitates local database storage of customer information although the system requires all essential fields to complete validation before saving. Data accuracy and efficiency increase through this implementation because manual errors during personal information extraction from MyKad no longer occur.

1. **Dependencies is installed on build.gradle (Module: app):**

```
dependencies {
   implementation 'com.google.mlkit:text-recognition:16.0.0'
   implementation 'com.yalantis:ucrop:2.2.6'
}
```

2. **Select Image from Gallery**

```
    private static final int PICK_IMAGE_REQUEST = 1;
    private Uri imageUri;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
       super.onCreate(savedInstanceState);
       setContentView(R.layout.activity_main);

       Button btnUpload = findViewById(R.id.btnUpload);
       btnUpload.setOnClickListener(v -> openGallery());
    }

    private void openGallery() {
       Intent intent = new Intent(Intent.ACTION_PICK,
    MediaStore.Images.Media.EXTERNAL_CONTENT_URI);
       startActivityForResult(intent, PICK_IMAGE_REQUEST);
    }

    @Override
```

```
protected void onActivityResult(int requestCode, int resultCode, Intent data)
{
   super.onActivityResult(requestCode, resultCode, data);
   if (requestCode == PICK_IMAGE_REQUEST && resultCode ==
RESULT_OK && data != null) {
      imageUri = data.getData();
      startCrop(imageUri);
   }
}
```

3. Using the UCrop Library

```
private void startCrop(Uri uri) {
   Uri destinationUri = Uri.fromFile(new File(getCacheDir(),
"cropped.jpg"));

   UCrop.of(uri, destinationUri)
      .withAspectRatio(1, 1)
      .start(this);
}

@Override
protected void onActivityResult(int requestCode, int resultCode, Intent
data) {
   super.onActivityResult(requestCode, resultCode, data);

   if (requestCode == UCrop.REQUEST_CROP && resultCode ==
RESULT_OK) {
      Uri croppedImageUri = UCrop.getOutput(data);
      imgMyKad.setImageURI(croppedImageUri);
      processImage(croppedImageUri);
   }
}
```

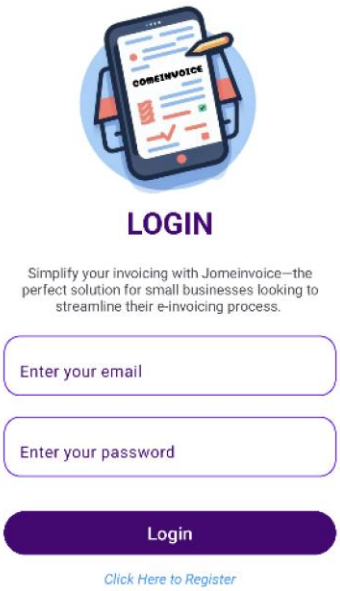
4. Process Image with MLKit

```
private void processImage(Uri uri) {
   try {
      Bitmap bitmap =
MediaStore.Images.Media.getBitmap(getContentResolver(), uri);
      InputImage image = InputImage.fromBitmap(bitmap, 0);
      TextRecognizer recognizer = TextRecognition.getClient();

      recognizer.process(image)
         .addOnSuccessListener(visionText -> extractMyKadData(visionText))
```

```
            .addOnFailureListener(e -> Toast.makeText(this, "Failed to recognize
text", Toast.LENGTH_SHORT).show());
    } catch (IOException e) {
        e.printStackTrace();
    }
}
```

5.2 System User Interface & System Operation

Login	
 Figure 5.1: UI of Login.	**Login Function**: Authenticates users for system access using Firebase Authentication. **Input Fields**: 1. Username field for user identification. 2. Password field for security. **Submission**: 1. "Login" button submits entered credentials for validation. **Validation**: 1. System checks if credentials match the records. 2. If valid, user gains access to system features and services.

Register	**Register Function**: Sign Up users through Firebase Authentication. **Input Fields**: 1. Username field for user identification. 2. Password field for security. **Submission**: 1. "Register" button submits entered email & credentials for validation. **Validation**: 1. System checks if email already signed up. 2. If new users, sign up successfully.

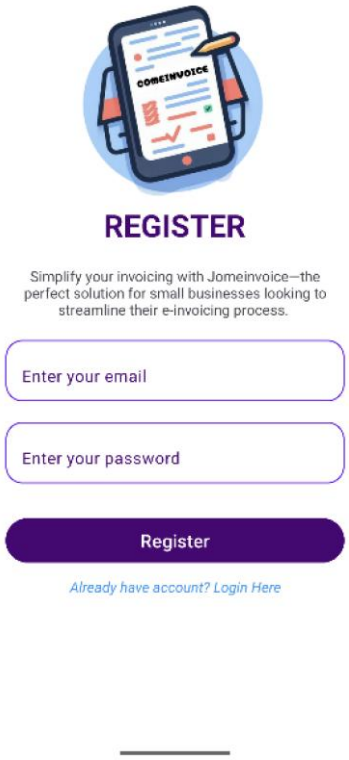
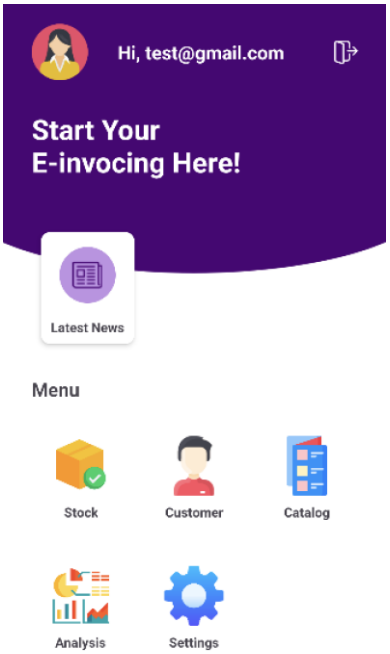
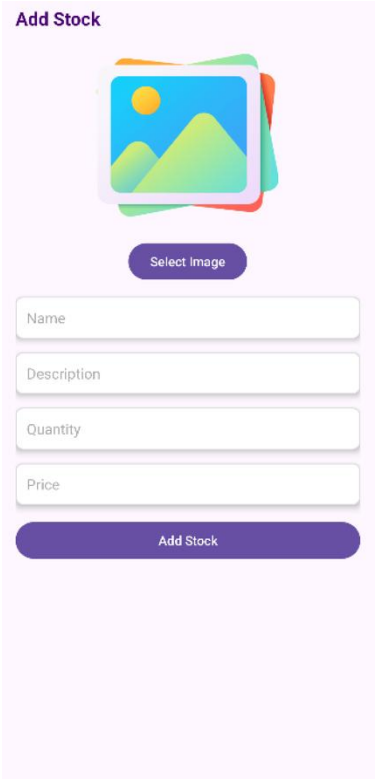


Figure 5.2: UI of Register.

Menu	
 Figure 5.3: UI of Menu.	**Welcome Message**: Displays a personalized welcome message. **Menu Options**: Five key functional areas represented by buttons or icons: 1. **Stock**: Likely leads to a feature for managing inventory or tracking stock levels. 2. **Customer**: Probably directs users to customer management tools such as client information or contact details. 3. **Catalog**: Likely takes users to the sales tracking or historical of sales feature. 4. **Analysis**: Could lead to data analytics or reporting tools for performance insights. 5. **Settings**: Manage user profile, tax and currency and logout function.

Stock List	
 Figure 5.4: UI of Stock List.	**Stock List**: A list of products with their respective details: 1. **Stock Name**: Displays the name of the product (e.g., iPhone 16, Banana, Pineapple, Pear). 2. **Quantity**: Shows the available quantity for each product (e.g., "10 in stock " for iPhone 16). 3. **Price**: Displays the price per unit of the product (e.g., RM 4199.00 for iPhone 16). **Add New Product Button**: At the bottom right corner, there is a circular button with a "+" symbol, indicating a function to add new products to the stock list. **Stock Aging Button**: At the top right corner, there is a stock aging button to access to view the details of stock aging.

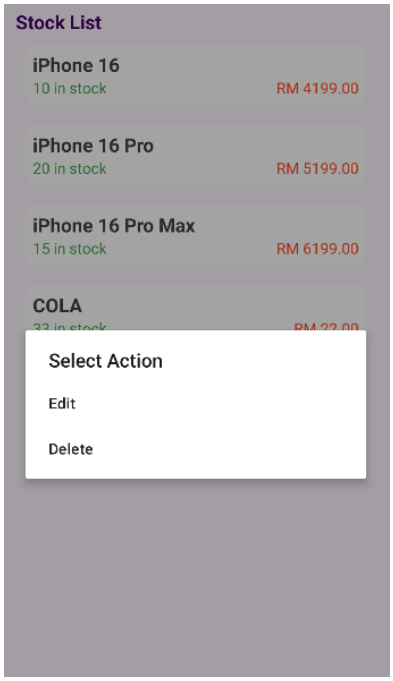
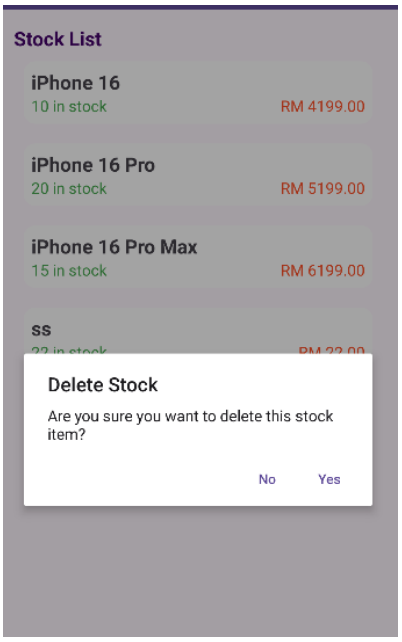
Add New Stock	
 Figure 5.5: UI of Add New Stock.	**Image Placeholder:** A circular area where an image of the stock item can be uploaded or displayed. **Stock Name:** A text input field to enter the name of the stock item. **Price:** A text input field to input the price of the stock item. **Quantity Field:** A numeric input field to enter the quantity of the stock item available. **Add Stock Button:** A button to save the entered stock information into the system.

View Stock Details	Display all information of the stock: Name, Image, Description, Quantity and Price.
 Figure 5.6: UI of View Stock Details.	

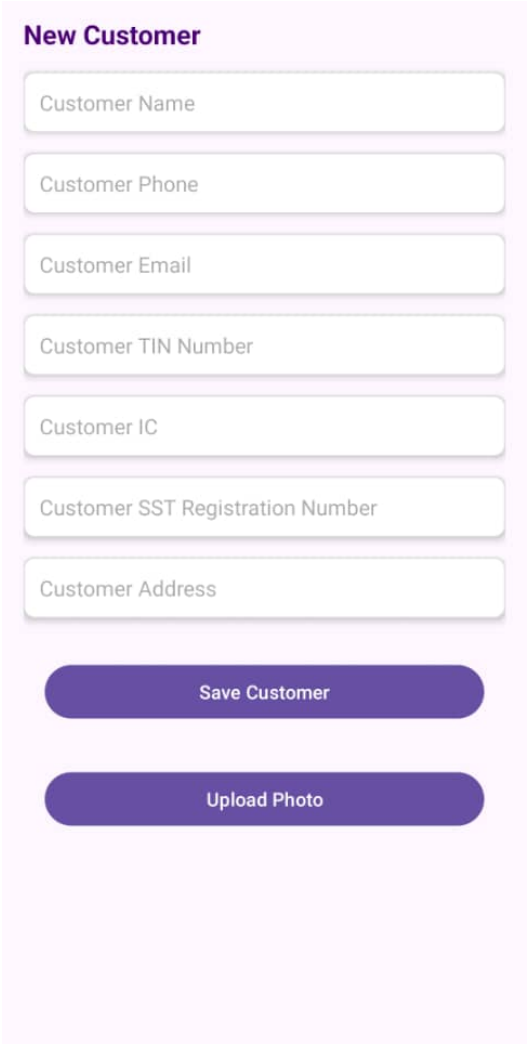
Edit Stock Details

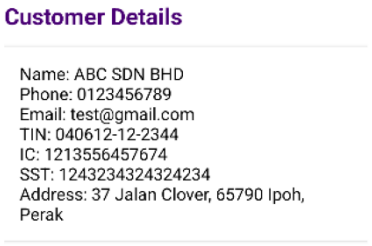
Figure 5.7: UI of Edit Stock Details.

