System with display and advisory on payment technology preference through modern networking

BY

Tsok Jia Xuan

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF INFORMATION TECHNOLOGY (HONOURS) COMMUNICATIONS AND NETWORKING

Faculty of Information and Communication Technology

(Kampar Campus)

JUNE 2025

COPYRIGHT STATEMENT

This Final Year Project report is submitted in partial fulfillment of the requirements for the degree of Bachelor of Information Technology (Honours) Communications and Networking 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

First of all, I wish to express my most sincere thankfulness to my Final Year Project supervisor, Ts. Dr. Ooi Joo On, for his counsels, unwavering encouragement, and constructive criticisms throughout the duration of this project. His counsels and encouragements have helped play an utmost significant role to guide the way and direction to the completion and course of this work.

I would also like to thank my family members for their patience, love, and support during this scholarly pursuit. Their understanding and encouragement have given me the motivation and strength to overcome obstacles and work towards excellence.

Without the assistance of these people, I could not have done this project, and I am truly thankful to their efforts.

ABSTRACT

In the modern evolving digital economy, businesses are facing increasing pressure to accommodate multiple wireless payment methods—such as digital wallets, credit cards, and cryptocurrencies—within their current enterprise infrastructure. However, the lack of integration among these systems typically results in business inefficiencies, security vulnerabilities, and siloed customer experiences. The purpose of this project is to address these problems by establishing a single, unified platform that integrates the wireless payment solutions into the Odoo Enterprise Resource Planning (ERP) system. The solution uses Stripe for processing digital wallet transactions and CoinGate for cryptocurrency payment processes, respectively, with real-time synchronization with critical business modules like sales, inventory, and accounting. A user-friendly dashboard has been developed that displays transactions and provides strategic payment recommendations using business intelligence (BI) approach. The system design, which is driven by Odoo's modular framework, offers secure payments, automation, and expandability to allow for future enhancements. By providing backend automation along with advisory visualization, the system provides enhanced financial intelligence, reduces errors, and encourages informed decision-making. Not only does this project ease payment management, but it also facilitates the greater good of establishing secure, smart, and efficient digital payment ecosystems for next-generation businesses.

Area of Study: Machine Learning, Financial Technology (FinTech)

Keywords: Wireless Payment Integration, Odoo ERP, Payment Advisory Dashboard, Digital Wallet Integration, E-commerce Payment Solutions, AI-Powered Payment Insights, Consumer Payment Preference Analysis

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

API	Application Programming Interface
ERP	Enterprise Resource Planning

Chapter 1
Introduction

As organizations are changing their business activities to accommodate the digital economy, they are under more pressure than ever to continue to provide wireless payment options and adapt to the new consumer reality. Integrating these payment technologies into and existing system, as you do with an ERP system such as Odoo, is extremely challenging. Separate payment systems in a business can led to inefficiencies, security issues, and no centralized way to review the business' transactions, meaning that businesses have no visibility over which consumers are purchasing what.

The aim of this project is to tackle these issues and create one connected system that integrates a cryptocurrency payment method into the Odoo ERP system and takes advantage of the FinTech solutions that provide automated transaction processing, synchronized in real time, and an integrated dashboard to visualize the financial position of the business and ultimately the consumer behavior. The solution should streamline business operations, enhance security, and provide organizations with access to the information required to make efficient decisions: to get ready for success in an ever-growing digital FinTech-based economy.

1.1 Problem Statement and Motivation

Problem Statement

As the digital economy evolves rapidly, vendors are required to provide their customers with multiple wireless payment options (credit cards, cryptocurrency payments, etc.). However, vendors are unsure which payment option for their customers prefer leading to confusion in decision-making, wasted resources, and inefficiencies (e.g. the proliferation of payment systems on platforms, difficulty tracking financials across different payments systems, lack of security and data theft). Embedding this wide variety of payment technologies into enterprise resource planning systems (i.e. Odoo, etc.) is a significant challenge.

Many businesses still must rely on manual processes or use separate systems for multiple payment methods which creates delays, duplication, lag time for payment records etc. and

then goes on to unnecessarily hold up the businesses cashflow and operations Because of the lack of integration and information, the business may not be able to determine whether they are meeting the needs of the customers with their payment mechanism, whether they are addressing financial concerns or if they are protecting sensitive customer data with the possible increasing threat of compromised systems. All of these issues affect operational efficiency, maintaining financial transparency, and customer satisfaction.

Motivation

This project is focused on simplifying and centralizing wireless payment integration with the Odoo ERP platform and will alleviate operational and decision-making challenges for businesses. Vendors are seeing demands for solutions that simplify the adoption of the most suitable payment solutions while eliminating the burdensome costs associated with managing several disparate platforms.

As the project will integrate payment methods including card payments, and cryptocurrency gateways into a single ERP, it will remove inefficiencies, automate transaction synchronization, facilitate real-time updates, and access persistent sale accounting inventory data across business modules. It will enable AI (artificial intelligence) insights with payment data to enhance decision-making abilities for vendors by revealing customer preferences and automatically suggesting actionable next steps.

Furthermore, the project will develop a single dashboard and an easy-to-navigate interface that has integrated AI insights to allow businesses to visualize, predict and receive alerts in real-time as payment trends occur. This ability will ultimately position businesses to improve operations, lower costs and increase customer satisfaction. Wouldn't one centralized "place" to manage operations and immediate AI-driven insights create a competitive advantage over workflows with disparate platforms and no data and insights? We will ensure operational effectiveness, cost savings, and scale that reliably and securely completes transactions in a competitive digitally demanded marketplace with automations, security, and intelligent analytics.

1.2 Objectives

The primary focus of this project is to develop a fully integrated system for the modern-day wireless payment technologies for the Odoo enterprise resource planning (ERP) system. Specifically, the project will integrate Stripe for card payments and CoinGate, for cryptocurrency payments into a unified system so that businesses can process various payment types on one platform. The solution will also provide a custom dashboard that incorporates artificial intelligence (AI) to help analyze payment data, forecast trends in payments, and enable businesses to leverage real-time insights for effective decision-making.

Real-time connectivity between the wireless payment operations and the back-end ERP modules is one of the main objectives. Payment details will be received via the Odoo APIs, and the payment data will automatically be captured, recorded and updated in the relevant modules i.e. sales, accounting and inventory. This will substantially eliminate the need for manual data input and significantly reduce errors and delays in payment reporting and create accurate and synchronized financial records across the enterprise.

The second project objective is to provide an engaging centralized dashboard to view consumers' payment behaviours, transaction history and financial summaries. The dashboard will use AI to not only present data but also provide intelligent insights and forecasts based on that data, enabling businesses to capitalize on changing payments trends.

In relation to security, both Stripe and CoinGate are PCI-DSS compliant. In addition, they use TLS encryption and tokenization to ensure that all transaction processing is conducted in a secure, encrypted environment. This not only protects the data associated with transactions but also provides experts with the security of the integrity of these transactions within the integrated system for both businesses and their customers. Consequently, if the project fulfills its objectives, it will simplify payment processing, improve operational efficiencies, reduce overall costs, and enable businesses to have information they can act upon to enhance customer satisfaction while remaining competitive in the digital economy.

1.3 Project Scope and Direction

Through this project, we will deliver a valid software solution that embeds multiple wireless payment technologies, such as digital wallets, Stripe payments, and Coingate for cryptocurrency payments, into the Odoo enterprise resource planning (ERP) system. The software will support real-time transaction processing, and it will allow for automatic synchronization between many important business modules (sales, accounting, inventory), and it will also be able to secure data handling through encrypted gateways and certified APIs.

Additionally, we will provide a consolidated, easy to use dashboard that will visualize trends in payment methods, transaction summaries, and reports about customer behaviour. The solution will provide ease of payment management; it will improve operational efficiency and facilitate firms to have real-time views of their financial activities. This project will be software integration onto the current platforms therefore will not require new payment means or hardware systems to be established.

1.4 Contributions

This project will have a major impact on businesses through the simplification of payment processes and the protection of financial transactions, making them efficient and convenient to use. The integration of digital wallets and cryptocurrency payment solutions into Odoo ERP systems will enable businesses to streamline payment processing, reduce human errors, and achieve real-time synchronization in core business modules such as sales, accounting, and stock. This will not only be cost- and time-effective but also improve decision-making and fiscal transparency, allowing companies to operate more efficiently and respond quickly to market trends. With more and more companies moving towards electronic payment mechanisms, the system is a much-needed step towards the adoption of future-proof technology that enhances customer satisfaction and operational efficiency.

The value of this project lies in its ability to simplify and secure the complicated world of electronic payments. As the global economy places increasing reliance on mobile and online payments, integrating these payment technologies into current ERP systems is a critical need for businesses to remain competitive. This solution offers an end-to-end solution that

addresses both the security needs and the functional inefficiencies of older payment processing. Lastly, the project not only meets business needs but also contributes to the broader initiative towards secure, automated financial systems that set the stage for faster, more secure payments to be widely accepted. Businesses and readers alike will see the benefit in a solution that keeps them at the forefront of what is achievable with technology, offering enduring value and secure, efficient operation in a growing cashless world.

1.5 Report Organization

This report has been organized into seven chapters in order to present the project in a logical and clear way. The first chapter describes the project background, a brief outline of the objectives, as well as scope and significance. The second chapter reviews the literature regarding ERP systems, payment gateways, cryptocurrency demonstrations (adoption) and AI dashboards, as well as theoretical gaps that this project addresses. The third chapter talks about the system methodology overview, outlining the development approaches, use case analysis, and workflow. The fourth chapter provides a thorough description of system design, including an overview of architecture, data flow, database structure, and integration strategies. The fifth chapter details the system development, which includes the software set up, payment gateway integration and AI dashboard development, as well as some challenges along the way. This information is further elaborated on in Chapter 6, with an evaluative analysis covering areas of system performance, functionality and security within the ERP demonstration system (findings and lessons learnt). Chapter 7 finalizes the report synthesising the findings and contributions to the research community, providing recommendations for further scalability and application.

Chapter 2
Literature Review

2.1 Decision-making support in engineering design based on collaborative dashboards: integration of business intelligence techniques

Louhichi et al. [1] introduce a decision support framework that integrates collaborative dashboards enhanced with Business Intelligence (BI) techniques to support decision-making in engineering design. Their framework consolidates heterogeneous data sources into key performance indicators (KPIs) and provides visualization interfaces for multiple stakeholders to view current states and trends. The dashboard framework also supports real time or near-real-time updating, allowing for a quicker and more informed response to operational issues. The collaborative aspect allows multiple actors (design engineers, project managers, possibly clients) to view common visualizations, increased the speed of coordination and communication [1].

Although Louhichi et al. do not directly address digital payment systems, much of their thinking can be applied. Their discussion of BI dashboards aligns with FYP's objective of building a visualization system in Odoo to help stakeholders understand technology preferences for payment options. Just like in engineering design (where different aspects such as cost, performance, timelines, etc. have to be visualized), different factors require visualization for digital payments, including number of transactions, payment types used, payment gateway fees, user satisfaction, and latency. Thus, Louhichi et al. [1] represent a strong benchmark of apprpaches to combine real-time dashboards with multiple data source and collaborative stakeholder input.

More recent research provides additional underpinning and precision. For example, Assessment of the influence of cybersecurity threats on digital banking adoption (2025) reaffirms that consumer trust is one of the greatest factors in adoption. Another lens in the field, Behavioral Study of Dashboard Mechanisms (2025), demonstrates how the design of a dashboard can contribute to user satisfaction and ultimate decisions by reducing cognitive loads in making inferences, and improving the accuracy of the inferences that are made. Both

sources reinforce the idea that, in addition to the manner of visualizing the information, interface design, feedback in real-time, trust/security, and user preference are key considerations.

2.1.1 Strengths

1. **Data Aggregation from Multiple Sources & KPI Focus:** Louhichi et al's work illustrates how data from disparate sources (different modules) can be synthesized into meaningful Indicators of Performance or KPI's. In the case of the preference dashboard, it would mean that payment algorithms, transaction logs, and maybe even a survey of users can come together to form some aggregate indicator of transactions (e.g. "most popular method", "cost per transaction", "delays / failures", etc..).
2. **Real-Time / Dynamic Visualization:** When people can see the most current data, they are generally able to make decisions faster and act on those decisions faster. Your FYP could visualize an ongoing change in payments during peak seasons (e.g. 11.11, 12.12), and businesses could react by deciding to promote or optimize which payment gateways to promote.
3. **Collaborative Interpretation:** The dashboards are not just an informational display - they are meant to facilitate shared decision making by multiple stakeholders. For your FYP that might be shop owners, finance manager, and IT / administrative staff using the same dashboard to decide what payment method should be promoted or troubleshot.
4. **Empirical / Real-World Case Studies:** The framework proposed by Louhichi et al. is validated in real engineering settings suggesting it can be implemented in practice rather than just a theoretical tool. Adding that to the literature on trends in payment gateways, interface design, security provides you with a depth of evidence in support of your design decisions.

2.1.2 Weaknesses

1. **Domain Mismatch/Absence of Payment Data Issues:** Louhichi et al. are focused on engineering design, not finance or payments. They do not deal with payment gateway fees, transaction failures, user preference among payment types, or cryptocurrency volatility.

2. **Limited Advisory/Predictive Features:** Their dashboards are more of descriptive monitoring systems. The dashboards do not seem to forecast predictive analytics such as offering suggestions of which decisions to make based on trends. (i.e., "if payment method a usage drops move marketing efforts toward b").
3. **User Preference & Behavior Modeling Weakness:** The paper discusses little to no ways the system learns from or makes alterations to user behavior or preferences. In payment systems, user experience, satisfaction, and trust are important for users to adopt any system. There have also been recent studies showing that security was a big deterrent to crypto payment adoption globally.