1. User performs a long click on the stock item they wish to modify.
2. A message box appears with options: "Edit" and "Delete."
3. User selects the "Edit" option.
4. The system navigates the user to the Edit Stock Details page.
5. On this page, the user can update the stock name, description, quantity, and price.
6. Once the updates are made, the user clicks the "Update Stock" button.
7. The updated stock details are saved, and the stock information is modified accordingly.

Delete Stock	
  Figure 5.8: UI of Delete Stock.	1. User performs a long click on the stock item they wish to modify. 2. A message box appears with options: "Edit" and "Delete." 3. User selects the "Delete" option. 4. The system prompts for confirmation to ensure the user wants to delete the item (optional step if confirmation is required). 5. Upon confirmation, the stock item is deleted from the list. 6. A success message is displayed, indicating that the stock item has been deleted successfully.

Customer List	Customer Entries:
 Figure 5.9: UI of Customer List.	1. **Customer Name (e.g., "ABC Sdn Bhd"):** The name of the customer or company. 2. **Phone Number (e.g., "011-12345678"):** The phone contact for the customer. **Add Button (+):** • A circular button with a plus sign located at the bottom of the screen, which likely allows the user to add a new customer to the list.

Add New Customer	
 Figure 5.10: UI of Add New Customer.	• **Customer Name:** A text input field for entering the customer's name or the company name. • **Phone:** A text input field for entering the customer's phone number. • **Email:** A text input field for entering the customer's email address. • **Others:** It might have others field with did not include in wireframe due to not enough space such as Customer's TIN Number, Company Registration Number or other required data fields for e-Invoice. Please refer to this [4]. • **Address:** A text input field for entering the customer's address. • **Save Button:** A button at the bottom of the form to save the customer's details into the system. • **Upload Photo:** Alternatively, user can upload the customer ic photo and let system performs OCR function.

View Customer Details  Figure 5.11: UI of View Customer Details.	□ Display all information of Customer: Name, Phone, Email, TIN, IC, SST, Address.

Edit Customer Details

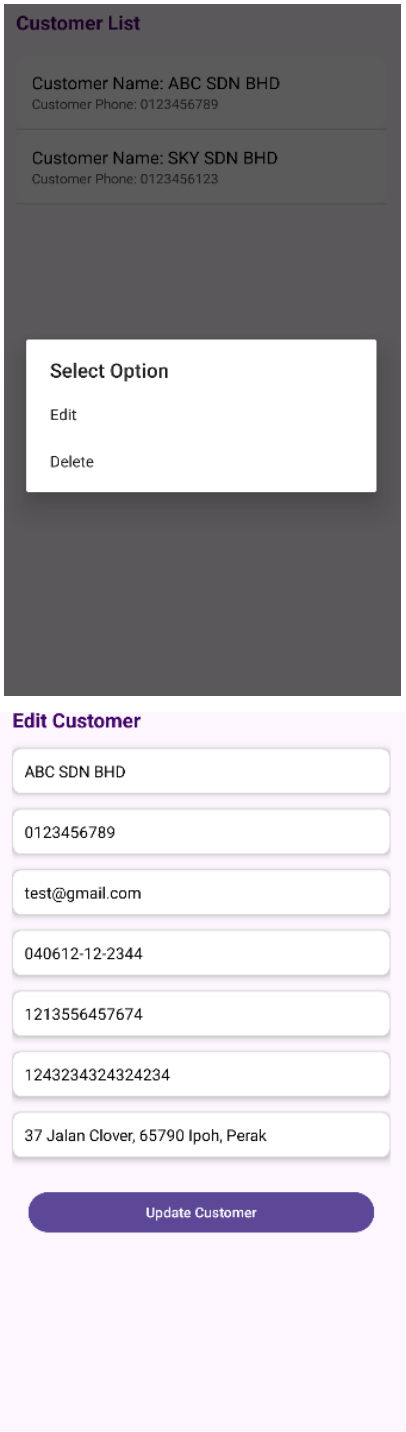
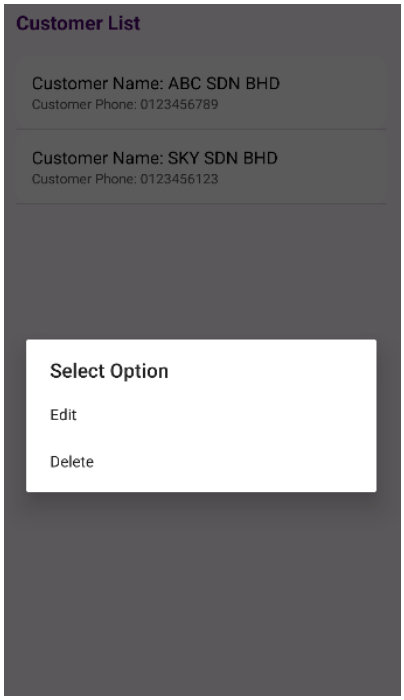
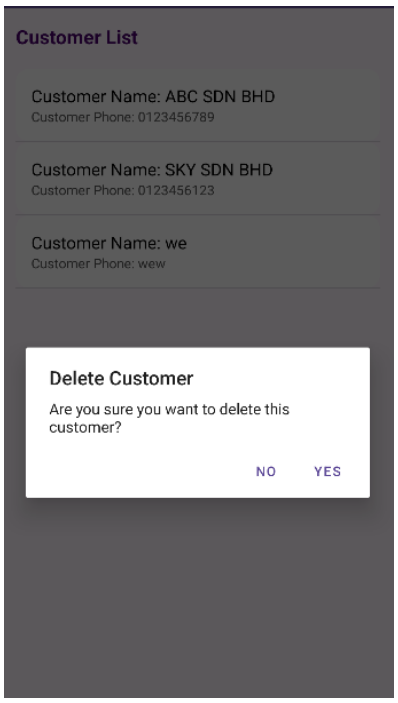


Figure 5.12: UI of Edit Customer Details.

1. **User performs a long click** on the customer item they wish to modify.
2. A **message box appears with options**: "Edit" and "Delete."
3. **User selects the "Edit" option**.
4. The system navigates the user to the **Edit Customer Details page**.
5. On this page, the user can update the **customer's name, contact information, address, and other relevant details**.
6. Once the updates are made, the user clicks the **"Update Customer" button**.
7. The updated customer details are saved, and the **customer information is modified accordingly** in the system.

Delete Customer	
  Figure 5.13: UI of Delete Customer.	• **User performs a long click** on the customer item they wish to delete. • A **message box appears with options**: "Edit" and "Delete." 1. **User selects the "Delete" option**. 2. The system **prompts for confirmation** to ensure the user wants to delete the customer item (optional step if confirmation is required). 3. Upon confirmation, the **customer item is deleted** from the list. **4.** A **success message is displayed**, indicating that the customer item has been deleted successfully.

Catalog

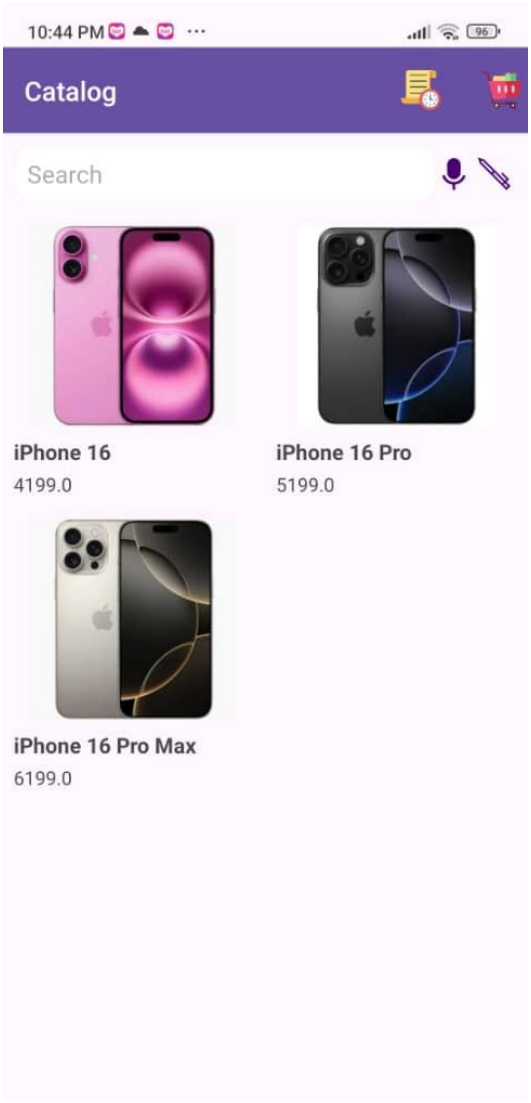


Figure 5.14: UI of Catalog.

Search Bar: Allows users to search for specific items or products.

History Button (Clock Icon): Possibly shows recently viewed items or the history of searches.

Cart Icon: Takes the user to their shopping cart, where they can review and proceed to checkout.

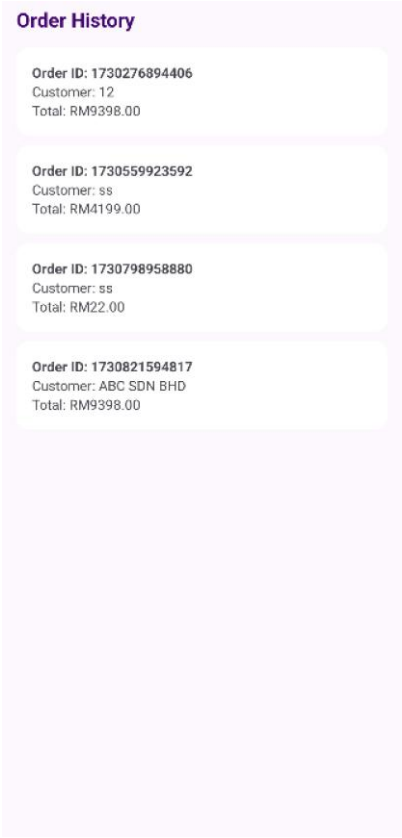
Product Display Boxes:

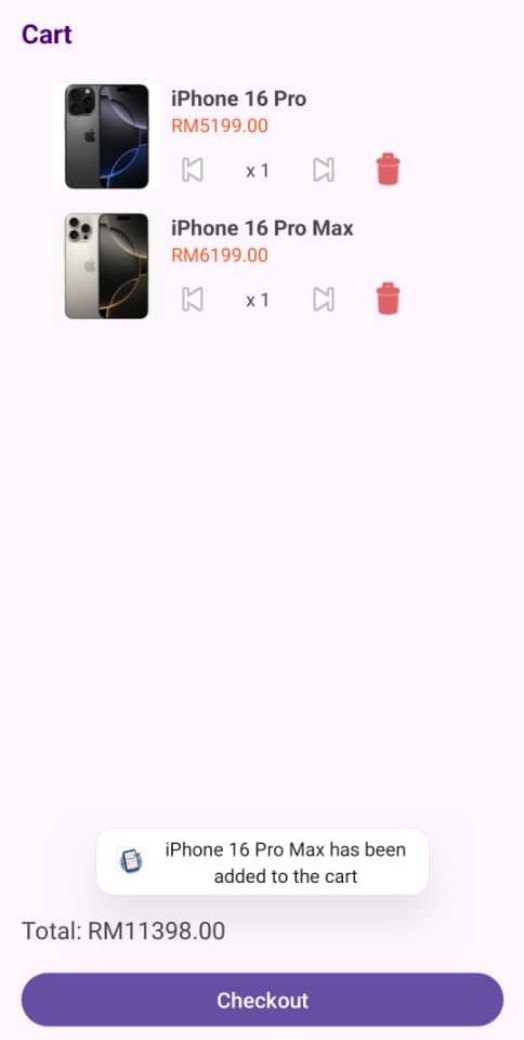
- **Image Placeholder**: Space for product images.
- **Product Name**: Name of the product (e.g., iPhone 16).
- **Price**: The cost of the product (e.g., RM 4199.00 for iPhone 16).