2.1.3 Gap

1. **Metrics Specific to the Payment Domain:** Louhichi et al. write about KPIs that are directly related to engineering design, while my FYP requires payment-specific metrics that may include gateway fees or charges, rates of payment method usage, rates of transaction failures, and measures of consumer satisfaction.
2. **Interface and Visualization Design:** They do not deal with the optimization of UI/UX or development of user-feature friendly dashboards, but my FYP aims to develop a clear and interactive visualization interface in Odoo.
3. **Regulations and Security Compliance Requirements:** The article does not address industry regulations/standards for security, compliance (PCI DSS, crypto regulations), or privacy standards. My FYP requires the ability to integrate secure payments from third-party payment gateways (e.g. Stripe, Coingate) and remain compliant with these regulations and standards.

2.2 Implementation of odoo based on enterprise resource planning system with sales and purchasing module using rapid application development

ERP systems have emerged as important solutions for small and medium-sized enterprises (SMEs) wanting to incorporate their sales, procurement, inventory, and finance management into one platform [2]. Odoo is one of several ERP platforms that has received significant interest from SMEs because of its open-source nature, modularity, and affordability [2], [13].

Utami [2] describes a case study that implemented Odoo's sales and purchasing modules in an SME in Yogyakarta, Indonesia. Using RAD (Rapid Application Development) as the

development methodology, Odoo's iterative, prototyping approach and ongoing stakeholder participation aided in improving order tracking, data accuracy, and efficiency by tackling challenges associated with manual data handling and breaking workflows. Odoo also improved asset monitoring status and workflow automation for the Al Rajhi Endowment after adopting Odoo [3].

A significant contribution of the studies mentioned is that they illustrate how Odoo can be both scalable and usable in SMEs with resource limitations. Both works indicate that using iterative development cycles produced very usable and effective ERP solutions, justifying Odoo as a tool on top of which further system innovation could occur such as developing dashboards to be viewed in real-time or developing a decision-support system.

Nonetheless, current applications are still limited to basic ERP functions such as sales, procurement, and inventory management, and do not incorporate advanced functionalities such as AI-based payment advisory applications, real-time analytics, or multi-gateway payment functionality. These limitations indicate that there are opportunities to extend Odoo beyond its core ERP capabilities to meet current business requirements more fully, particularly in the area of e-commerce and digital payments.

Despite the existing research demonstrating the benefits and versatility of Odoo, the aim of the proposed FYP is to resolve these issues by embedding real-time online payment advisory dashboards, multi-gateway options (e.g., Stripe, CoinGate), and high-level visualization techniques into the features of the Odoo environment for improved decision-making in fast-paced digital payment ecosystems.

2.2.1 Strengths:

1. **Demonstrated ERP Benefits for SMEs:** The study provides evidence of Odoo's ability to digitize and improve core business processes for SMEs with limited resources [2], therefore confirming Odoo as a viable platform for transaction processing, which aligns directly with the FYP's goal of developing real-time visualization dashboard and advisory system.
2. **Utilization of Rapid Application Development (RAD):** The iterative, user-centered RAD methodology made sure that the system evolved according to operational needs

instead of defined specifications [2]. This provides rationale for the use of agile principles in current FYP to refine and improve the system continually.

3. **Operational Improvements:** Reporting on tangible operational improvements, including faster transactions, fewer manual errors, and improved inventory management, further provides evidence for ERP's ability to enhance SME operations and organizational workflows [2].

2.2.2 Weaknesses

1. **No Payment Systems Integrated:** The study exclusively focuses on the traditional ERP modules and does not cover the integration of the modern payment systems such as Stripe, CoinGate, or other wireless payment systems.
2. **Single Case Study Limitation:** The research was conducted in one context of a local SME (Yogyakarta, Indonesia), meaning there is no evidential equivalence in scaling it up to larger businesses, or even across an international e-commerce ecosystem.
3. **No Predictive Analytics or Advisory Modules:** Mechanisms for real time decision making, forecasting or advisory modules were not discussed in the study - thereby limiting its relevance to any data-driven financial ecosystems.

2.2.3 Gap

1. **Absence of Integration with Multiple Gateway Payment Systems:** The study primarily emphasizes the core ERP modules focused on sales and purchasing, without considering integration with contemporary payment systems like Stripe, CoinGate, or wireless payments. For SMEs that are e-commerce businesses, a multi-gateway integration provides for seamless business transactions with diverse payment methods based on customer's payment preferences.
2. **Absence of Predictive Algorithms and Real Time Analytics:** Although Odoo provides centralization of data, the study does not consider real time analytics or other algorithms to give users insights into data to take action for informed decision making. The goal of the FYP is to implement advisory modules that are supported by AI that provide businesses with consideration for their optimal payment methods when busy and hindering payment performance based on prediction for transaction processing.

2.3 Integrating payment solutions into online marketplaces

The increasing popularity of online marketplaces has highlighted the need for unified, secure, and user-friendly payment systems. In this shift to technology, Stripe has emerged as a darling Payment Service Provider (PSP), acclaimed for its ease of integration, robust API framework, and secure handling of online transactions. Oat [4], in her master's thesis, presents a step-by-step description of how Stripe and other PSPs like it can be effectively implemented in online marketplace platforms for efficient transactions on both sides, that is, for buyers as well as sellers.

Oat establishes Stripe's technical foundation, from tokenization to native PCI DSS compliance, and thus forms the optimal choice for developers to provide secure yet flexible integration plans. The thesis emphasizes the strengths of Stripe with respect to developer experience, detailed documentation, and plug-and-play building blocks, all intended to simplify development and reduce security risk. These aspects are particularly significant for projects like the one in development here, which involves integrating multiple wireless payment methods—including Stripe—into a broader system for visualizing and advising payment preferences.

The applicability of this research to the current Final Year Project lies primarily in the shared focus on user-focused design and backend automation. Oat demonstrates how payment APIs like Stripe can be used to automate streams of payments, track transactions in real-time, and support multi-platform access. All of these capabilities complement the project's use of the Odoo platform, where backend automation and visualization via dashboard are key.

Moreover, the thesis touches on relevant business environments and decision-making processes in the selection of payment gateways, which is a supplement to advisory purposes of this FYP in guiding users to pick the most suitable payment technologies.

In addition, Oat's study emphasizes that designers should develop modular systems which will be capable of responding to future pay technology advancements. Not only introducing Stripe as a payment processing solution but also as a strategic building element for constructing scalable and maintainable platforms, the research provides technical as well as

conceptual support to the development of an intelligent system with the capability to visualize payment preferences and offering suggestions on their implementation.

2.3.1 Strengths

1. **Comprehensive Technical Foundation:** Describes Stripe integration to the level of tokenization and PCI DSS compliant implementations, which provides a solid security and implementation foundation [4].
2. **Developer Experience and Modularity:** Describing API architecture, documentation, and modular systems provides simple deployments, while ensuring scalability for application [4].
3. **Backend Automation and Real-Time Monitoring:** Research provides support for automation and monitoring transactions, which will align directly with the FYP objectives, including data visualization and advisory features [4].

2.3.2 Weaknesses

1. **Limited Operational Understanding:** Practical issues such as transaction failures, fraud detection, and regulatory compliance across regions are mostly unexplored in the research [4].
2. **Developer-Focused Approach:** The research emphasizes technical integrations but lacks UX design and integration under a real-world business workflow [4].

2.3.3 Gap

1. **No Advisory or Decision-Support Mechanisms:** Although the research looked at technical integration, the research didn't include real-time advisory dashboards or predictive analytics on which payment method is the best fit. This FYP emphasizes the integration of visualization dashboards, AI-based advisory tools, and multiple gateway payment integrations (e.g. Stripe, CoinGate) into Odoo to aid decision-making in a dynamic e-commerce environment.

2.4 Cryptocurrency as a method pf payment in the tourism sector

The rapid adoption of blockchain technology and cryptocurrencies has tremendously transformed the payments landscape among many industries, ranging from e-commerce to tourism. As digital currencies are increasingly integrated into the mainstream, businesses are shifting more toward platforms such as CoinGate in order to support secure, swift, and boundary-less transactions [5].

Within the tourism industry, traditional payment system, credit cards and bank transfers have remained the standard procedure. However, these come with inherent drawbacks such as transaction charges, currency conversion charges, and the possibility of fraud. Cryptocurrencies, supported by decentralized blockchain technology, offer a competing alternative that promises increased security, reduced charges, and faster settlement times [5].

The World Tourism Organization and travel website Destinia have questioned and discovered a seismic shift in the behavior of customers: approximately 80% of the customers showed their wish to pay for travel expenses with digital currencies [5]. Such strong demand from customers is a broad tendency towards the digitalization of payment and testifies to the growing popularity of cryptocurrencies as an accessible alternative. Platforms like CoinGate are well-positioned to serve this demand by enabling businesses to accept crypto payments seamlessly.

Hotel Name	Country	Year of Adoption	Payment Technology
Bobby Hotel	USA	2021	BitPay
The Kessler Collection	USA	2021	BitPay
Dolder Grand	Switzerland	2019	Inapay App
Sri Panwa Phuket	Thailand	2021	BitcoinWide
The Pavilions Hotels & Resorts	Hong-Kong	2021	Coindirect
The Chedi Andermatt	Switzerland	2021	Worldline

Table 2.4.1 Hotel that are using cryptocurrency as a payment method

Name and URL	Country	Year of adoption	Cryptocurrency	Software to pay
(1) Airbaltic https://airbaltic.com/	Latvia	2014	Bitcoin	BitPay
(2) CheapAir.com https://cheapair.com/	United States	2013	Bitcoin	Coinbase
(3) Virgin Galactic https://virgingalactic.com/	United States	2014	Bitcoin, Ethereum, Litecoin	Coinbase
(4) Peach Aviation https://flypeach.com/	Japan	2017	Bitcoin	N/A
(5) Surf Air https://surfair.com/	United States	2017	Bitcoin, Ethereum	Coinbase

Table 2.4.2 Airline that are using cryptocurrency as a payment method

Scholarly research also informs us of the potential for blockchain, and cryptocurrency use in service sectors. Önder and Treiblmaier [6] argue that blockchain has the potential to disintermediate established players with improved transparency and transaction integrity. Thees et al. [7] and Erceg et al. [8] also explore the potential for reconfiguration by blockchain in the value chain of tourism with the argument that blockchain-backed payments will be more efficient and better regulated.

CoinGate, for example, has been tested in comparison with other payment processors and included in usability and efficacy comparative evaluation. Despite having a mid-user score of 2.7 out of 5, the fact that it forms part of the ecosystem alongside other interfaces such as Paytomat, Coinbase Commerce, and CoinsPaid determines its share of influence towards cryptocurrency adoption through payments [5]. Such interfaces' usability and simplicity are paramount in gauging the extent to which cryptocurrencies can be integrated in customer-facing industries.

Whereas tourism is a natural use case for cryptocurrency due to its borderless nature, e-commerce too is a suitable field for cryptocurrency adoption. Having similar requirements for fast and secure worldwide transactions, e-commerce businesses can utilize CoinGate by introducing alternative ways to pay in order to keep pace with changing consumer demands.

With ever more users demanding alternative ways to pay and with cryptocurrency awareness rising, the utilization of CoinGate in e-commerce could mirror its expanding success within tourism.

However, there are challenges. Legal uncertainty, regulatory hurdles, and low merchant and consumer awareness are barriers to mass adoption [9]. To overcome this, researchers propose forward-looking efforts like education initiatives, the development of stable coins, and the establishment of clear regulatory guidelines to promote trust and encourage adoption.

Briefly, literature and market trends show a growing interest and readiness for cryptocurrency usage in various industries. CoinGate, as a cryptocurrency payment processor, is a good solution for the tourism and e-commerce industries, offering businesses a way to meet customer demands for new, secure, and efficient payment systems.

2.4.1 Strengths

1. **Contemporary Relevance:** The research provides inquiry on cryptocurrency usage in tourism, an emerging industry directly relevant to digital commerce [5].
2. **Use of Case Studies:** Current examples, such as hotels and airlines accepting cryptocurrency, situate the academic discourse in practical terms [5].
3. **Consumer Behavior:** Surveys indicating 80% consumer interest in digital currencies provide credence to a recommendation to integrate services like CoinGate for modern consumption livelihoods [5].
4. **Literature Review:** The inclusion of multiple studies [6]–[8] bolsters the discussion with literature on blockchain regarding efficiency, transparency, and redesigning the value chain.

2.4.2 Weaknesses

1. **Absence of Quantitative Data:** The research relies heavily on qualitative observations rather than detailed statistical analysis that may validate the trends observed [5].
2. **Limited Quantity of Material Related to CoinGate:** The technological capacities, integration, and usability of CoinGate do not feature in sufficient detail [5].
3. **Generalized Discussion of Regulatory No Technical Barriers:** Regulatory and technical barriers were discussed generically without region-specific insight useful to policymakers or businesses [9].
4. **Geographical Limitations:** The research focused almost exclusively on Europe and North America, leaving out cryptocurrency use and adoption in the context of emerging economies [5].

2.4.3 Gap

1. **Lack of Real-Time Data Integration:** Studies have yet to examine real-time monitoring or visualization of cryptocurrency payment transactions.
2. **Limited Analysis of the User Experience:** A detailed evaluation of user experience in terms of usability, interface design, or interaction experience of CoinGate are not discussed.
3. **No Decision-Support Integration:** Existing studies do little to analyze integrating cryptocurrency payment choices into decision-support systems, which would support businesses in maximizing payments.