Price Display: Shows the price of each product.

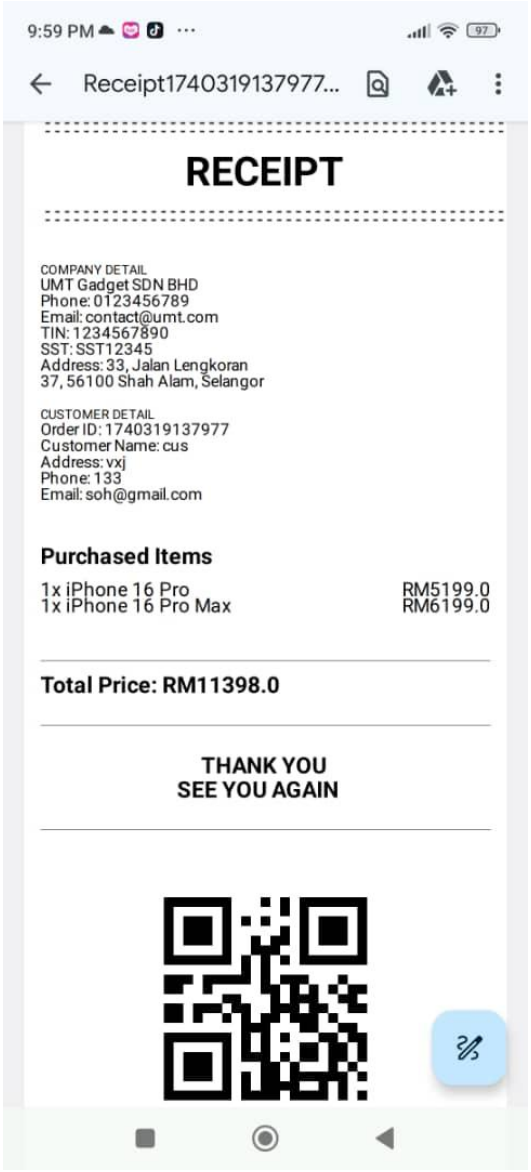
Speech Icon: Performs speech recognition function.

Pen Icon: Performs handwriting to text function

History Sales	
 Figure 5.15: UI of History Sales.	**Invoice ID**: Each item represents an invoice with a unique identifier. **Company Name/Customer**: The name of the company or customer associated with each invoice (e.g., ABC Sdn Bhd, Sky Sdn Bhd, Guest). **Amount**: The total amount associated with each invoice is displayed in Malaysian Ringgit (RM).

Cart	Stock Items:
 Figure 5.16: UI of Cart.	• Each product in the cart is displayed with an image placeholder, name, and price. • **Stock Name**: Indicates the specific product. • **Price**: Displays the price of a single unit of the product. • **Add/Reduce Button**: Add/Reduce quantity of the product. • **Delete Button**: Delete the product. **Total**: The final amount to be paid, combining the subtotal and tax, which totals RM5.00.

Receipt	
 Figure 5.17: UI of Receipt.	**Company Details:** Displays the quantity information. **Customer Details**: Displays the customer or company information (ABC Sdn Bhd with the address provided). **Order ID**: Shows the order ID for tracking purposes. **Total Amount**: The total amount due is "RM11398.00", displayed prominently in red. **Itemized List**: • **First Item**: "iPhone 16 Pro" - 1 piece, priced at RM5199.00. • **Second Item**: "iPhone 16 Pro Max" - 1 piece, priced at RM6199.00. **Share Button**: A button likely intended to share the receipt order to customer. **QrCode:** User can scan the qrcode to perform E-Invoice Validation.

Share Receipt	
 Figure 5.17: UI of Share Receipt.	**Company Details:** Displays the quantity information. **Customer Details**: Displays the customer or company information (ABC Sdn Bhd with the address provided). **Order ID**: Shows the order ID for tracking purposes. **Total Amount**: The total amount due is "RM11398.00", displayed prominently in red. **Itemized List**: • **First Item**: "iPhone 16 Pro" - 1 piece, priced at RM5199.00. • **Second Item**: "iPhone 16 Pro Max" - 1 piece, priced at RM6199.00. **QrCode:** User can scan the qrcode to perform E-Invoice Validation.

Analysis

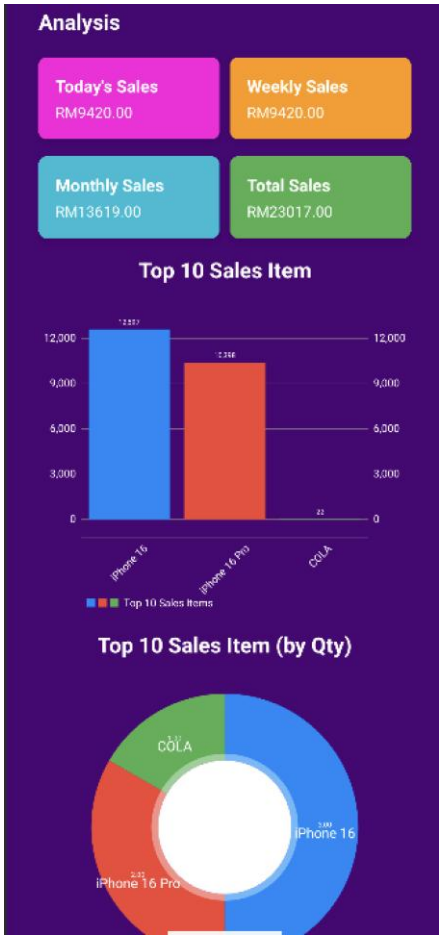


Figure 5.19: UI of Analysis.

Sales Overview:

- **Daily Sales**: Displays the total sales for the day.
- **Weekly Sales**: Shows the total sales for the week.
- **Monthly Sales**: Indicates the total sales for the month.
- **Total Sales**: Presents the total sales.

Top 10 Stock Sales:

- A bar chart visually represents the sales performance of the top 10 stock items.
- In the chart, "iPhone 16" has the highest sales, followed by "iPhone 16 Pro" and "COLA."

Voice Recognition	**Search Bar**: • A standard text search bar where users can type in their queries. **Microphone Icon**: • A large microphone icon below the search bar indicates that users can perform voice searches. • By clicking or tapping on this icon, users can speak their search query instead of typing it.

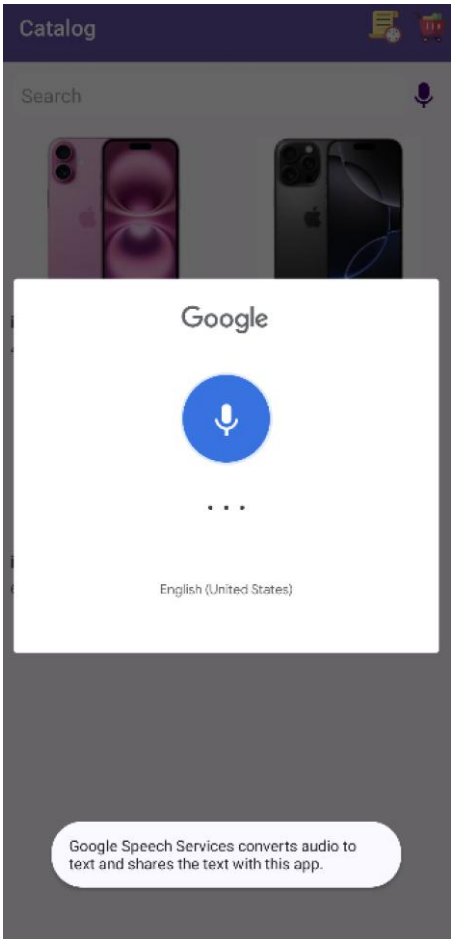
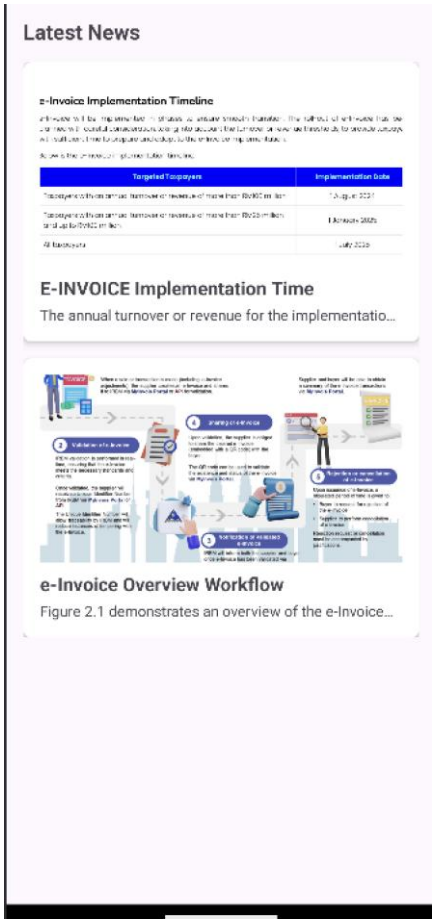
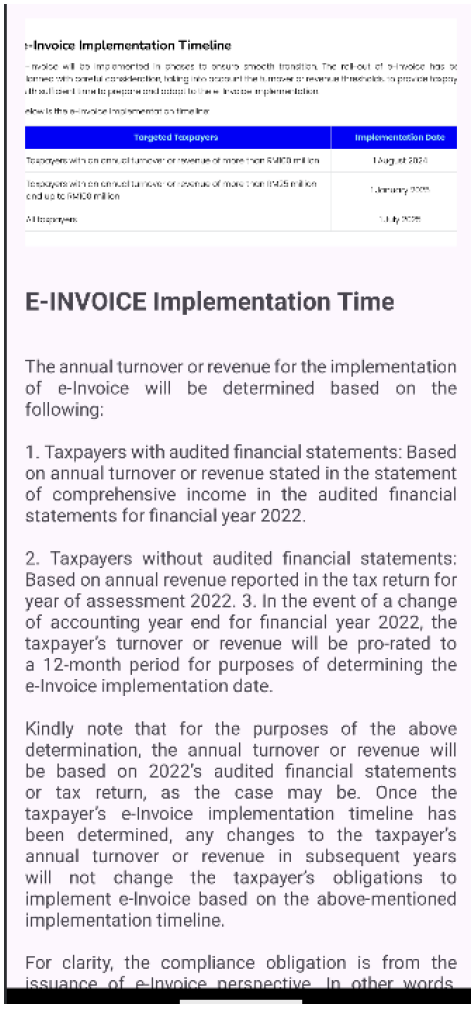


Figure 5.20: UI of Voice Recognition.

Latest News List	**Latest News List:** A List that list down the latest news of e-Invoice.
 Figure 5.21: UI of Latest News List.	