2.5 Exploring the Use of Crypto-Assets for Payments

The last few years have witnessed the fintech space experience a revolution in digital payment systems, dominated by the growth in convergence of both legacy and distributed financial systems. As both companies and customers continue to switch towards digital interfaces for making payments, there has been a parallel need for intelligent systems that would be able to recommend best-fit payment technologies. This project serves that purpose using a display and advisory system from the Odoo platform, featuring disparate payment vehicles such as Stripe, CoinGate, and other wireless options. The literature suggests the use of crypto-asset-based payments and the underlying technology infrastructures.

Koutrouli and Manousopoulos [10] provide an elaborate overview of available usage of crypto-assets in payment, focusing on unbacked crypto-assets and stablecoins. Their work demonstrates a number of use cases, including micropayments, streaming payments, instant settlements, and cross-border payments. These use cases are critical to determine how payment habits can vary based on transaction types and user needs. For instance, micropayments and streaming payments, which are often handled by DeFi protocols such as Sablier and LlamaPay, are very relevant in backend automation systems with continuous or event-driven streams of transactions.

The incorporation of crypto assets into business platforms also highlights their growing presence in modern payment systems. Gateways like Coinbase Commerce, CoinGate, and BitPay enable e-commerce websites to accept and process crypto payments via real-time confirmations and automatic fiat conversion. These features align with the objectives of the

proposed system, which utilizes Odoo to display and suggest suitable payment technologies by context and user preference. Through API integration from such gateways, the system is able to real-time suggest best modes of payment while enabling backend automation and transaction processing.

While crypto assets have the potential, the article notices that their usage in payments is still in infancy. The authors attribute this to reasons such as price volatility, legal uncertainty, and regulatory dis-integration, which are discouraging mainstream use [10]. For example, whereas PayPal, Revolut, and Visa have brought on board services to enable crypto transactions, these are often followed by complex compliance processes and limited geographical reach. This underlines the necessity of an advisory mechanism that can direct users towards more stable and supported payment alternatives based on local legislation and platform support.

In addition, the article refers to the European Union's adoption of the Markets in Crypto-Assets Regulation (MiCAR) for the purpose of standardizing crypto-asset operations and promoting stablecoin usage. Regulatory clarity of this nature would be particularly pertinent to use for the purpose of guiding customers within the EU and could be added to the system logic in order to enhance the dependability of payment recommendations.

From a consumer behavior perspective, the article reports that most consumers currently utilize crypto assets as investment vehicles rather than payment tools, with specialist adoption solely for payments [10]. Increased expansion of infrastructure such as cPOS terminals and crypto ATMs, though, indicates a steady but progressive trend towards broader usage. The system proposed in this work aims to bridge this adoption gap by providing payment options based on availability, price, method of transaction, and user familiarity—hence enabling informed decisions.

In summary, literature attests to the necessity of developing an advisory system to support both old and new payment technologies. Gateways, legislation, and behavior are incorporated into a robust platform for designing the advisory engine and transaction modules of Odoo. As such, the work of Koutrouli and Manousopoulos is critical in guiding the theoretical and practical underpinnings of this Final Year Project.

2.5.1 Strengths

1. **Broad Real-World Applicability:** Includes several examples of crypto applications such as micropayments, machine-to-machine payments, real-time streaming payments, and cross-border payments. Understands the end goal of the FYP, i.e., backend automation and payment routing logic.
2. **Integration into Payment Ecosystem:** Discusses how platforms like CoinGate, BitPay, and Coinbase Commerce have been integrated into the existing infrastructure of commercial payment channels, demonstrating feasibility for Odoo ERP.
3. **Regulatory Awareness:** Discusses European Union regulation (i.e. MiCAR) to address a number of legal issues associated with crypto-payment system usage.
4. **Data Review:** Includes statistical data and surveys of user adoption, adding a level of empirical evidence to support the writer's observations about general trends in the market and user behavior.

2.5.2 Weaknesses

1. **Lack of Technical Depth:** Lacks technical information around backend integration, as well as APIs or protocols for payment gateway implementation.
2. Reliance on Voluntary Data Sources: Where it discusses sources of data acquisition, the academic source is often limited to publically available or voluntary data acquisition which is not standardization and can bias observed adoption trends.
3. **Limited Global View:** Primarily focused on Europe to the neglect of challenges faced by other regions, i.e. Asia, the Americas, and Africa.
4. **No Behaviour / User Experience Analysis:** There are very few or no consideration for behavioural models or user interface design processes adopted in decision-support systems.

2.5.3 Gap

1. **Absence of Technical Guidelines for Implementation:** There is no specific roadmap for integrating crypto-payment platforms with ERP systems such as Odoo to leverage advisory, real-time proxies for payment decisions.
2. **Standardized Global Data as a Need:** There is no globally harmonized data source which decreases the reliability of trend analysis in relation to both adoption tendencies and regulatory considerations.

3. **Behavioral Insights for Advisory Engine:** No exploration has been undertaken for different consumer behavior or UX modeling to assist in the development of an intelligent advisory dashboard which is user-friendly.
4. **Regional Regulatory Variations:** Does not identify issues associated with different legal regimes outside Europe which is essential for the global scalability of advisory payments solutions.

Chapter 3

System Methodology

3.1 System Design Diagram/Equation

This project's system architecture is constructed with a layered modular development approach to facilitate scalability, maintenance, and data flow between components of the system. Each layer has a separate role while maintaining a separation of responsibilities, which simplifies the integration of the Odoo ERP backend, custom-built modules, payment gateways, and AI tools.

The architecture supports two primary stakeholders:

1. Customers: Customers engaging with the e-commerce frontend to browse products, submit orders, and make payments.

2. Business administrators: Business administrators engaging with the web interface dashboard to see transaction monitoring, see the payment trend, and receive AI recommendations.

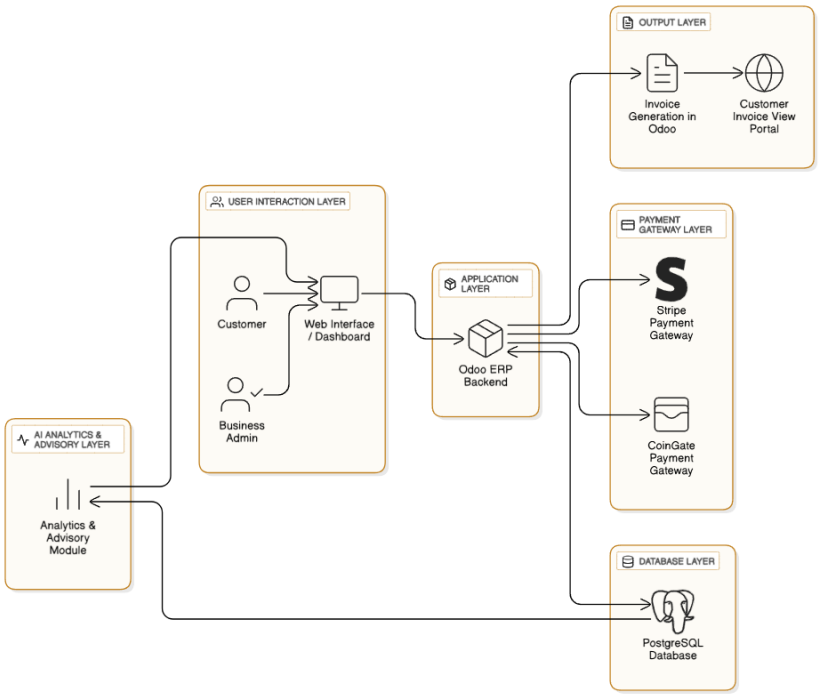


Figure 3.1 System Architecture diagram

3.2 Use Case diagram and description for E-commerce Checkout with Multi-Payment Methods

3.2.1 Use case diagram

Figure 3.2.1 Use Case diagram of E-commerce Checkout with Multi-Payment Methods

3.2.2 Use case description

Use Case ID	UC001	Version	1.0
Use Case	E-commerce Checkout with Multi-Payment Methods		
Purpose	Enable users to purchase items online through the Odoo system by selecting an integrated payment method, verifying funds via third-party gateways, and displaying a successful or failed payment result.		
Actor	User		
Trigger	User clicks “Checkout” after adding items to cart.		
Precondition	User has a valid payment method linked.		
Scenario Name	**Step**	**Action**	
Main Flow	1	User browses items.	
	2	User adds items to cart.	
	3	User clicks “Checkout.”	
	4	User selects a payment method (digital wallet, visa card, CoinGate).	
	5	User confirms payment.	
	6	System sends payment requests to the external payment provider.	
	7	Payment provider checks balance and processes payment.	
	8	Payment provider sends responses to system (Success or Failure).	
	9	Payment success and system mark order as Paid and generate receipt.	
Alternate Flow – Insufficient balance	9.1	Payment fails and system shows in payment failed.	
	9.2	Users select other payment methods with sufficient balance.	
Rules	-		
Author	Tsok Jia Xuan		

Table 3.2.2 Use Case Description of E-commerce Checkout with Multi-Payment Methods

3.3 Use Case diagram and description for AI Dashboard

3.3.1 Use Case diagram

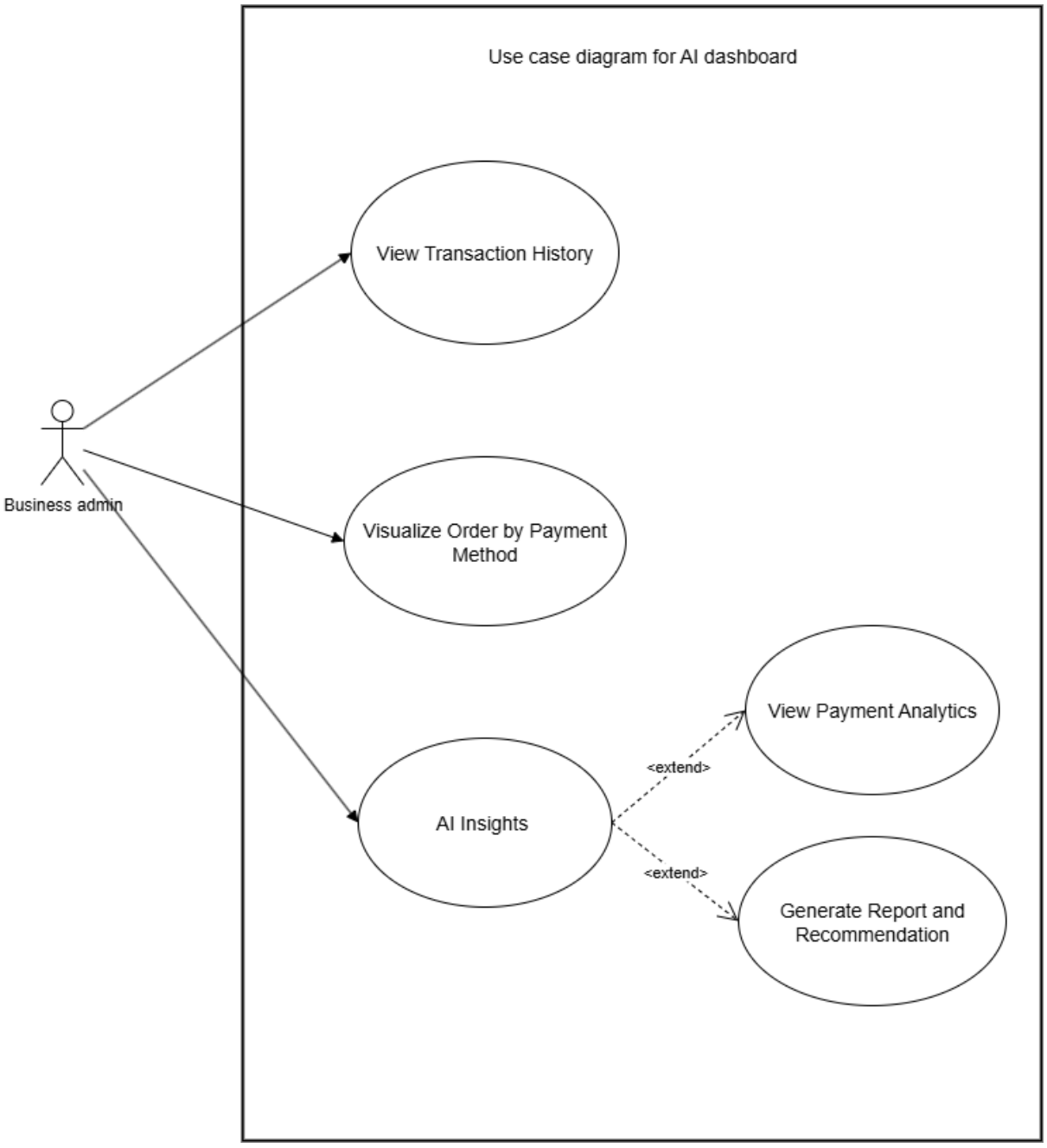


Figure 3.3.1 Use Case diagram of AI Dashboard

3.3.2 Use Case description

Use Case ID	UC002	Version	1.0
Use Case	AI Dashboard		
Purpose	Enable the Business Admin to analyze payment data, visualize orders, and optionally generate AI-powered insights and recommendations for better decision-making.		
Actor	Business admin		
Trigger	Business admin accesses the AI Dashboard to review payment-related information.		
Precondition	Dashboard is accessible with all connected modules (orders, payments, AI engine).		
Scenario Name	**Step**	**Action**	
Main Flow	1	Business Admin accesses the AI Dashboard.	
	2	The system displays available options of View Payment Transaction History, Visualize Orders by Payment Method and AI Insights.	
	3	Business Admin selects View Payment Transaction History.	
	4	The system retrieves and displays all past payment transactions from the database.	
	5	Business Admin selects Visualize Orders by Payment Method.	
	6	The system generates graphical representations of orders categorized by payment methods.	
	7	Business Admin selects AI Insights.	
	8	The system loads AI-powered analytics tools for deeper insights.	
Extension Flow – View Payment Analytics	7.1.1	Business Admin may select View Payment Analytics under AI Insights.	
	7.1.2	The system generates AI-powered analytics, such as trends and preferred payment methods.	
Extension Flow – Generate Report & Recommendation	7.2.1	Business Admin may choose to Generate Report & Recommendation after viewing analytics.	
	7.2.2	The system produces an AI-generated report with recommendations for payment methods merchants should prioritize.	
Rules	AI recommendations are suggestions; final decisions rest with the Business Admin.		
Author	Tsok Jia Xuan		