View Latest News	View the details of latest news of e-Invoice



e-Invoice Implementation Timeline

e-Invoice will be implemented in phases to ensure smooth transition. The roll-out of e-Invoice has so far been with careful consideration, holding into account the annual revenue thresholds, to provide businesses with sufficient time to prepare and institute the e-Invoice implementation.

Below is the e-Invoice implementation timeline:

Targeted Taxpayers	Implementation Date
Taxpayers with an annual turnover or revenue of more than RM100 million	1 August 2024
Taxpayers with an annual turnover or revenue of more than RM25 million and up to RM100 million	1 January 2025
All taxpayers	1 July 2025

E-INVOICE Implementation Time

The annual turnover or revenue for the implementation of e-Invoice will be determined based on the following:

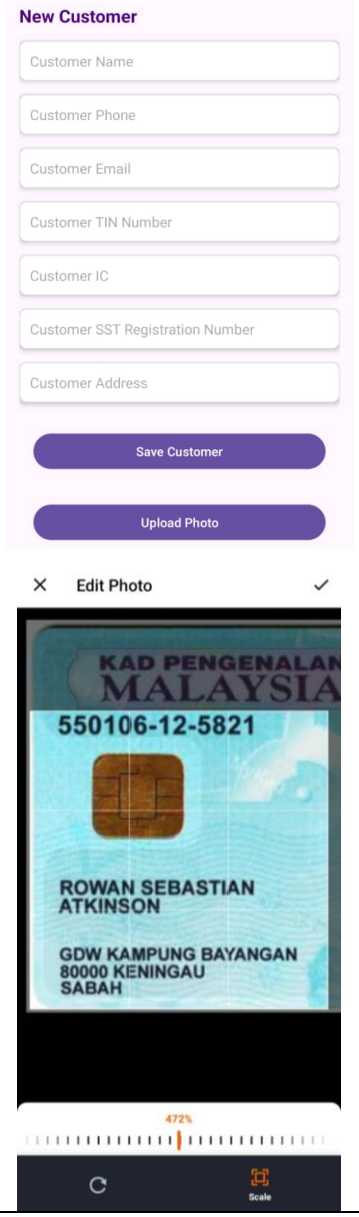
1. Taxpayers with audited financial statements: Based on annual turnover or revenue stated in the statement of comprehensive income in the audited financial statements for financial year 2022.

2. Taxpayers without audited financial statements: Based on annual revenue reported in the tax return for year of assessment 2022. 3. In the event of a change of accounting year end for financial year 2022, the taxpayer's turnover or revenue will be pro-rated to a 12-month period for purposes of determining the e-Invoice implementation date.

Kindly note that for the purposes of the above determination, the annual turnover or revenue will be based on 2022's audited financial statements or tax return, as the case may be. Once the taxpayer's e-Invoice implementation timeline has been determined, any changes to the taxpayer's annual turnover or revenue in subsequent years will not change the taxpayer's obligations to implement e-Invoice based on the above-mentioned implementation timeline.

For clarity, the compliance obligation is from the issuance of e-Invoice perspective. In other words

Figure 5.22: UI of View Latest News.

OCR Text	
	• By clicking or tapping on the "Upload Photo", users can snap/photo. • After that user need to crop the photo IC until the format shown. • The system will detect the text and auto fill in in text field.

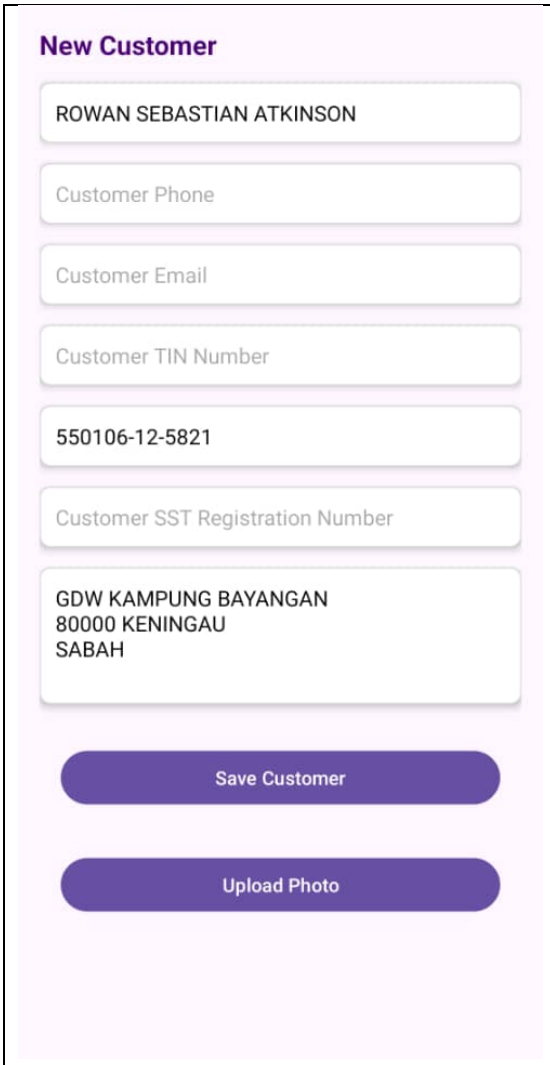
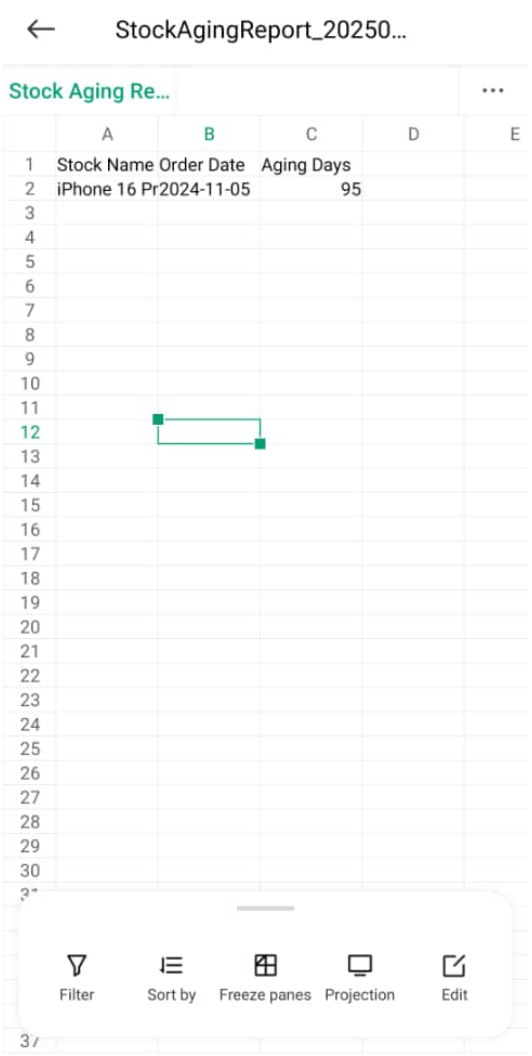


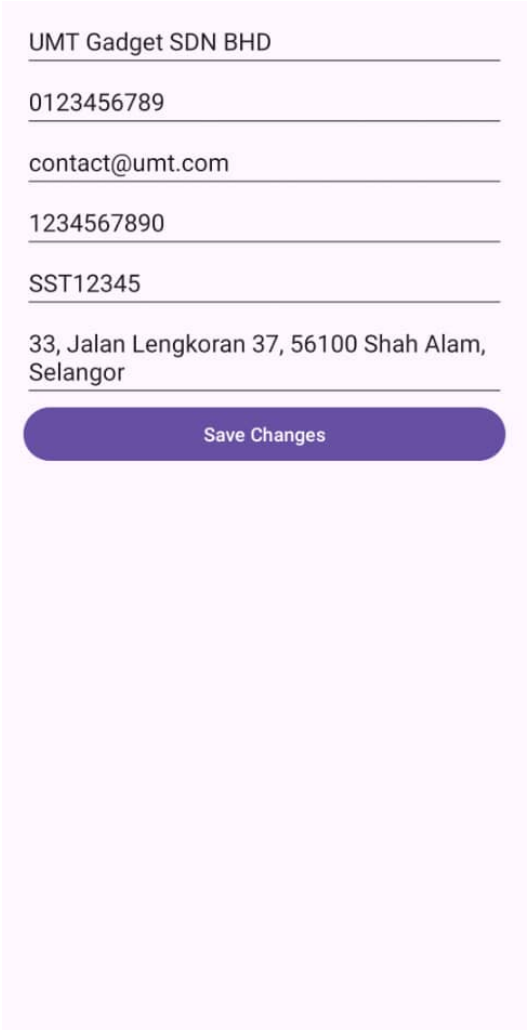
Figure 5.23: UI of OCR Text.

Stock Aging	• Stock List that show the stock name, last purchased date and aging days.
 Figure 5.24: UI of Receipt.	

Stock Aging Report in Excel  Figure 5.25: UI of Stock Aging Report in Excel Format.	• Stock Aging Report in Excel format.

Handwriting to Text	• A white board is displayed to let user performs handwriting • System will detect the handwriting and fill-in in search query.

Figure 5.26: UI of Handwriting to Text.

User Profile	
 Figure 5.27: UI of User Profile.	**Company Code**: A unique identifier for the company. **Company Name**: The registered name of the company. **TIN Number**: Tax Identification Number, used for tax-related purposes. **SST Registration Number**: Sales and Services Tax registration number. **Identification Number**: Likely refers to a IC or business registration ID. **Email**: Contact email address for the company. **MSIC Code**: Malaysian Standard Industrial Classification code, identifying the business industry. **Address**: Address of the company.

5.3 Implementation Issues and Challenges

One of the most significant challenges in implementing COME-Invoice is the difficulty in obtaining access to the government's official e-invoice API. Without this integration, the system cannot automate the retrieval and submission of e-invoices, making it harder for users to comply with regulations. Delays or restrictions in securing API access limit the platform's functionality and overall efficiency. Additionally, regulatory compliance poses another hurdle, as government policies regarding e-invoicing may change frequently. These updates require continuous system modifications, which can be resource-intensive and time-consuming.

Another major challenge is the limited digital literacy among hawkers and small business owners, who are the primary users of the system. Many may struggle with adopting new digital tools, making it essential to provide proper training and design a user-friendly interface. Furthermore, data security and privacy concerns must be addressed, as handling sensitive financial information requires robust encryption, secure authentication, and protection against cyber threats to maintain user trust.

Technical challenges also arise in ensuring reliable system performance. Many users may operate in areas with unstable internet connections, affecting real-time transactions and data synchronization. Implementing offline functionality can help mitigate this issue, but it adds complexity to system development. Additionally, integrating COME-Invoice with existing accounting and point-of-sale (POS) systems can be challenging, as businesses may already use different software solutions. Ensuring seamless compatibility requires additional customization and development efforts.

Lastly, user adoption and market penetration remain key obstacles. Encouraging small business owners to transition from manual invoicing to an e-invoicing system requires awareness campaigns, incentives, and demonstrations of the platform's benefits. Resistance to change can slow adoption rates, making it necessary to address concerns and simplify the transition process. Overcoming these challenges is crucial to ensuring the successful implementation and widespread use of COME-Invoice.

5.4 Concluding Remark

Through COME-Invoicing development a system was created to build structured order processing while managing customers and tracking inventory and ensuring e-invoicing compliance standards. The well-structured database design in the Entity-Relationship Diagram (ERD) enables different entities to work in harmony thus facilitating uninterrupted business operations. A complete automated invoicing system emerges from the integration between customer data and order logs and inventory tracking and company profile documents.