Table 3.3.2 Use Case Description of AI Dashboard

3.4 Activity diagram

3.4.1 Activity diagram of E-commerce Checkout with Multi-Payment Methods

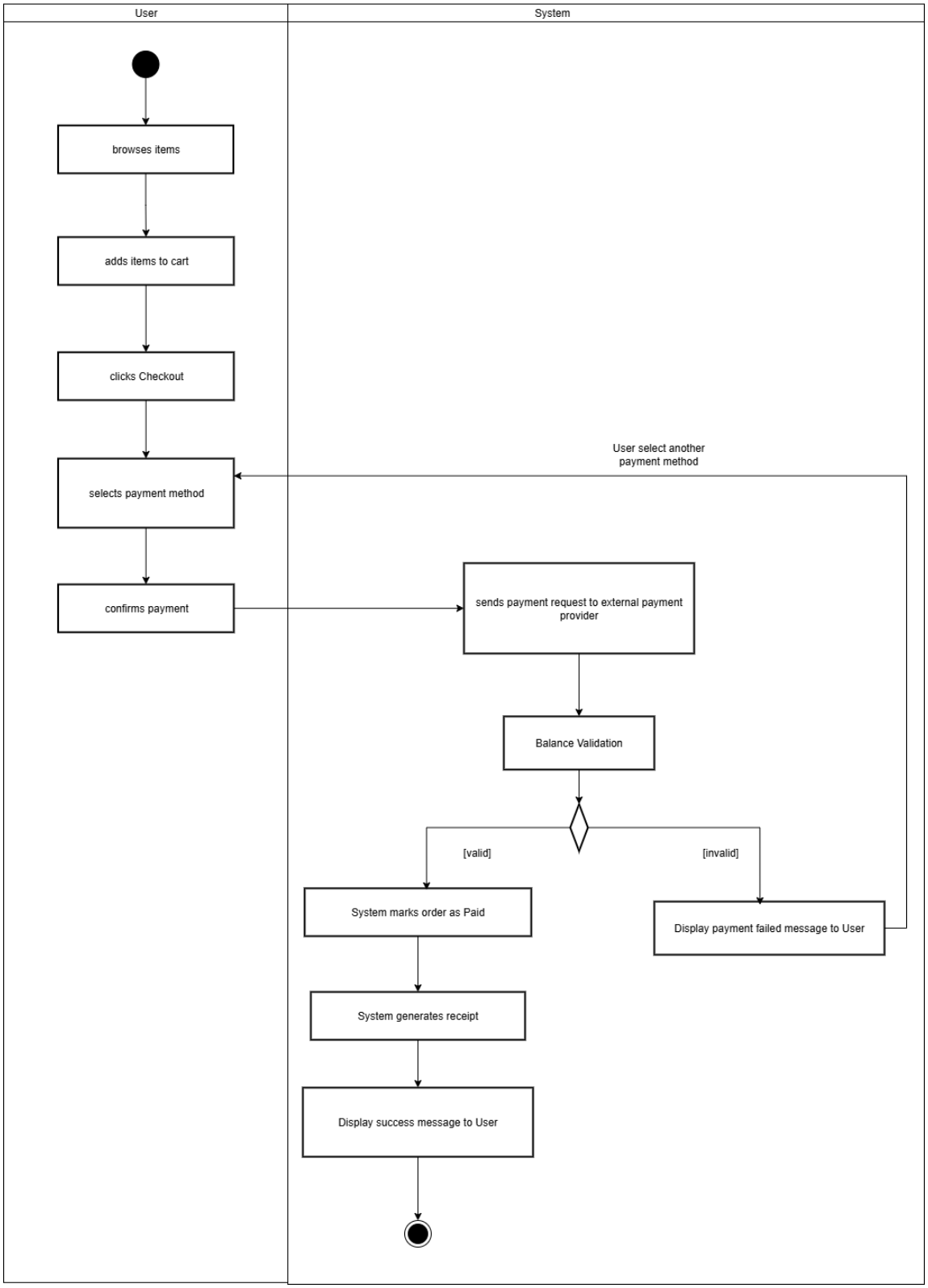


Figure 3.4.1 Activity diagram of E-commerce Checkout with Multi-Payment Methods

3.4.2 Activity diagram of AI Dashboard

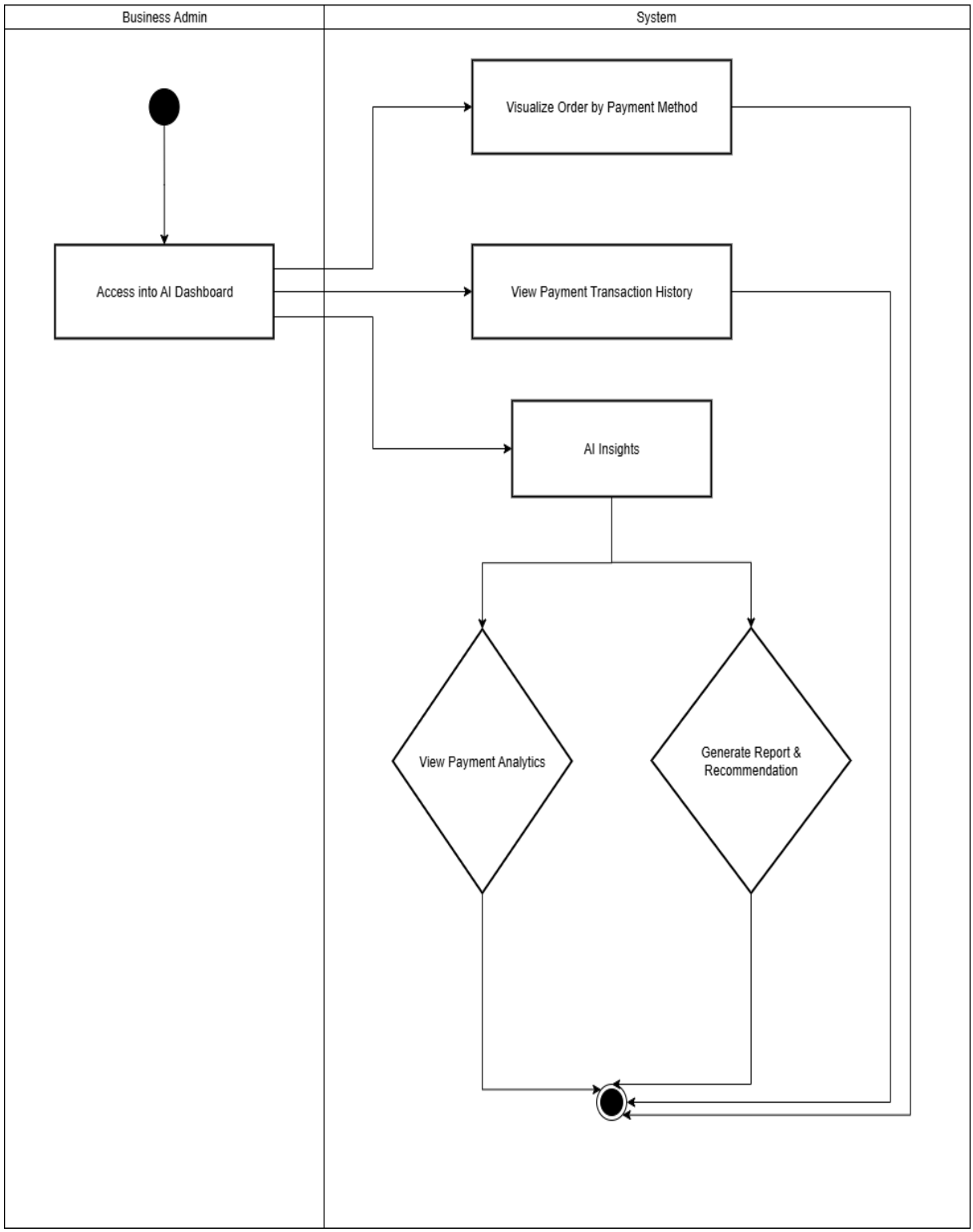


Figure 3.4.2 Activity diagram of AI Dashboard

Chapter 4

System Design

4.1 System Design

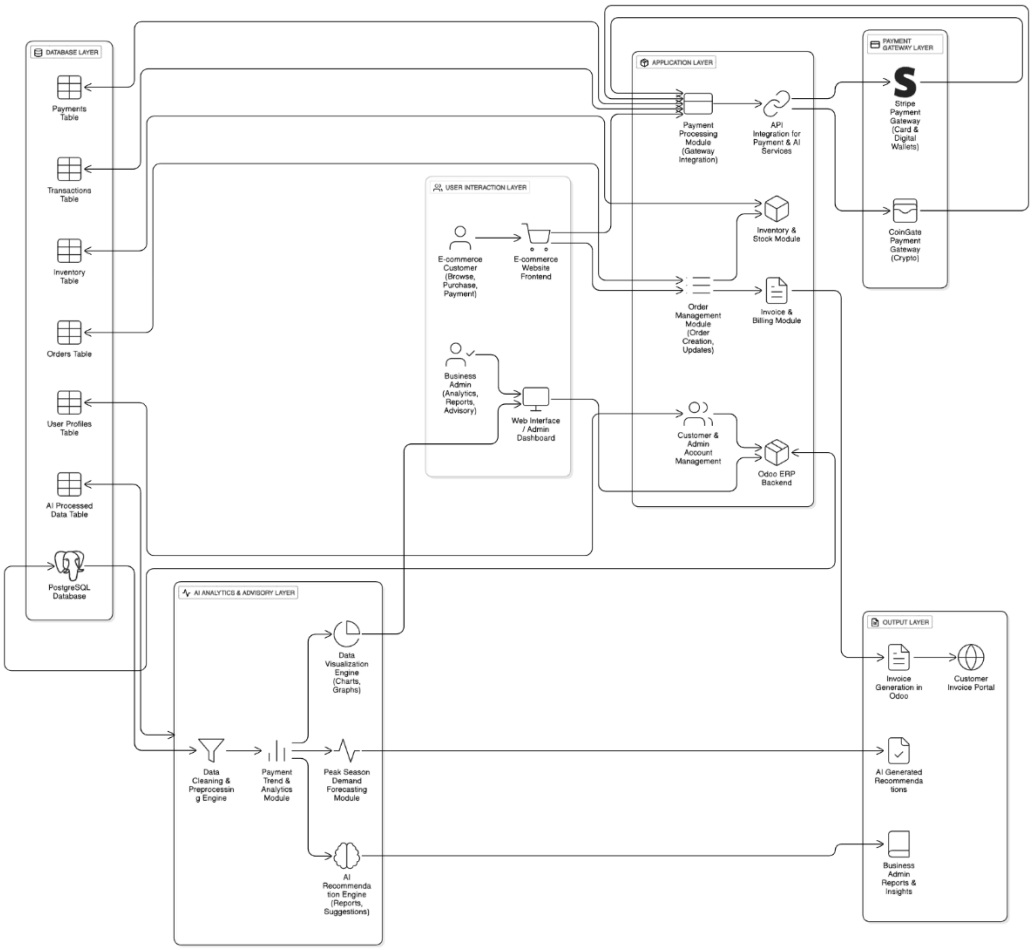


Figure 4 System Design Diagram

The system block diagram in this project is formulated with a layered modular design to promote clarity, visual flow of data, and proper separation of responsibility between different components. This design provides a pathway for seamless connection between the custom modules developed for this project and Odoo ERP internal functionality, creating an integrated system to streamline payments, analytics, and reporting.

At the highest level is the User Interaction Layer, which provides the main interface for business administrator and customer interactions. The customers will use the e-commerce frontend to browse product offerings, as well as place orders and make payments, while

business administrators will use the Web Interface Dashboard to monitor transactions as well as visually see customers' payment patterns, and read AI-tools generated business insights. This layer contains usability and accessibility to create a frictionless user experience for both stakeholders.

Below is the Application Layer, which is built upon Odoo ERP Backend. Most of the functional modules in this layer, Order Management, Inventory Management, Invoice Management, and User Management, are native modules provided by Odoo's internal ERP system, these modules manage core business processes, including order management, products, invoicing and account management. However, two of the modules in this application layer have been custom developed as follows:

Payment Processing Module – A custom module developed for this project; it processes real-time payment transactions and communicates the ERP backend with the external payment gateways.

API Integration Module – A custom module, it facilitates communication between Odoo, the payment gateways (Stripe and CoinGate), and the AI analytics system for automatic and secure data exchange.

The Payment Gateway Layer comprises all custom integration work. The Stripe Payment Gateway is responsible for processing digital wallets and credit card mix, while the CoinGate Payment Gateway facilitates cryptocurrency payments. Both gateways have integrated custom APIs developed for this project, which validates, confirms, and syncs all transactions with Odoo’s platform in real-time.

The Database Layer uses PostgreSQL, a mature relational database that stores all business and transactional data. There are separate tables for all orders, payments, transaction history, inventory records, user profiles, and AI analytics outputs. Odoo’s built-in ORM framework manages the database for most standard tables, while additional tables for AI analytics outputs were developed from scratch for transaction forecasting and advisory functionality.

The AI Analytics and Advisory layer was uniquely customized for this project. Several sub-modules exist in this layer, including a Data Preprocessing Engine that cleans sanitary transactional data, an Analytics Advisory Module that detects payment trends, an Analytics Visualization Engine that creates interactive dashboards, a Recommendation Engine that suggests less expensive payment options, and a Forecasting Engine that predicts payment methods for peak seasons. This layer turns raw transactional data into actionable insights for businesses.

In conclusion, the Output and Reporting Layer presents outcomes to end users. Consumers automatically receive invoices from Odoo's proprietary Invoice Management Module; as businesses, administrators receive recommendations on an interactive dashboard, and customized analytics reports from AI data processing. The reports allow businesses to prepare for periods of peak demand and select payment technology aligned with consumer preferences.

The entire process begins with a customer placing an order via the e-commerce frontend. The order then is processed in the Odoo ERP backend, passed to a custom developed Payment Processing Module for either Stripe or CoinGate integration, and stored in the PostgreSQL database. The data then is fed through the AI Analytics Layer for patterns derivation and predictions, which are showcased on the AI Dashboard for administrative decisions.

4.2 System Components Specifications

The layered modular architecture of the system is structured for clarity, maintainability, and integration. Each component of the system has a purpose, and together they create a seamless experience for the customer and the business administrator. This section discusses all the components of the system with respect to the functional specifications, inputs, outputs, and dependencies.

4.2.1 User Interaction Layer

The User Interaction Layer is the interface through which the stakeholders interact with the system. It has two primary interfaces:

1. **E-commerce Frontend (Customer Interface):** This module allows customers to browse products, add products to the cart, place orders, and check out with payments. It offers usability and a smooth purchasing experience.
   Input: Customer selections, payment information, order.
   Output: Order request to the backend, payment request, order confirmation.
   Dependencies: Odoo ERP backend, Payment Processing module.

2. **Web Interface Dashboard (Admin Interface):** For business administrators, the interface provides payment trend analysis, transactional data, AI-based recommendations, and reporting.
   Input: Backend ERP data, AI analytics results.
   Output: Interactive dashboards, business reports, alerts.
   Dependencies: Odoo ERP backend, AI Analytics & Advisory Layer.

4.2.2 Application Layer (Odoo ERP Backend + Modules)

The Application Layer is the functional core of the entire system, which utilises both native Odoo ERP modules and custom modules.

1. **Order Management Module:** Manages order creation, updates, and tracking.

Input: Customer order details from frontend.

Output: Order confirmation, inventory updates, invoice notification.

Dependencies: Inventory Module, Invoice management module, PostgreSQL database.

2. **Inventory & Stock Module:** Tracks product availability, stock levels, and updates inventory when orders are completed.

Input: Orders from Order Management Module.

Output: Stock updates and low-stock alerts.

Dependencies: PostgreSQL database.

3. **Invoice Management Module:** Generates invoices for completed transactions and maintains billing details.

Input: Confirmed orders, payment status.

Output: Customer invoices, records for ERP billing records.

Dependencies: Order Management Module, Payment Processing Module, PostgreSQL database.

4. **Payment Processing Module (Custom):** Processes customer payments using Stripe and CoinGate gateways. Verifies and updates transactions in real-time.

Input: Requests for payments from the frontend system.

Output: Payment success or failure, transaction updates.

Dependencies: Stripe API, CoinGate API, and PostgreSQL database.

5. **API Integration Module (Custom):** Ensures secure communication among Odoo ERP, payment gateways, and the AI Analytics system.

Input: Data for transactions in ERP, requests for AI analytics.

Outputs: API responses processed, updates on data synchronization across provided services.

Dependencies: Payment Processing Module, AI Analytics & Advisory Layer.

4.2.3 Payment Gateway Layer

This layer integrates third-party payment gateways to securely and efficiently process different payment methods.

1. **Stripe Payment Gateway:** Processes card payments and digital wallets for customers.

Input: Payment requests from Payment Processing Module.

Output: Validated payment, success/failure response.

Dependencies: Payment Processing Module, Odoo ERP backend.

2. **CoinGate Payment Gateway:** Processes cryptocurrency payments from customers.

Input: Crypto payment requests.

Output: Confirmation of transaction, payment records.

Dependencies: Payment Processing Module, Odoo ERP backend.

4.2.4 Database Layer

This system utilizes PostgreSQL for transactional operational and business analytical data storage. There is a specific purpose each table serves:

- **Orders Table:** Stores customer order information.
- **Payments Table:** Stores payment status and payment information.
- **Transactions Table:** Records all transactional events.
- **Inventory Table:** Maintains stock and product information.
- **User Profiles Table:** Stores admin and customer account data.
- **AI Analytics Data Table:** Stores processed data for utilization in recommendations, forecasting, and dashboards.

Each table within the database has its own purpose. All tables are managed through the ORM framework of Odoo, creating custom tables for AI Analytics to support forecasting and advisory use cases.

4.2.5 AI Analytics & Advisory Layer

This layer takes raw transactional data and turns it into meaningful insights to aid businesses in their decision-making.

- **Data Preprocessing Engine:** Cleanses and structures raw data for analysis.

- **Analytics Advisory Module:** Identifies trends in payment methods and customer behavior.
- **Recommendation Engine:** Provides AI-powered suggestions for optimal payment methods.
- **Forecasting Engine:** Predicts peak season payment trends to support planning and inventory management.

Summary table of System Component Specifications

Layer	Component	Function	Inputs	Outputs	Dependencies
User Interaction	E-commerce Frontend	Customer browsing, orders, payments	Customer selections & orders	Order/payment requests, confirmations	Odoo ERP, Payment Module
	Web Interface Dashboard	Admin analytics & reports	ERP & AI data	Dashboards, reports	Odoo ERP, AI Layer
Application	Order Management	Manage orders	Customer orders	Confirmations, inventory updates, invoices	Inventory, Invoice Modules, Database
	Inventory & Stock	Track & update stock	Orders	Stock updates, alerts	Database
	Invoice Management	Generate invoices	Confirmed orders, payments	Customer invoices, billing records	Order, Payment Modules, Database
	Payment Processing (Custom)	Process payments	Payment requests	Payment status, transactions	Stripe/CoinGate APIs, Database
	API Integration (Custom)	ERP, Payment & AI	ERP & AI data	API responses, data sync	Payment Module, AI Layer
Payment Gateway	Stripe	Card & wallet payments	Payment requests	Transaction confirmation	Payment Module, ERP

	CoinGate	Crypto payments	Payment requests	Transaction confirmation	Payment Module, ERP
Database	PostgreSQL	Store transactional & operational data	Orders, payments, inventory, AI data	Structured Database records	records ERP & custom modules
AI Analytics & Advisory	Analytics Advisory	Identify trends	Preprocessed data	Trend reports	Visualization, Recommendation Engines
	Recommendation	Suggest optimal payment methods	Analytics outputs	Recommendations	Advisory Module
	Forecasting	Predict peak trends	Analytics outputs	Forecast reports	Advisory Module

Table 4.2 Summary table of System Component Specifications

4.3 System Components Interaction Operations

The system has been implemented as layered modular design, with each module performing independent operations and then interacting with other modules in a specific order. The interaction operations support the efficient processing of customer orders, check payments, and generate actionable business insights for business administrators.

4.3.1 Component Interaction Sequence

1. **Customer Initiates Actions (User Interaction Layer):**
    - Customer choices, orders, and payments are captured by the E-commerce Frontend.
    - Order and payment requests are requested through secure API calls to Order Management Module and Payment Processing Module within the Application Layer.

2. **ERP Processing (Application Layer):**
    - Order Management Module checks the order, checks inventory levels through the Inventory & Stock Module, and triggers invoice generation through Invoice Management Module.
    - User Management Module checks users and manages account access.
    - Payment Processing Module sends payment requests to the concerned Payment Gateway (credit cards/electronic wallets via Stripe, cryptocurrencies via CoinGate) and waits for confirmation.

3. **Payment Gateway Interaction (Payment Gateway Layer):**
    - Payment gateways execute transactions and provide real-time feedback (success or failure) to the Payment Processing Module.
    - ERP backend writes to the Transactions Table and Payments Table of the PostgreSQL database as appropriate.

4. **Database Storage (Database Layer):**
    - Transactional, operational, and analytical data are stored in the PostgreSQL database.
    - Database integrity is maintained through relationships among Orders, Payments, Transactions, Inventory, User Profiles, and AI Analytics Data tables.

5. **AI Analytics & Advisory Processing (AI Layer):**
    - The Data Preprocessing Engine extracts transactional data from the database, cleanses and standardizes it.
    - The Analytics Advisory Module processes payment behaviors and creates advisories.
    - The Recommendation Engine suggests best payment means for different contexts.
    - The Forecasting Engine predicts peak season payment behaviors to aid inventory and resource planning.

6. **Admin and Customer Output (Output Layer):**
    - Business administrators see AI-created dashboards, reports, and suggestions via the Web Interface Dashboard.
    - Customers receive automatic invoices through the Customer Invoice Portal.
    - The system gives real-time updates and accurate data synchronization across all layers.

4.3.2 Design Considerations

1. **Modularity:** Each module is autonomous, simplifying maintenance and additional expansion (e.g., adding new payment gateways).
2. **Data Flow Integrity:** Event logging and data synchronization between ERP, gateways, and AI analytics avoid data inconsistency.
3. **Security:** Payment transactions and API calls are secured using HTTPS and authentication steps.
4. **Error Handling:** Failed payments or invalid orders trigger notifications for manual or automatic action.
5. **Scalability:** Supports many concurrent transactions and holiday bursts without performance degradation.

Chapter 5
System Implementation Experiment

5.1 Software Setup

The system's software environment has been fully set up and configured for the development and integration and testing of the project. The software environment consists of Odoo 18 Community Edition, PostgreSQL, Visual Studio Code, and payment gateways (Stripe and CoinGate sandbox). The implementation of the software was done personally during the course of the project

Odoo 18 Community Edition was installed as the core ERP application for the system. For the installation, I randomly downloaded the stable version of Odoo Community Edition from the official Odoo repository along with installing all system dependencies and Python libraries required for Odoo installation. Odoo was configured to make a connection with PostgreSQL as a backend database along with enabling a developer mode for the development process and customization of the specific modules sensible to the system functionally.

PostgreSQL was installed with Odoo during the package install to use as a reliable and secure database system. A database was created for the project in PostgreSQL to accommodate and control records for products, customers, and transaction data. The project database in PostgreSQL was connected to Odoo which allowed the application to perform "live" data operations for all modules to ensure data consistency and integrity.

Visual Studio Code (VS Code) was utilized as the main development environment. During the project, it was used very frequently for writing and modifying Python scripts for custom Odoo modules, configuring integrational files, debugging code, and managing source control. The IDE's features, such as syntax highlighting, Odoo development extensions, and built-in Git support, all contributed to the development process that was efficient and accurate during the implementation phase.

For payment functionality, Stripe was enabled directly in the Odoo 18 Community Edition. As Stripe is natively integrated with Odoo, only a small amount of configuration was needed for this purpose, where only the API credentials had to be entered in order for the card payments to be processed securely by Odoo. The CoinGate sandbox, on the other hand, was configured manually to simulate cryptocurrency payments. A sandbox account was registered, and API keys were created in Odoo for testing purposes, which enabled the testing of cryptocurrency transactions without having to interact with real money.

All software installs and associated configuration were successfully completed, which created a fully functional system environment. This provided a strong base to develop modules, payment function integration, and testing systems functionalities, to make sure that all components would work effectively together. The software installations were an essential milestone in the implementation phase, enabling the upcoming phases of development and testing to take place effectively.

5.2 Setting and Configuration

Following the software installation phase, the next task involved system configuration within Odoo 18 Community Edition to establish the required functional content for the project. Configuring the system involved activating modules, configuring payment providers and building custom modules to provide additional functionality.

The first step was activating the e-commerce module in Odoo. The e-commerce module provides the ability to create an e-commerce website and manage the website from Odoo. After activation the e-commerce website was created utilizing the website builder within Odoo which allows you to customize the look and feel, layout, and product display as per the requirements of the project. This setup ensured the system was able to simulate a fully functional online store environment for the purpose of testing payment transactions and user experience interactions.

Once the website was established, the Stripe payment provider was configured in the module settings. Any information required including API credentials were input into the settings along with the available payment option providers from Stripe. In this project the available payment options for customers on the website included Visa card, Alipay and GrabPay as

payment options. In order to prevent any real transactions from testing the site, the Stripe payment provider was set to test mode and for the purpose of simulating payment transactions for demonstration.

In addition to Stripe, a custom module for CoinGate was constructed to incorporate cryptocurrency payments. This module was developed manually through code writing directly in the Odoo folder using Visual Studio Code. All configurations were set up for CoinGate, including API credentials and supported cryptocurrency payment types. Following the completion of the CoinGate module, the developer mode in Odoo was enabled, and the app list was updated to be able to install this module of CoinGate and it was subsequently installed and activated successfully and was able to support simulated cryptocurrency transactions.

Furthermore, a custom module was developed for the AI dashboard, which was a great way to incorporate relevant analytical features to the system. This module was developed in Visual Studio Code and developed to analyze and display payment transaction data in relation to the different payment method types. It also included forecasting and the ability to identify customer preferred payment methods. Similar to the CoinGate module, the AI dashboard module was installed by first enabling developer mode and updating the app list in Odoo, after which it was successfully activated and operational.

Overall, the setting and configuration process in Odoo involved module activation, payment provider configuration, and the development and activation of custom modules. These steps ensured that the system was fully functional for testing payment transactions, displaying transactional data, and analyzing customer preferences, forming a solid foundation for subsequent implementation and evaluation phases of the project.

5.3 System Operation

The system operation describes how customers interact with the e-commerce website and how the backend processes transactions and visualizes data for business analysis.

When a customer loads the e-commerce website, they can browse the products available for sale, select the goods they want to purchase, then add those items to the shopping cart.