The system tracks inventory changes and generates accurate invoices through established relationships between orders, customers, stock and order items. The system maintains tax compliance by adding SST and TIN information thus becoming suitable for small businesses starting their journey toward digital invoicing.

The implemented core functionalities have proven successful yet the system will benefit from additional reporting capabilities as well as advanced stock forecasting tools and AI-based business analytics. The system usability will get additional improvements through better data synchronization features and expanded multi-device accessibility options. COME-Invoicing provides a complete business solution that streamlined operations and decreased manual labor and enhanced financial preciseness for small business owners.

CHAPTER 6

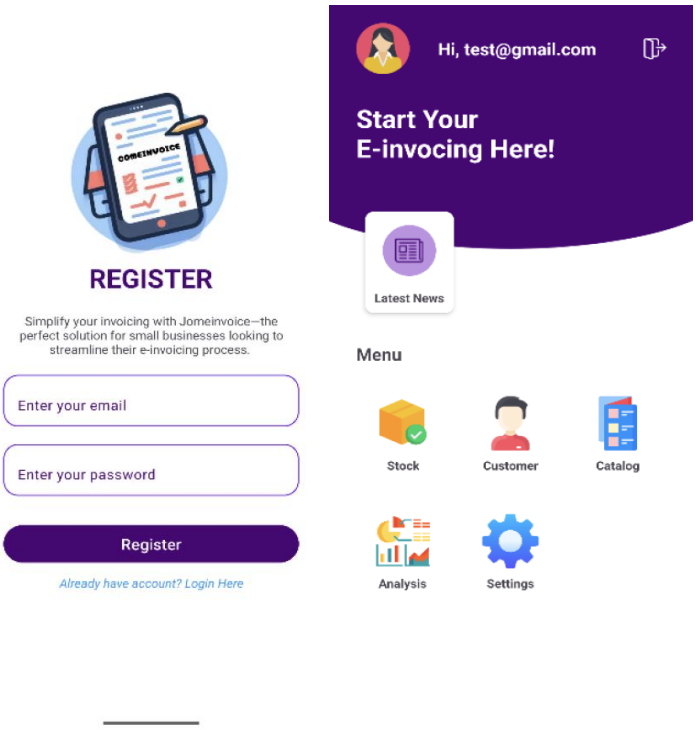
System Evaluation and Discussion

6.1 Testing Setup and Result

6.1.1 Login and Sign Out

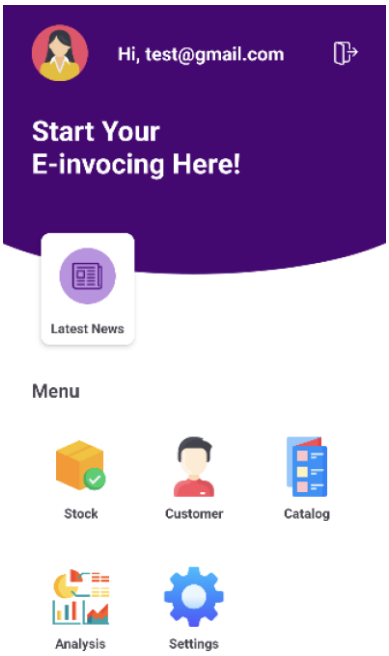
Test Case 1: Firebase Authentication Login

Table 6.1.1.1. Test Methods for Firebase Authentication Login.

Execution Steps	1. Enter the application. 2. Login with Username and Password.
Expected Result	• Redirect to the Home Screen.
Actual Output	• Redirected to Home Page. 
Pass/Failed	Pass

Test Case 2: Firebase Authentication Auto Login

Table 6.1.1.2. Test Methods for Firebase Authentication Auto Login.

Execution Steps	1. Enter the application. 2. Auto login and Redirect to Home Screen.
Expected Result	• Redirect to the Home Screen.
Actual Output	• Redirected to Home Page. 
Pass/Failed	Pass

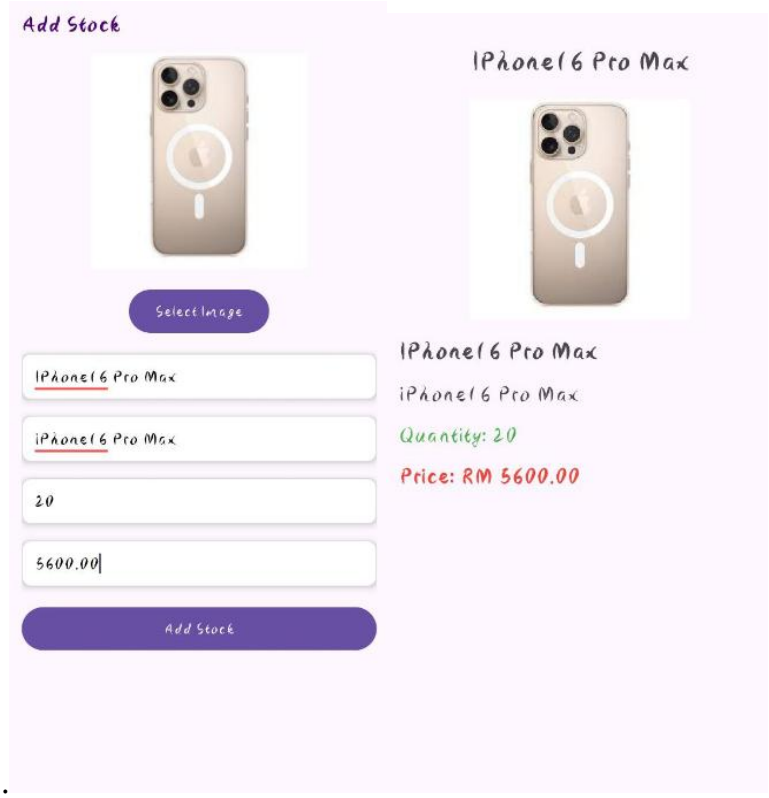
Test Case 3: Firebase Account Register

Table 6.1.1.3. Test Methods for Firebase Account Register

Execution Steps	1. Enter the application. 2. Register with Username and Password.
Expected Result	• Account Created.
Actual Output	• Account Created. 
Pass/Failed	Pass

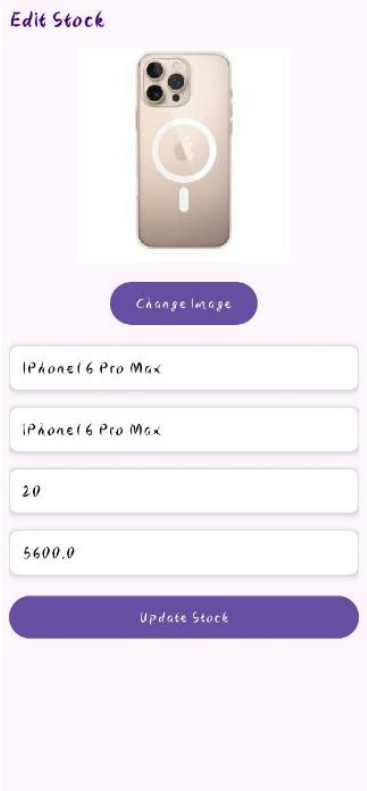
Test Case 4: Create Stock

Table 6.1.1.4. Test Methods for Create Stock

Execution Steps	1. Enter add stock page. 2. Fill in the information. 3. Click add stock button.
Expected Result	• New stock added and display in stock list..
Actual Output	• New stock added and display in stock list  .
Pass/Failed	Pass

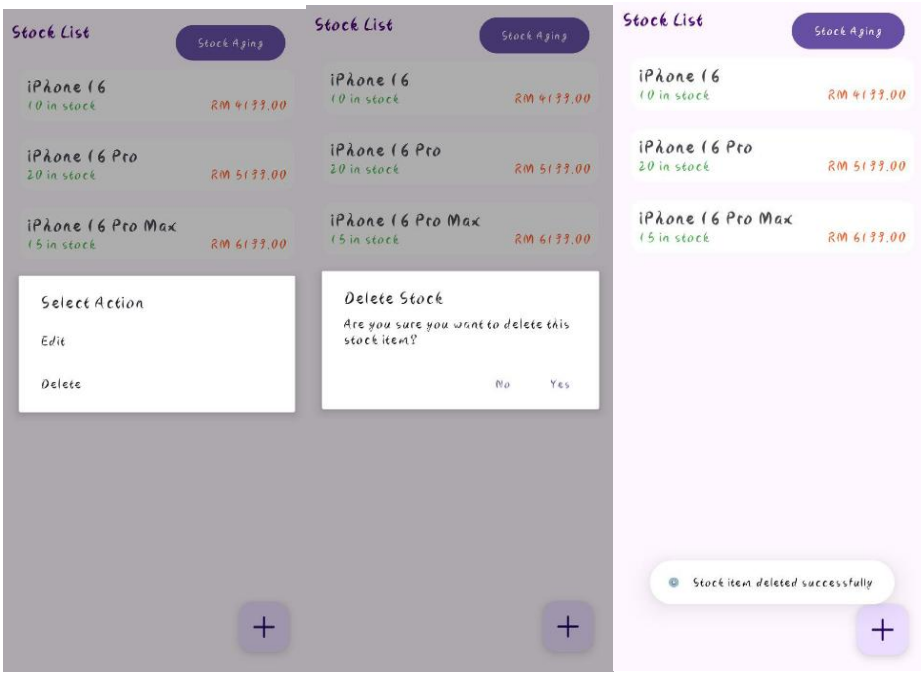
Test Case 5: Edit Stock

Table 6.1.1.5. Test Methods for Edit Stock

Execution Steps	1. Enter stock list page. 2. Long press on the stock item want to edit. 3. Fill in the information. 4. Click update stock button.
Expected Result	• Stock information updated.
Actual Output	• Stock information updated. 
Pass/Failed	Pass

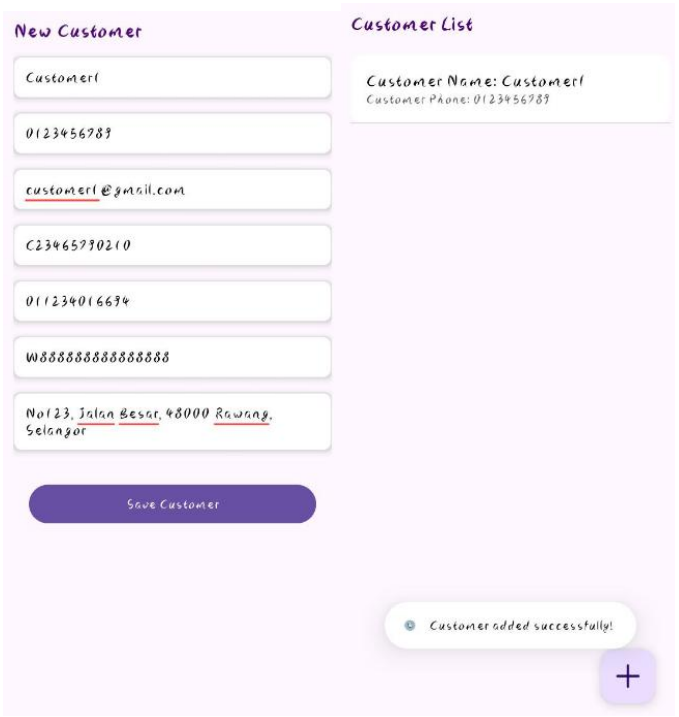
Test Case 6: Delete Stock

Table 6.1.1.6. Test Methods for Delete Stock.