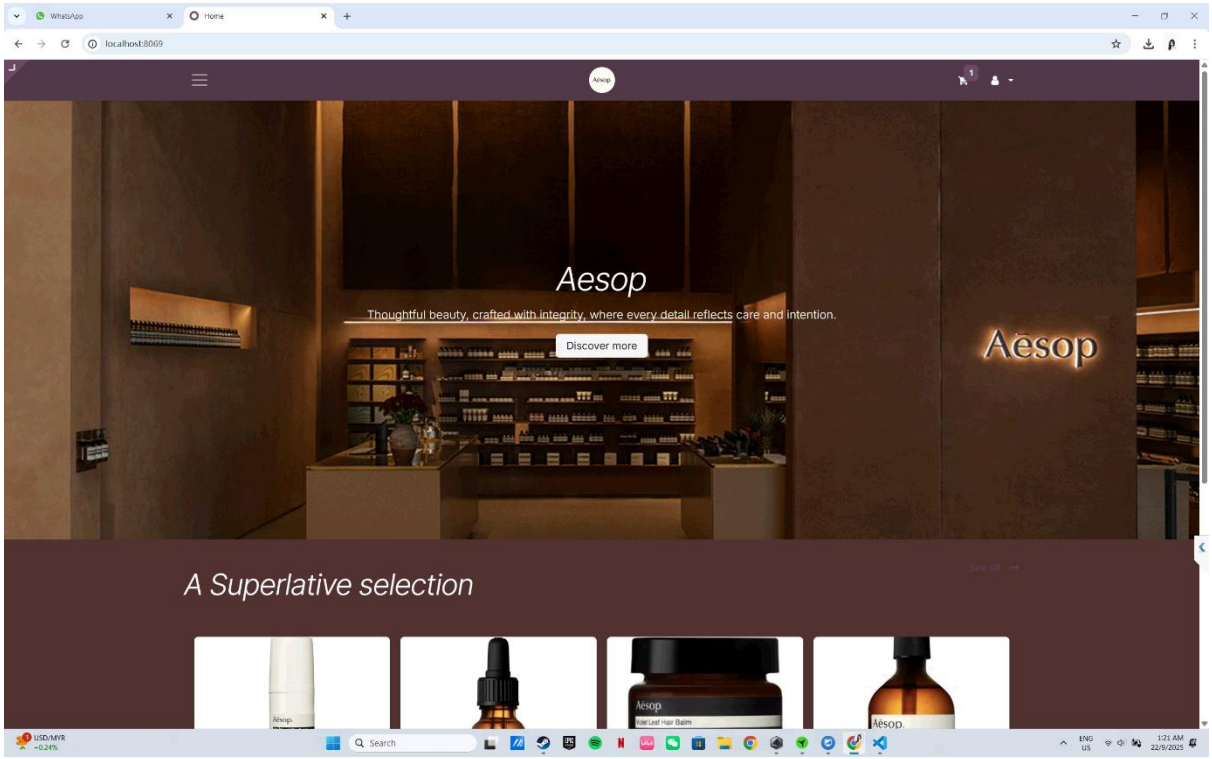


Figure 5.3 E-commerce website (front-end)

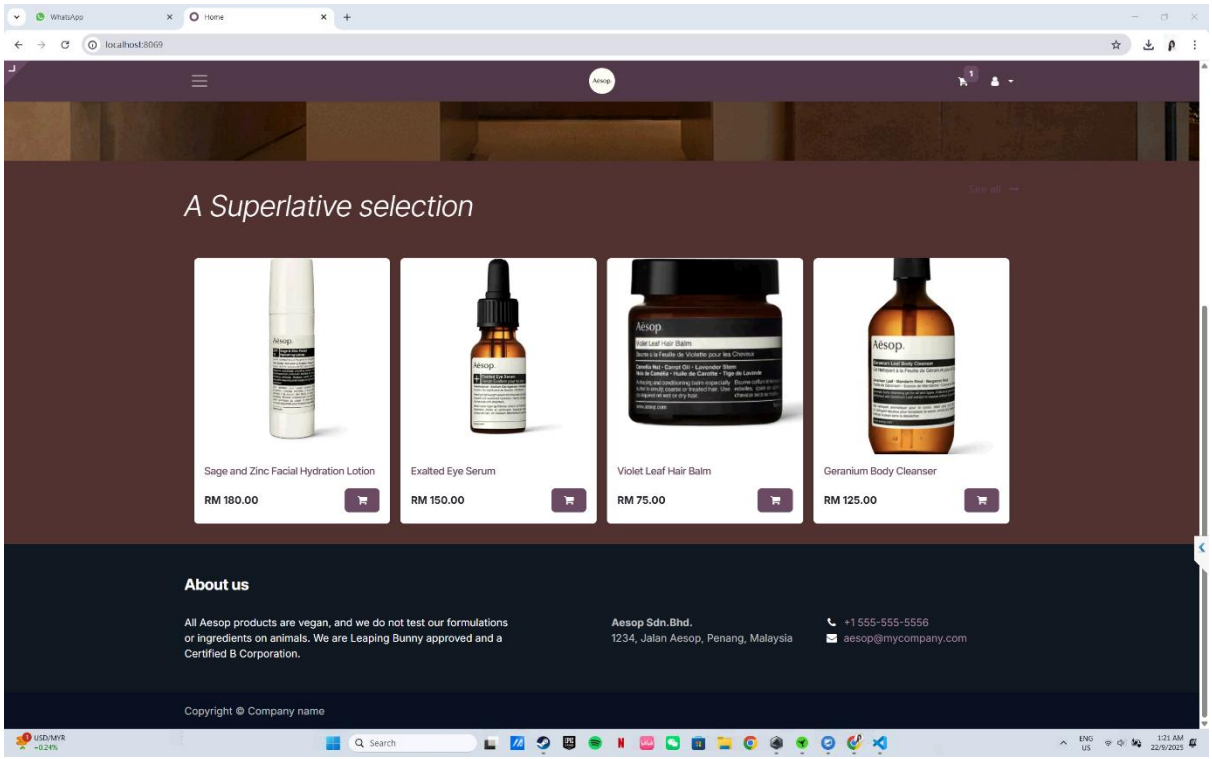


Figure 5.3.1 Product Page of E-commerce website (front-end)

Once the customer adds any item, they are interested in the shopping cart, the customer then selects View Cart to see their selections and finally chooses to Checkout.

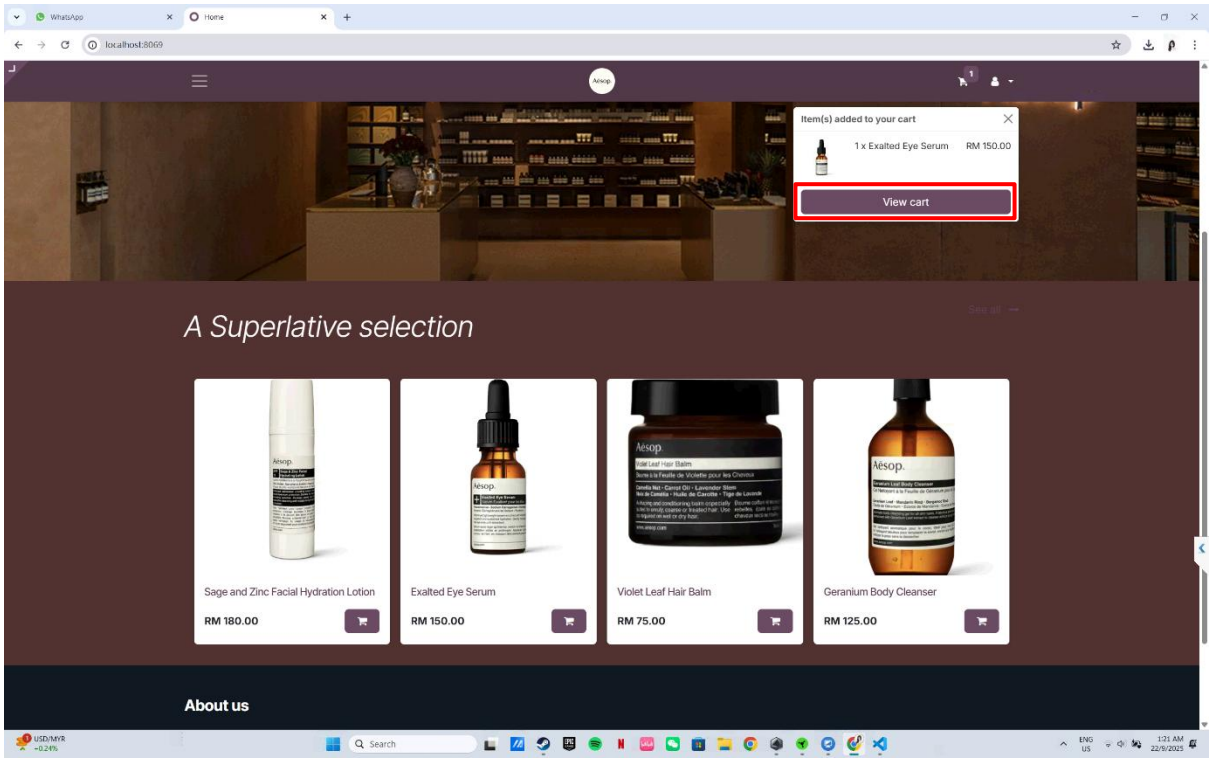


Figure 5.3.2 Item added into cart

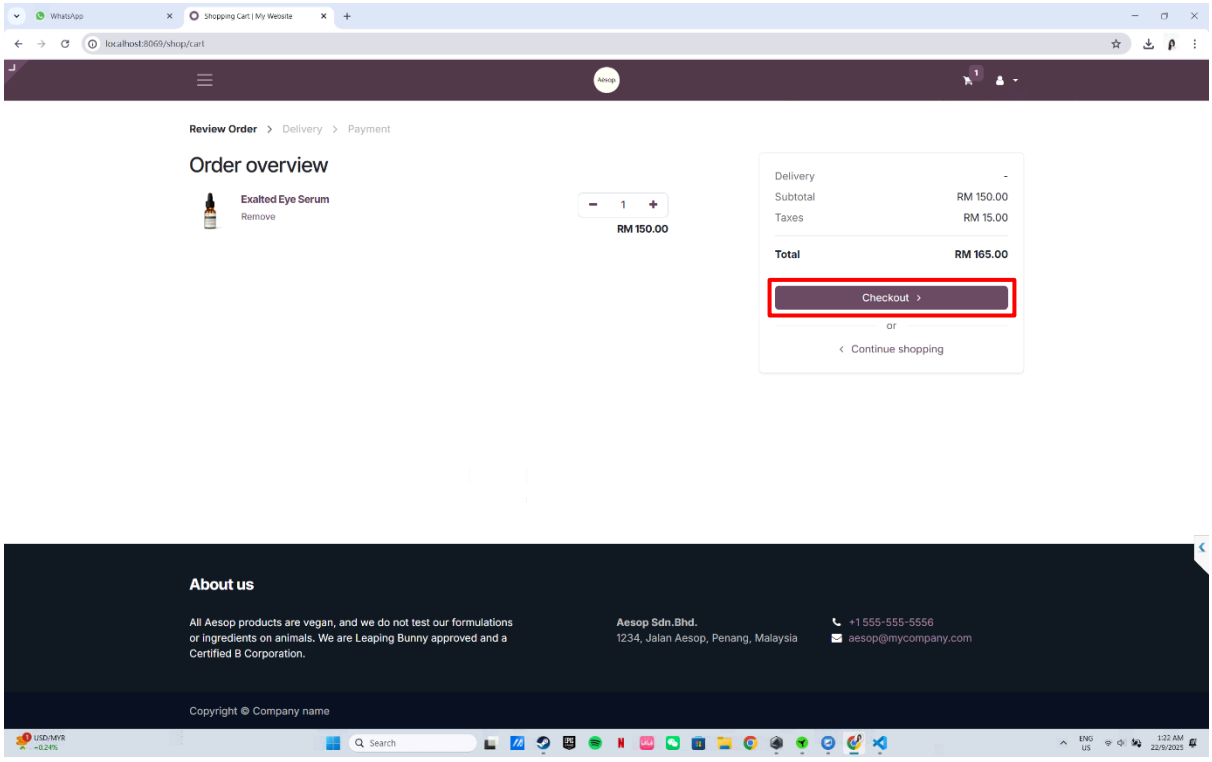


Figure 5.3.3 Check out page of cart

During the Checkout step, the customer is asked to complete the billing address. If the customer wants to enter a different shipping address besides the billing address they just entered, the customer either enters it, or the customer is expected prompt to enter it and leave the billing address as it is.