Execution Steps	1. Enter stock list page. 2. Long press on the stock item want to delete. 3. Click delete. 4. Confirm delete.
Expected Result	• Stock deleted.
Actual Output	• Stock deleted. 
Pass/Failed	Pass

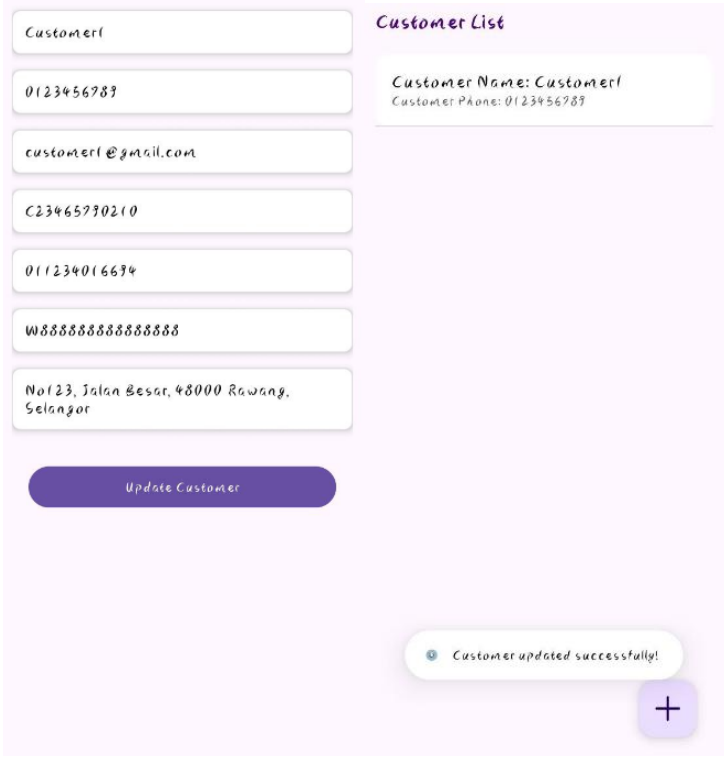
Test Case 7: Create Customer

Table 6.1.1.7. Test Methods for Create Customer.

Execution Steps	1. Enter create customer page. 2. Fill in information. 3. Click save customer button.
Expected Result	• Customer created.
Actual Output	• Customer created. 
Pass/Failed	Pass

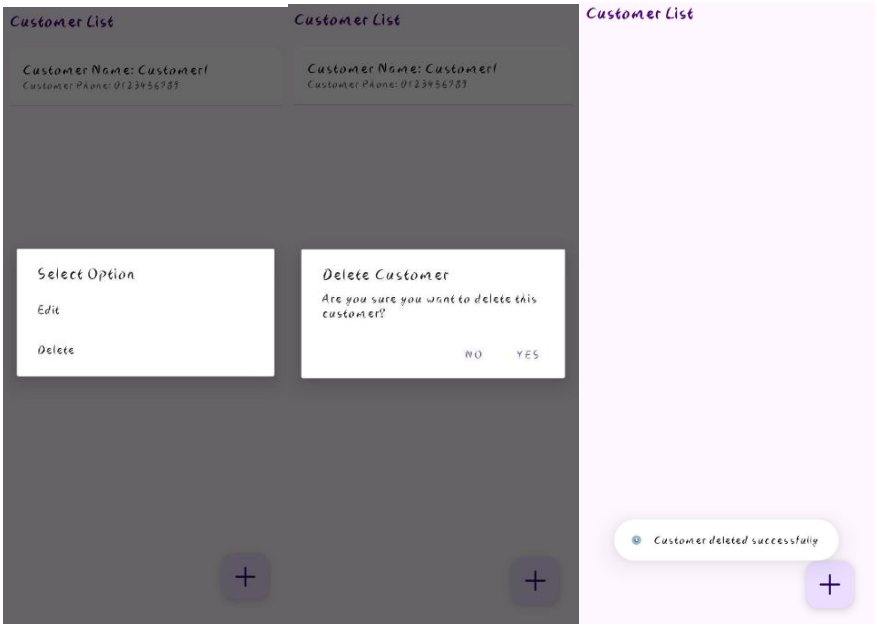
Test Case 8: Edit Customer

Table 6.1.1.8. Test Methods for Edit Customer.

Execution Steps	1. Enter customer list page. 2. Long press on the customer want to edit. 3. Fill in the information. 4. Click update customer button.
Expected Result	• Customer information updated.
Actual Output	• Customer information updated. 
Pass/Failed	Pass

Test Case 9: Delete Customer

Table 6.1.1.9. Test Methods for Delete Customer.

Execution Steps	1. Enter customer list page. 2. Long press on the customer want to delete. 3. Click delete. 4. Confirm delete.
Expected Result	• Redirect to the Home Screen.
Actual Output	• Redirected to Home Page. 
Pass/Failed	Pass

Test Case 10: Create Order

Table 6.1.1.10. Test Methods for Create Order.

Execution Steps	1. Create Order. 2. Fill in customer information.
Expected Result	• Redirect to the Receipt Screen.
Actual Output	• Redirected to Receipt Page. 
Pass/Failed	Pass

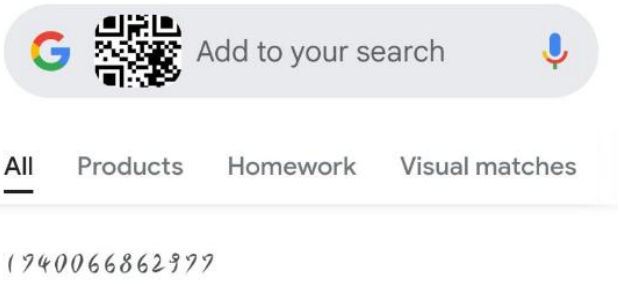
Test Case 11: Delete Order

Table 6.1.1.1. Test Methods for Delete Order.

Execution Steps	1. Enter order history page. 2. Long press on the order want to delete. 3. Click delete. 4. Confirm delete.
Expected Result	• Order deleted.
Actual Output	• Order deleted. 
Pass/Failed	Pass

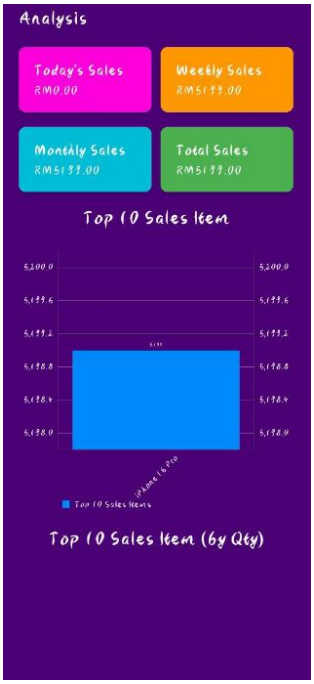
Test Case 12: E-Invoice Validation

Table 6.1.1.12. Test Methods for E-Invoice Validation.

Execution Steps	1. Click the selected order. 2. Scan the qrcode
Expected Result	• Order id displayed to show submission successful.
Actual Output	• Order id displayed to show submission successful. 
Pass/Failed	Pass

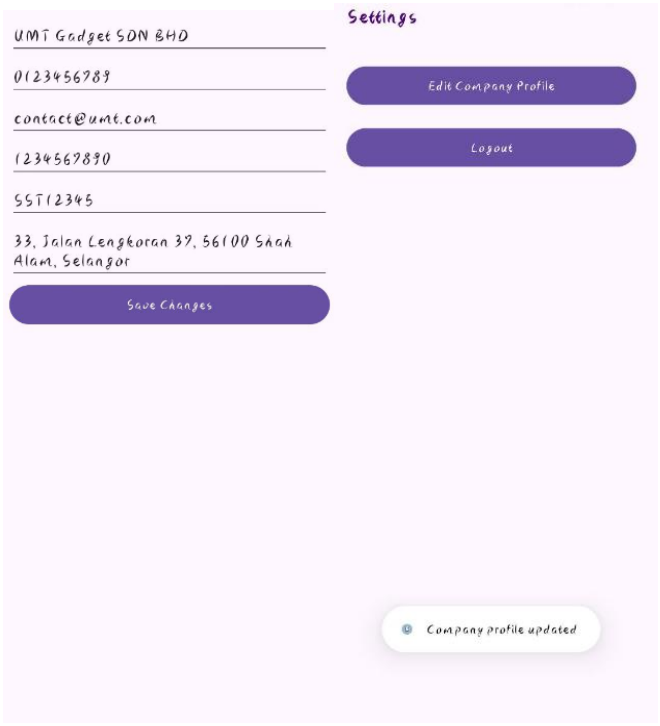
Test Case 13: View Analysis

Table 6.1.1.13. Test Methods for View Analysis.

Execution Steps	1. Enter the analysis page.
Expected Result	• Sales Analysis displayed.
Actual Output	• Sales Analysis displayed. 
Pass/Failed	Pass

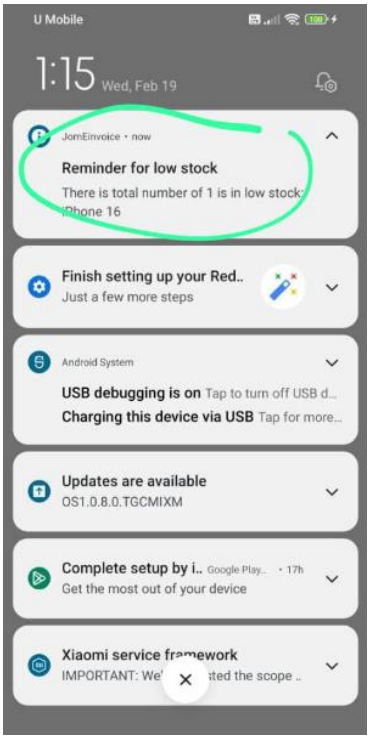
Test Case 14: Edit Company Profile

Table 6.1.1.14. Test Methods for Edit Company Profile.

Execution Steps	1. Enter the edit profikle page. 2. Update the information. 3. Click save change button
Expected Result	• Profile information updated.
Actual Output	• Profile information updated. 
Pass/Failed	Pass

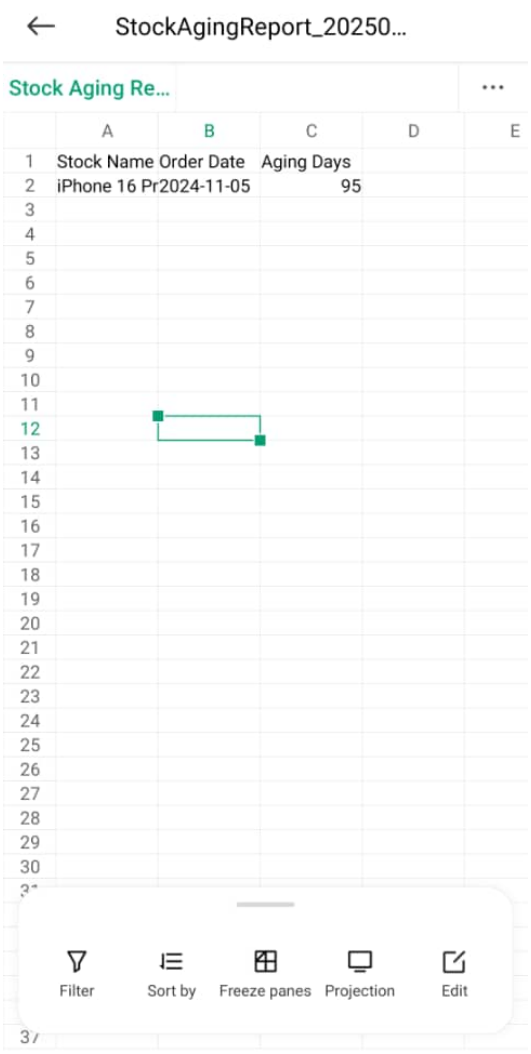
Test Case 15: Stock Notification

Table 6.1.1.15. Test Methods for Stock Notification.