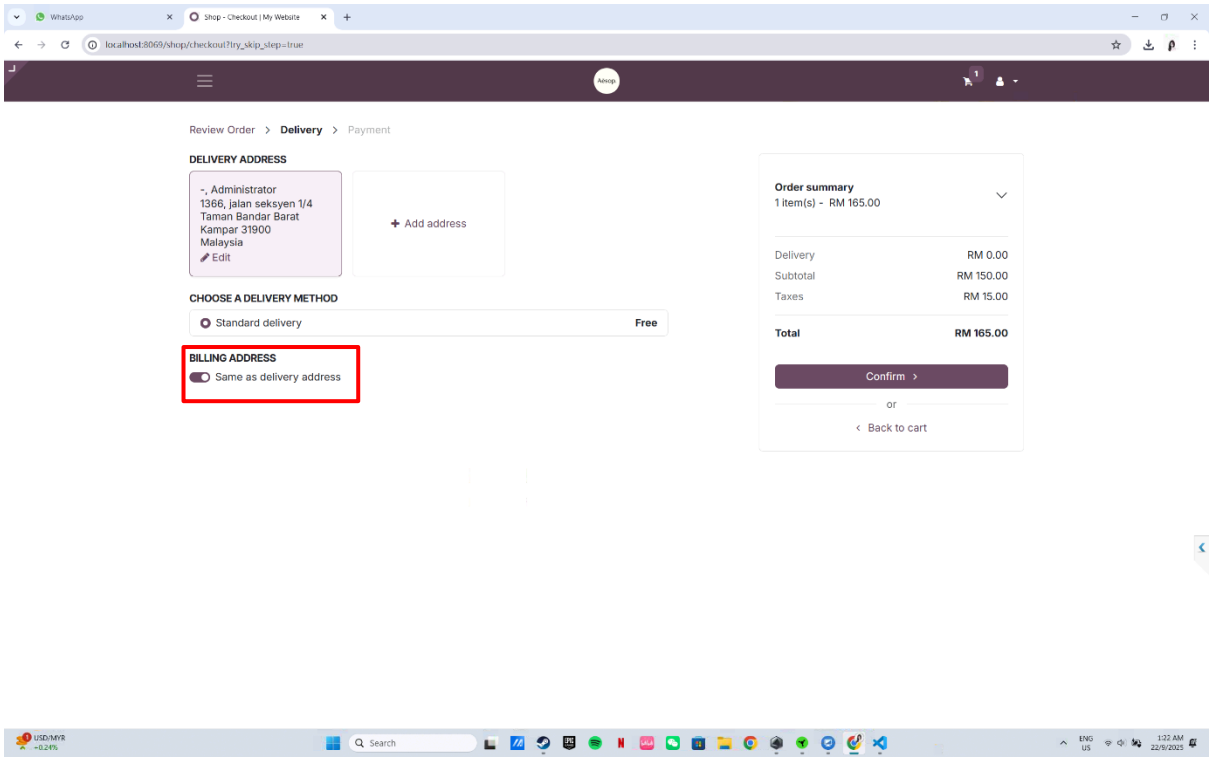


Figure 5.3.4 Shipping Address same as Billing Address

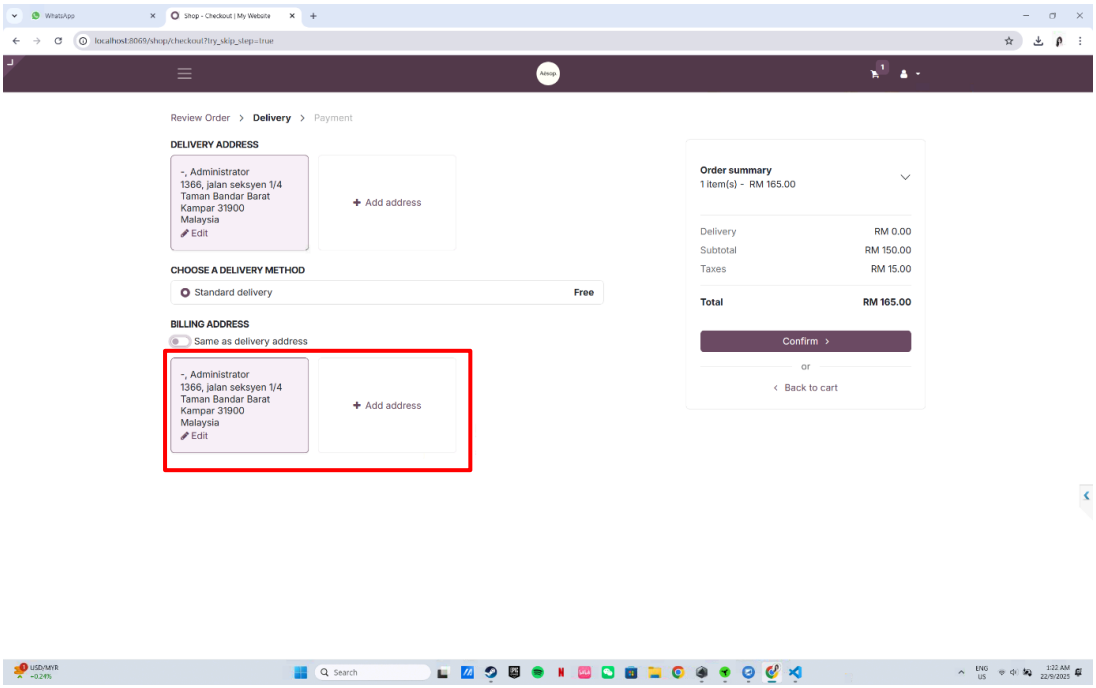


Figure 5.3.5 Add new shipping address

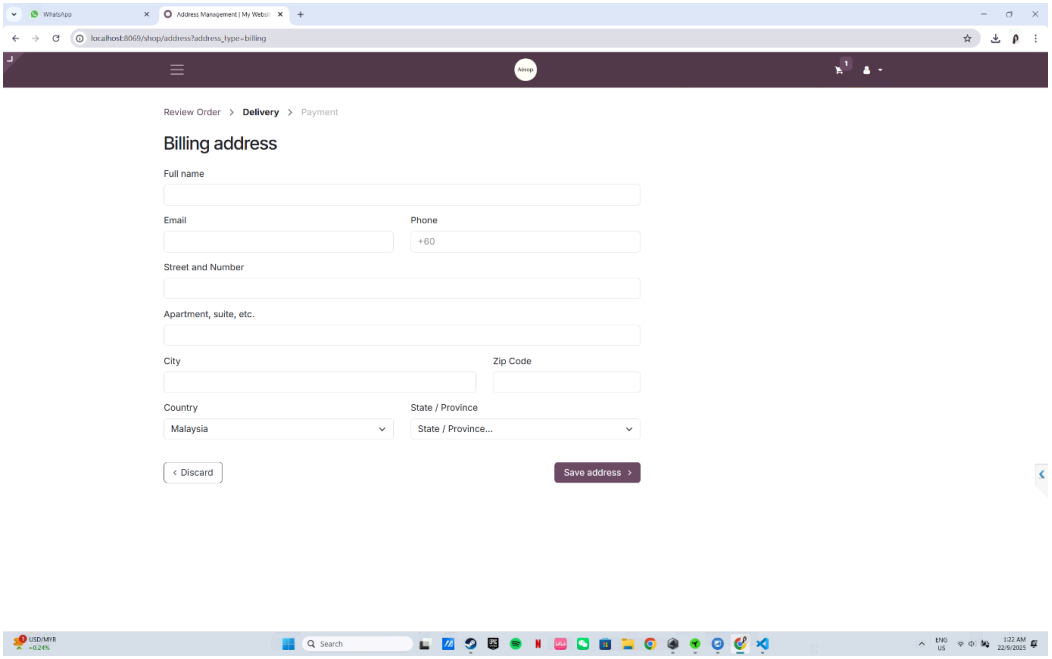


Figure 5.3.6 Fill in new shipping address information

Once the customer has completed those address inputs, the customer moves on to the Payment step for the transaction selection. Once done, the item prompts the customer to select their preferred payment option. Available options for the customer from the system: Visa card, Alipay, GrabPay, and CoinGate cryptocurrency payment. After selecting their payment option, the customer chooses Confirm Payment option to complete the transaction.