Execution Steps	1. System trigger task every 12am. 2. System check the stock that less than 5 quantity.
Expected Result	• System send a notification.
Actual Output	• System send a notification. 
Pass/Failed	Pass

Test Case 16: Stock Aging

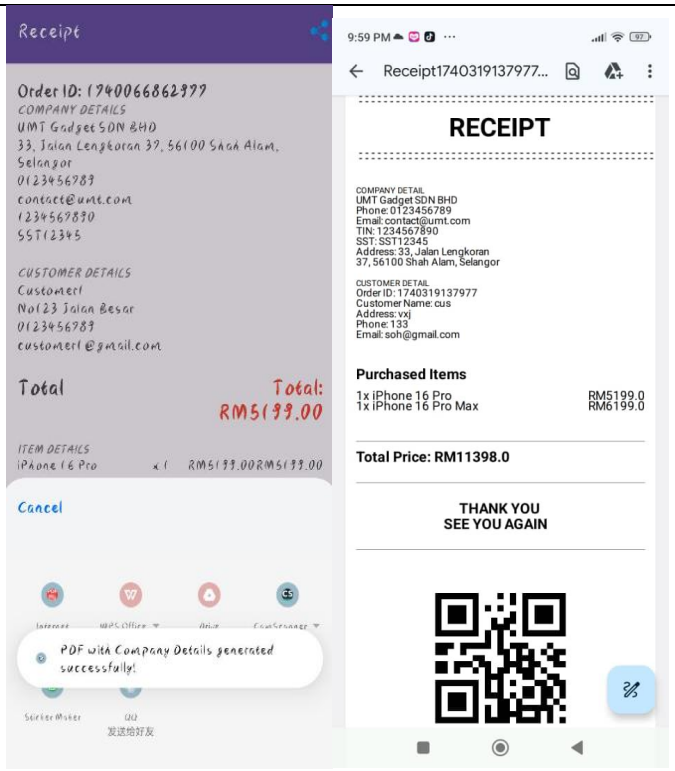
Table 6.1.1.16. Test Methods for Stock Aging.

Execution Steps	1. Click stock aging button.
Expected Result	• Excel file downloaded.
Actual Output	• Excel file downloaded. 
Pass/Failed	Pass

Test Case 17: Print Receipt PDF

Table 6.1.1.17. Test Methods for Print Receipt PDF.

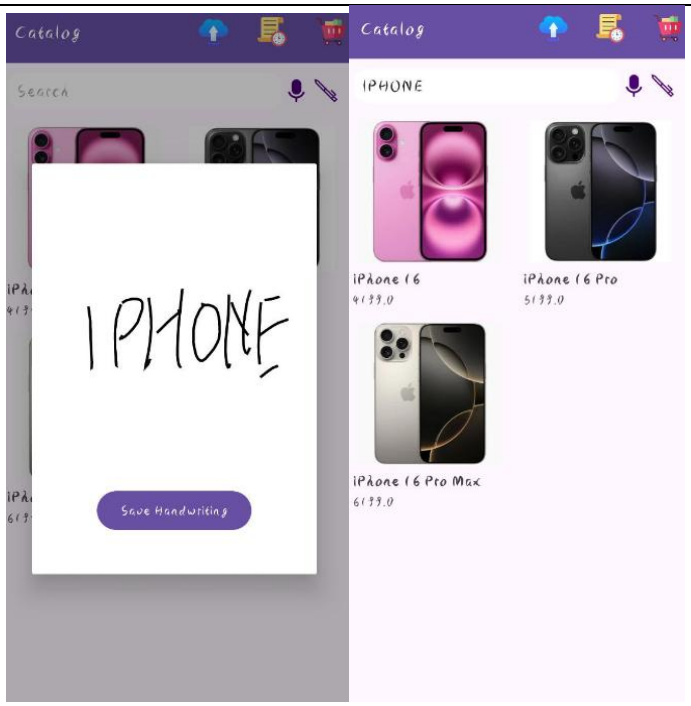
Execution Steps	1. Enter the order want to print. 2. Click share button.
Expected Result	• Generate and print receipt PDF.
Actual Output	• Generate and print receipt PDF.

	
Pass/Failed	Pass

Test Case 18: Handwriting Detection

Table 6.1.1.18. Test Methods for Handwriting Detection.

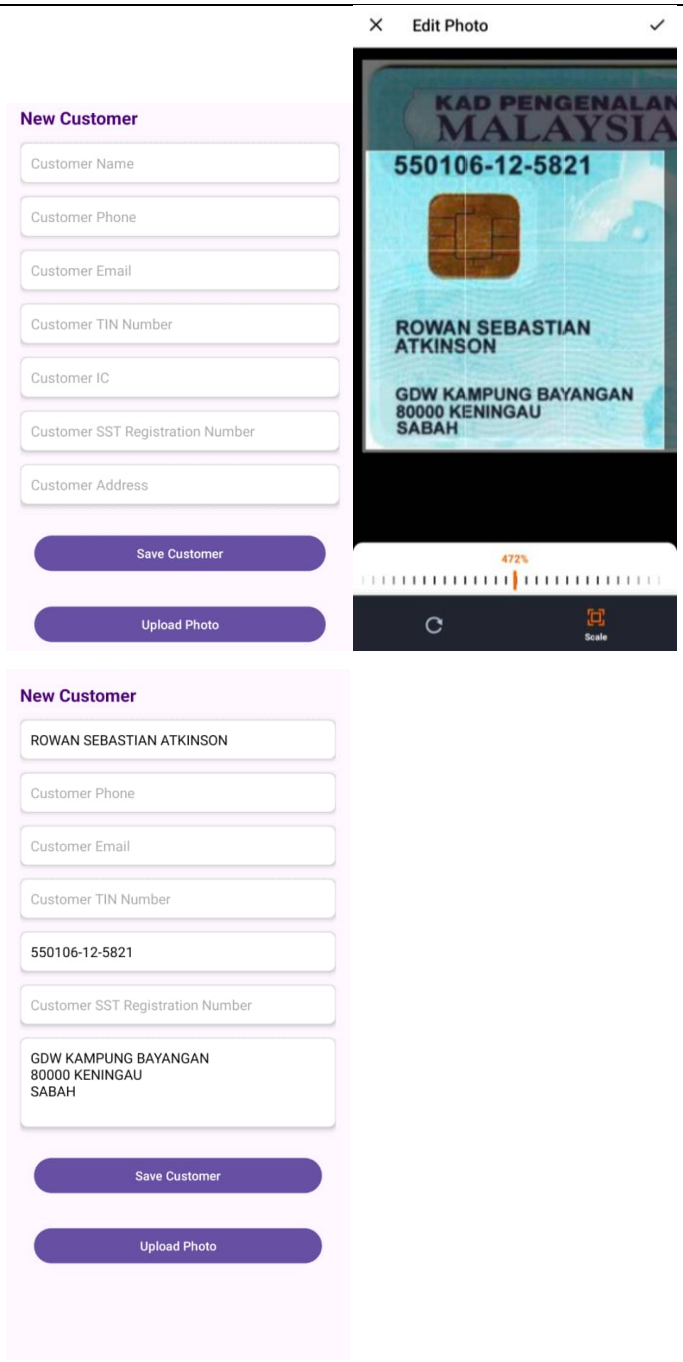
Execution Steps	1. Click handwriting button. 2. Write a sample text using finger.
Expected Result	• System detects and converts the handwritten text into digital text.
Actual Output	• System detects and converts the handwritten text into digital text.

	
Pass/Failed	Pass

Test Case 19: OCR Text Detection

Table 6.1.1.19. Test Methods for OCR Text Detection.

Execution Steps	1. Click Upload Photo button. 2. Choose a image. 3. Crop the image.
Expected Result	• System detects and converts the text and fill in the text fields.
Actual Output	• System detects and converts the text and fill in the text fields.

	
Pass/Failed	Pass

6.2 Project Challenges

Several major obstacles emerged in the development of COME-Invoicing application which demanded comprehensive testing and optimization to build an efficient end solution. The application development faced three main difficulties because of API implementation, handwriting detection precision and support for offline usage by users without Internet access.

The primary challenge involved connecting the COME-Invoicing application to the LHDN e-Invoice API which maintained by Malaysia. The Inland Revenue Board of Malaysia (LHDN) requires all businesses to meet their government regulations about electronic invoicing so the system required full compliance with these specifications. The integration of API required thorough work because the system needed to perform real-time validation and authentication while handling secure data exchange between the application and the official e-invoice portal. Guaranteeing a secure connection with API systems for invoice submission together with automatic error resolution and error tracking became essential to maintain smooth API operation.

The main difficulty arose from using Optical Character Recognition (OCR) technology to recognize handwritten documents. The sales receipt processing system in the app utilizes an AI-powered OCR to capture handwritten data therefore it required the ability to read diverse handwriting styles alongside different ink thicknesses and inconsistent writing surfaces. The performance of recognition decreased because the images showed poor quality and experienced low light levels along with noisy backgrounds. Several pre-processing operations including image enhancement, noise reduction and character segmentation were used to enhance the OCR accuracy.

Another challenge that required attention was the implementation of SQLite3 for developing offline data storage capability. The online availability of internet access in unstable regions required an app feature enabling users to manage their inventory and track sales along with invoice creation while being disconnected from the internet. The major hurdle involved in the synchronization process occurred when the application restored data between the local SQLite database and Firebase while internet connectivity returned. Conflict resolution systems required implementation to avoid problems that might occur during offline data synchronization with the cloud database. The problematic aspect involved making sure big data operations such as stock inventory and customer transactions could synchronize properly while avoiding server

load issues. The application required performance optimization to avoid slowdowns mainly on the entry-level devices which were standard equipment for small business proprietors.

6.3 Objectives Evaluation

The COME-Invoicing application achieved its development goals to support small businesses in e-invoicing regulations and sales tracking and digital literacy growth. The project evaluation assesses the accomplishment level of established targets while suggesting future improvements.

Objective 1: Promote Digital Literacy

The project first goal was to provide assistance that guided small business owners especially low-tech hawkers toward moving from traditional record keeping to digital invoice management systems. Users of varied technology backgrounds could access simpler options to enter sales data when the project added AI-powered OCR together with voice-to-text functionality. Through its Latest News section the system maintained user awareness about any government policies involving e-invoicing. An enhancement of this goal could be realized by integrating in-app tutorial guidance which would let new users understand the system better during their first experience.

Objective 2: Facilitate Compliance with E-Invoicing Regulations

The main priority of this project revolved around establishing that COME-Invoicing would adhere to all rules of Malaysia's LHDN e-invoicing system. The integration maintains invoice accuracy by requiring necessary data entry in Tax Identification Number (TIN) fields and SST Registration Number and customer information components to stop businesses from participating in non-compliance activities. Incomplete or incorrect invoices must undergo validation to prevent submission due to the system's designed process which reduces tax reporting errors. Additional refinements were necessary to handle errors and implement automatic retries throughout the implementation phase due to API response delays alongside authentication failures. The system development process should incorporate an automated alert system for invoice submission to maintain regulatory compliance through deadlines.

Objective 3: Simplify Sales Tracking

The sales tracking module carried out its intended purpose by enabling users to make daily transaction entries and inventory management and produce automatic weekly and monthly reports. The integration of SQLite3 allowed users to conduct offline business activity because it provided functionality without needing internet access. Data synchronization through Firebase operated when the system connected to the internet to perform real-time backup of all records thus preventing information loss. Through its analysis module users obtained business data insights using bar charts and pie charts to see their most popular products along with general sales metrics. Decision-making abilities of small business owners get substantially better due to this improvement.

6.4 Concluding Remark

COME-Invoicing application provides effective solutions to resolve major difficulties Malaysian small businesses encounter in e-invoicing compliance and sales tracking and digital literacy adoption. The system ensures complete regulatory compliance through the integration of LHDN's e-Invoice API so businesses can generate and submit invoices smoothly. Users with limited IT skills now have easier access to digital invoicing through built-in features that use AI OCR technology combined with voice-to-text functionality and automatic sales reporting.

This initiative presents additional opportunities to enhance performance. Strategic additions to the application including tutorial functions as well as improved AI-based handwriting recognition capabilities together with enhanced financial data analytics would optimize user interaction. The next releases of the system must prioritize several key elements such as system performance enhancement together with data synchronization enhancement and regulatory compliance adaptability capabilities.

Through this project small companies established an effective base for their digital evolution and now manage their invoices using modern technological systems instead of traditional manual processes. The ongoing refinement of COME-Invoicing will make it a dependable yet easy-to-use solution which meets regulatory standards for Malaysian small business owners.

CHAPTER 7

Conclusion

7.1 Conclusion

The COME-Invoicing solution addresses the principal hurdles Malaysian hawkers and small companies face when implementing e-invoicing through government regulations. The platform delivers a simple digital adoption solution including automatic compliance assistance and sales tracking automation which creates a smooth invoicing platform for nontechnical users. Users benefit from automatic monthly and weekly reports and real-time compliance status as well as OCR and voice-to-text tools and an easy data entry system through the application's features. The application delivers business owners user-friendly design together with multilingual support that enables operation streamlining and effortless regulatory change monitoring.

The agile framework of COME-Invoicing enables it to integrate user suggestions along with changing legal criteria for ongoing improvements. The platform maintains its effectiveness and continued usefulness in Malaysia's changing digital environment because of its adaptable features.

COME-Invoicing functions as an entry point for small businesses to start using digital solutions and artificial intelligence propositions. COME-Invoicing functions as a vital technological infrastructure to boost Malaysian small businesses' sustainable growth while ensuring their digital competitiveness in Malaysia's advancing digital economy framework.

7.2 Recommendation & Future Development

The adoption and effectiveness of COME-Invoice can be enhanced by implementing specific future improvements. The top requirement needs to focus on obtaining access to the government e-invoice API because it enables automated invoicing and maintains regulatory compliance standards. By actively engaging relevant authorities as well as through industry collaboration or pilot program participation users can speed up this process. A system design must integrate a mechanism which enables quick and automatic regulatory changes without causing interruption to user operations.

Businesses that already use AutoCount and SQL Accounting can benefit tremendously from an integrated version of COME-Invoice. Seamless integration enables users to synchronize financial data automatically which reduces manual input work and minimizes errors that arise from human interference. Integration of AI-based tax automation systems combined with intelligent expense tracking abilities would improve both functionality and user experience.

The platform's effective use requires extensive digital training together with simple tutorials for all users including multilingual support to enable hawkers and small business owners to master the platform. The adoption of the platform will expand through a simple and intuitive design approach. An improved offline feature set proves fundamental to deal with connectivity problems because it enables users to create and save invoices even when internet connections are unreliable.

Future development should center on security because protecting sensitive financial information remains important and developers should enhance data encryption as well as implement multi-factor authentication and fraud detection capabilities. The system can become more effective by adding real-time sales tracking alongside inventory management capabilities and expense reporting tools for users to obtain a complete business management solution.

Finally, market penetration efforts should include awareness campaigns, government collaborations, and incentive programs to encourage adoption among small business owners. Additional incentives such as financial assistance and trial periods

should be provided to companies for enhancing the adoption of digital invoicing strategies. A combination of system improvements and focus on these recommended measures will lead COME-Invoice toward becoming a broad e-invoicing solution that Malaysian businesses embrace.

REFERENCES

[1] LHDN Malaysia, “About e-Invoice | Lembaga Hasil Dalam Negeri Malaysia”. About e-Invoice. https://www.hasil.gov.my/en/e-invoice/about-e-invoice/ (accessed August 22, 2024)

[2] MIT Press Direct, “The Untimely Demise of the Goods and Services Tax (GST) in Malaysia: A Postmortem and the Way Forward”, The Untimely Demise of the Goods and Services Tax (GST) in Malaysia: A Postmortem and the Way Forward. https://direct.mit.edu/asep/article/23/1/1/118973/The-Untimely-Demise-of-the-Goods-and-Services-Tax (accessed August 22, 2024)

[3] Cleartax, “What is SST in Malaysia: Meaning, Exemption List, Rate 2024, Calculation, and More”. What is SST in Malaysia: Meaning, Exemption List, Rate 2024, Calculation, and More. https://www.cleartax.com/my/en/sst-in-malaysia (accessed August 22, 2024).

[4] LHDN Malaysia, “Benefits of e-Invoice | Lembaga Hasil Dalam Negeri Malaysia”. Benefits of e-Invoice. https://www.hasil.gov.my/en/e-invoice/benefits-of-e-invoice/ (accessed August 22, 2024)

[5] LHDN Malaysia, “E-INVOICE GUIDELINE INLAND REVENUE BOARD OF MALAYSIA”, Malaysia, 2024. Accessed: August 22, 2024. [Online]. Available: https://www.hasil.gov.my/media/fzagbaj2/irbm-e-invoice-guideline.pdf

[6] LHDN Malaysia, “Overview of the e-Invoice Model | Lembaga Hasil Dalam Negeri Malaysia”. Overview of the e-Invoice Model. https://www.hasil.gov.my/en/e-invoice/overview-of-the-e-invoice-model/ (accessed August 22, 2024)

[7] LHDN Malaysia, “e-Invoice Implementation Timeline | Lembaga Hasil Dalam Negeri Malaysia”. e-Invoice Implementation Timeline. https://www.hasil.gov.my/en/e-invoice/e-invoice-implementation-timeline/ (accessed August 22, 2024)

[8] Cleartax, “Impact of e-Invoicing in Malaysia: Advantages and Disadvantages”. Impact of e-Invoicing in Malaysia: Advantages and Disadvantages. https://www.cleartax.com/my/en/advantages-and-disadvantages-e-invoicing-malaysia (accessed August 22, 2024)

[9] LHDN Malaysia, “MyInvois Portal | Lembaga Hasil Dalam Negeri Malaysia”. MyInvois Portal. https://www.hasil.gov.my/en/e-invoice/myinvois-portal/ (accessed August 22, 2024)

[10] LHDN Malaysia, “About MyInvois Portal | Lembaga Hasil Dalam Negeri Malaysia”. About MyInvois Portal. https://www.hasil.gov.my/en/e-invoice/myinvois-portal/ (accessed August 22, 2024)

[11] Facebook, “SQL Account – My Software”. Facebook. https://www.facebook.com/photo?fbid=933236715472650&set=pcb.931969748932680 (accessed August 22, 2024)

[12] Jom eInvoice Sdn Bhd, “JomeInvoice.” Jomeinvoice.my. https://jomeinvoice.my/ (accessed July 22, 2024).

[13] Refrens Internet Pvt. Ltd., “Best e-Invoicing Software to Streamline your GST e-Invoices.” Refrens. https://www.refrens.com/e-invoicing-software (accessed July 22, 2024)

[14] Software Advice, Inc., “Refrens Software Reviews, Pros and Cons - 2024.” Software Advice. https://www.softwareadvice.com/accounting/refrens-profile/reviews/ (accessed July 22, 2024)

[15] Storecove, “Compliant E-Reporting in Malaysia | Peppol E-invoicing + LHDN Connection.” Storecove. https://www.storecove.com/sg/en/solutions/malaysia/?s=ga-pmax&ppc_keyword=&gad_source=1&gclid=cjwkcajwnei0bhb-eiwaa2xubmuybebtsvvyknlqobcsj_w0rp6dvlnpon_ov1vhasvpz-fbxpkeqroc3y0qavd_bwe&unbounce_brid=1721381720_7796295_cba64bd11904d8cc57b52114c6909dd9 (accessed July 22, 2024)

[16] Android Studio, “Download Android Studio & App Tools – Android Developers”. Android Studio. https://developer.android.com/studio?gad_source=1&gclid=CjwKCAjwxNW2BhAkEiwA24Cm9Fgw-A-4F3iEtbxnmNBpiUzw88Ll5n6MHtfKcSb8PYa-XYAsep2iZxoCNVUQAvD_BwE&gclsrc=aw.ds (accessed July 22, 2024)

[17] Google Cloud. “Cloud Speech-to-Text API | Cloud Speech-to-Text Documentation | Google Cloud”. Cloud Speech-to-Text API. https://cloud.google.com/speech-to-text/docs/reference/rest (accessed July 22, 2024)

[18] Google ML Kit. “ML Kit | Google for Developers”. Machine learning for mobile developers. https://developers.google.com/ml-kit (accessed July 22, 2024

POSTER

ABSTRACT

The COME-Invoicing application is designed to help Malaysian hawkers and small businesses comply with government e-invoicing requirements by simplifying sales tracking and reporting. With features like AI-powered OCR for digitizing receipts and voice-to-text for easy data entry, the app supports low IT literacy users and promotes digital adoption. Through an Agile approach, the application adapts to evolving needs and regulations, aiming to bridge the digital divide for small businesses in Malaysia.

TOOLS

SQLite Java

OBJECTIVES

DIGITAL LITERACY GAP AMONG HAWKERS AND SMALL BUSINESS OWNERS

Many Malaysian hawkers and small businesses lack digital skills and awareness, making it challenging to adapt to e-invoicing without a tool that supports their transition smoothly.

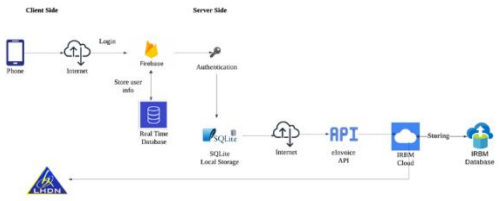
COMPLIANCE CHALLENGES WITH E-INVOICING REGULATIONS

Small business owners face difficulties keeping up with changing e-invoicing regulations, which could lead to legal issues, highlighting the need for a simple, compliant solution.

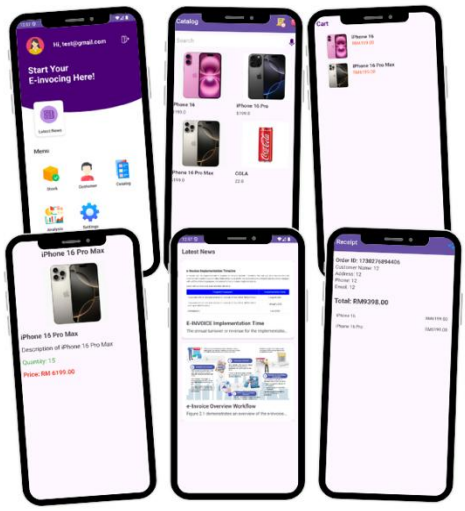
INEFFICIENCY IN MANUAL SALES TRACKING AND REPORTING

Manual sales tracking is time-consuming and prone to errors, underscoring the need for an integrated tool to automate reporting and streamline e-invoicing compliance.

METHODOLOGY



RESULTS



CONCLUSION

The COME-Invoicing app helps Malaysian hawkers and small businesses transition to digital invoicing, ensuring compliance with regulations and improving operational efficiency. Designed for ease of use, it bridges digital literacy gaps and automates sales tracking and reporting. This solution supports small businesses in navigating Malaysia's digital economy.

PROJECT DEVELOPER: Wong Sum Hui

SUPERVISOR: Ts Dr Chai Meei Tyng