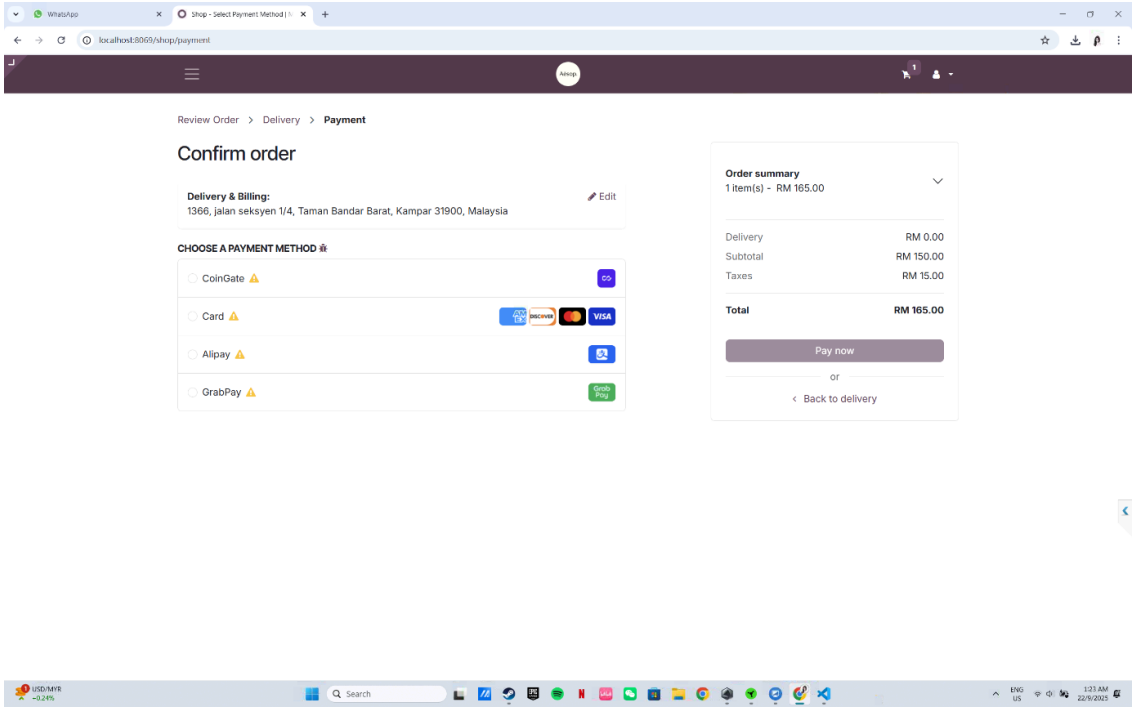


Figure 5.3.7 Multiple payment method

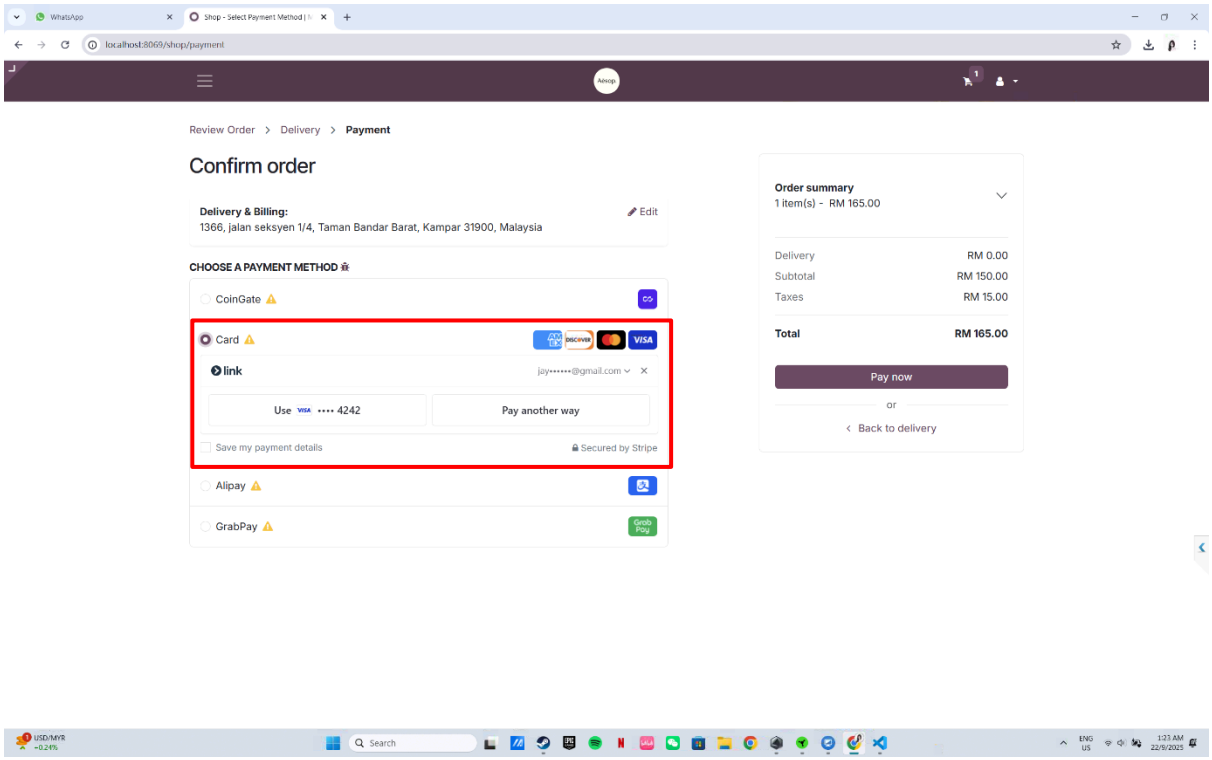


Figure 5.3.8 Visa Card Payment method via Stripe Payment Provider

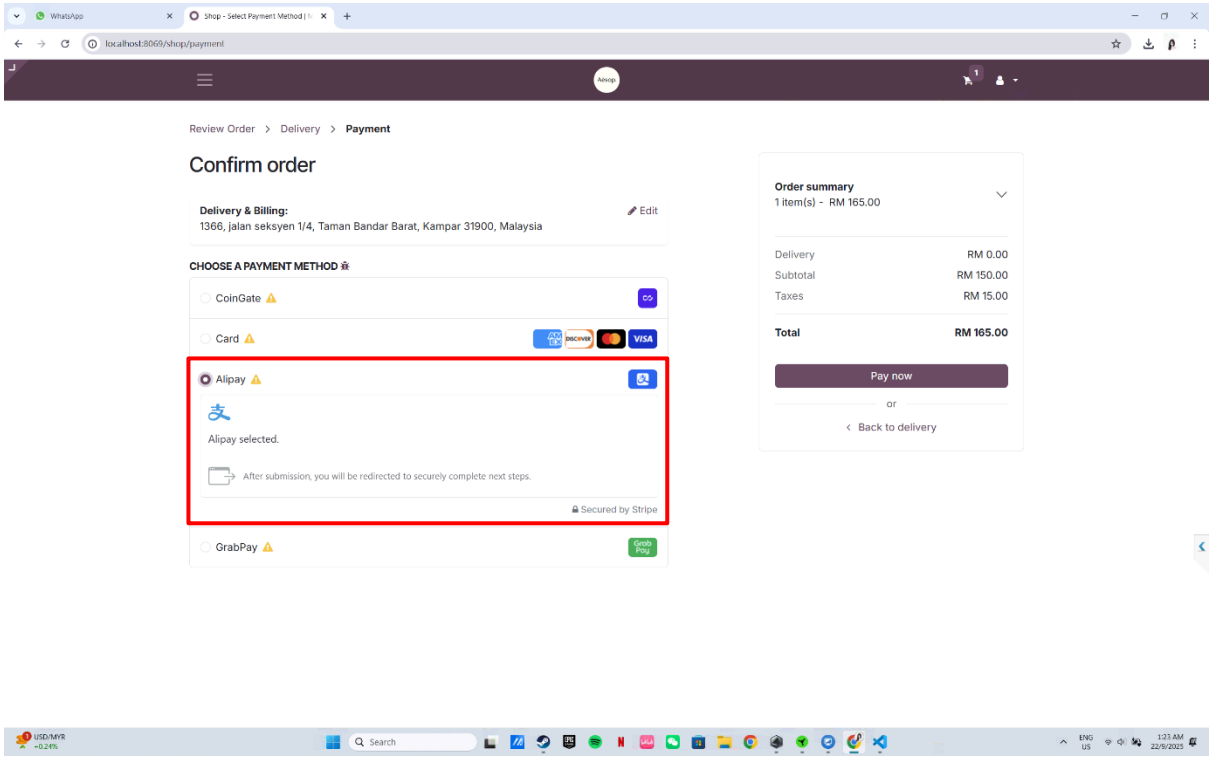


Figure 5.3.9 Alipay Payment method via Stripe Payment Provider

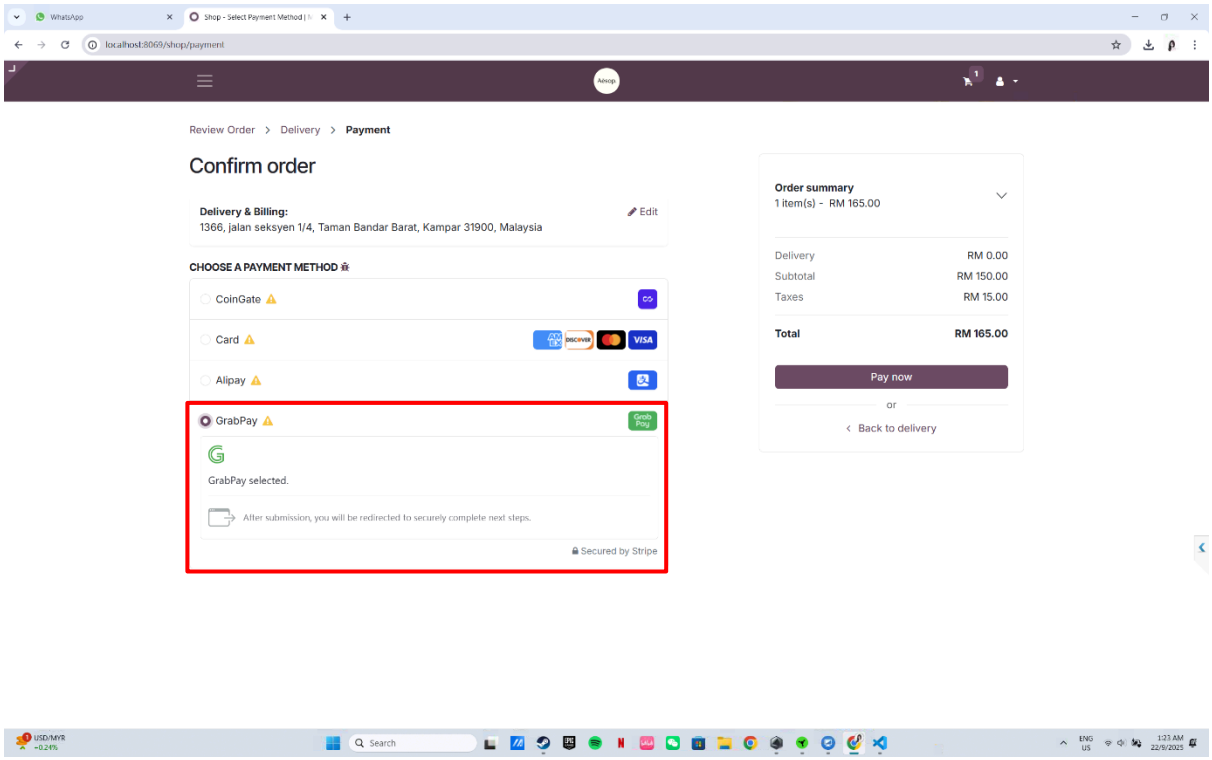


Figure 5.3.10 GrabPay Payment method via Stripe Payment Provider

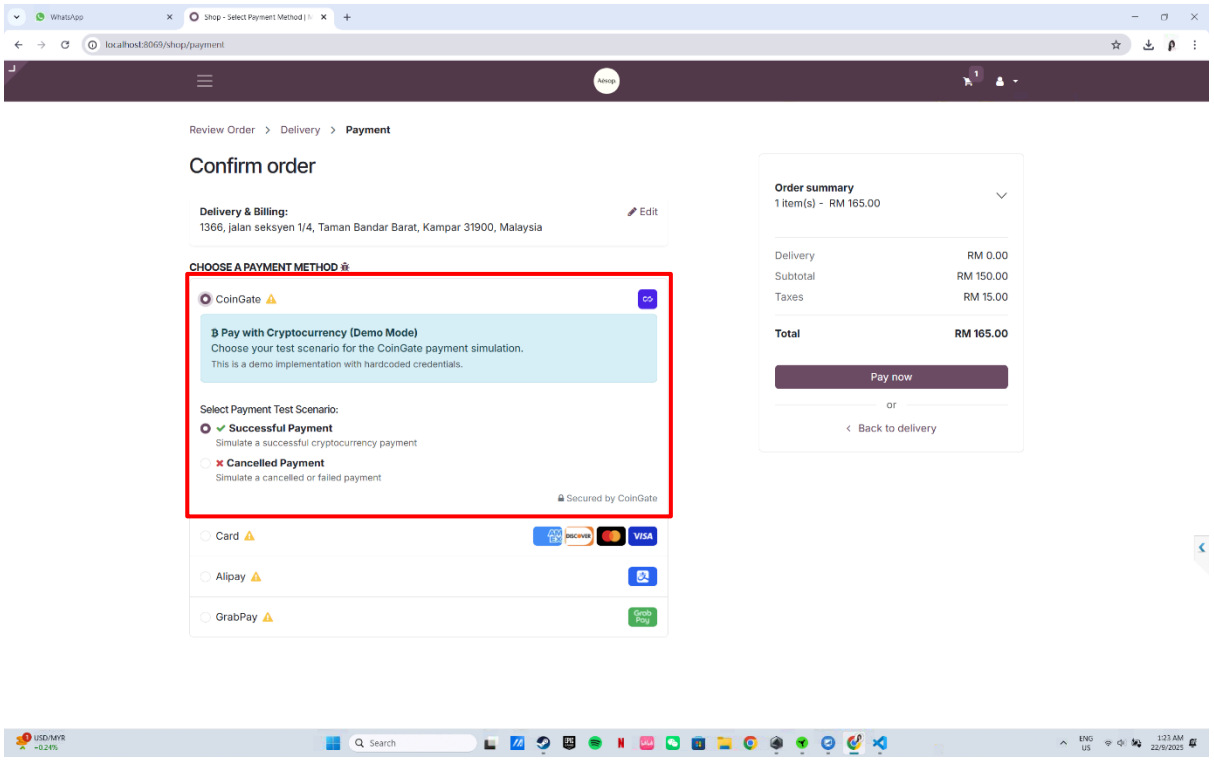


Figure 5.3.11 CoinGate Payment method

The system then contacts the appropriate payment provider to ascertain if the customer has enough funds to complete the transaction. If enough funds are available, the payment is carried out and confirmation is visible to the customer.

If there are not enough funds available, the payment fails and the customer is again prompted to go back to the payment selection, where an alternative payment choice is selected. This approach facilitates a transaction system that is secure and accurate, while at the same time giving feedback to the customer.

For business purposes, the system incorporates an AI-powered dashboard that visualizes transaction data. The dashboard will describe each payment according to the payment method and allow the business administrator to observe customer payment behavior. The administrator will be able to see trends and utilize AI analysts to predict consumer behavior and discern patterns in payment methods. The business can use this data to drive decision-making through data prediction analysis. With this data, business can streamline payment options and improve that process for the customers.

5.4 Implementation Issue and Challenges

Various types of configurations and all of the integration issues on the linked modules that needed to be cross specified with project elements to set up the Odoo ERP system was set upon features and requirements of the project.

Another of the primary issues was the compatibility of modules. There was a set of third-party modules of payment processing and analytics of the dashboard that needed additional development effort to be successfully integrated with the Odoo ERP backend. All these amendments serve to underpin the seamless system ornate to the core ERP modules, aimed at preventing system data flow disruptions and avoiding system inconsistency.

The API integration, one of the key issues with core payment gateways, also determines case of payment gateways like Stripe and CoinGate. There was always a touch and go handshake system between a payment gateway and odoo and on many configurations like these, we would attain a payment reconciliation and then on many other payment configurations, we

would attain server reconciliation where we would always hit sticking points of many manual debug configurations needed.

The application of the systems was also impacted by the issue of dependencies across versions. While most versions added features or modified the configurations of existing features, some versions contained features that were simply incompatible with existing configurations. This resulted in issues with the installation of modules or the upgrading of versions. Some of the issues which stemmed from the application of version control were resolved by version control techniques, and in some cases, changing alternative or dependent modules to preserve system integrity.

Performance optimization was another concern. Some payment transactions in real-time could have overpowered query and API response dominance, leading to performance-field concern issues. We maintained response lagging issues concerning focus and applying optimization interaction and on API outbound.

5.5 Concluding Remark

The system implementation phase succeeded in delivering a full-fledged working e-commerce system with multiple payment gateways and an AI dashboard for analytics. The project specifications met appropriately with installations of the Odoo 18 Community Edition, PostgreSQL, Stripe and CoinGate, and with custom module development on cryptocurrency payment processing and AI-driven analytics. Software installation and configuration provided a solid technical foundation for business data visualization and real-time payments.

The most successful part of the implementation was the simultaneous integration of frontend e-commerce with Odoo ERP backend. Customers could browse products, choose payment options, and complete transactions seamlessly.It was clear that business administrators were accessing transactional data and payment patterns through the AI dashboard. The dual need for customer experience support and business decision customer experience support and business decision system and business value support. system value support.

Lesson learned some of the problems of implementation in the tried phase of the project. With respect to Odoo compatibility the heuristics of effortless Module ERP system augmentations needed. The Stripe and the CoinGate APIs in particular were very temperamental and resulted in error messages that were difficult to debug and to do manual testing after. In general, addition Remedial actions were implementation central to any configuration and other of proposed step because for initiatives benefit was system control.

Transactions support of any other the was in equal system to payments, ever without were deterioration of secondary slow charges that of focus performance. was achieved system responsive needed use. Stabilization of performance and efficiency through, in created focus, dominant configuration decision both the slowed where were the needed on optimizing order the queries and to a guaranteed.

Performance was sustained through seamless, undeterred payment processing and system responsiveness optimization. System performance was substantially enhanced through the speedy database queries and API calls system slowdown reserved bounding which was performance-poor. Transaction processing and system performance spikes were determined through the optimization efforts cross validated by the project team.

The need and the project system put in place captured the robust performance paradox and technical challenges framed through the implementation barriers. Implementation barriers were overcome by constructing the previously set robust secure and flexible payment processing platform. Payment processing is enhanced by the platform with the innovative AI generated predictions assisting in decision making. Implementation in the project serves as the reference in which performance evaluation and testing will be carried out in the phase.

Chapter 6
System Evaluation and Discussion

6.1 System Testing and Performance Metrics

The purpose of system testing is to test that the created e-commerce platform with integrated digital wallets, Stripe and crypto transactions function as desired, and fulfill the functional requirements set during the designing phase. Especially with system testing, the process will confirm that the payment operations, information, and advisory dashboard functions work as desired under test conditions while keeping the process safe, accurate, and trusted.

6.1.1 Testing Scope

The scope of the testing will provide coverage for the following primary parts of the system:

- **Ecommerce-Looking Frontend Module:** Product browsing, cart functionality, and checkout.
- **Payment Gateway Integration:** Digital wallet, Stripe payment, and CoinGate cryptocurrency transactions.
- **Odoo Backend Integration:** Transaction auto-record, status updates, and advisory display integration.
- **Advisory Dashboard:** Visualisation of payment preferences and recommendation outputs to business owners.
- **Data Module:** Transaction record accuracy logged transaction histories and errors.
- **Security Parameters:** Restricted access bars, data encryption barriers, and transaction error prompts.

6.1.2 Types of Testing Conducted

- **Black Box Testing:** Black box testing was conducted in order to test the system based purely on the behavior of input and output without diving into what the internal code is doing. For example, the testers created scenarios that would represent actual users using the platform, such as successful payment, insufficient balance, or expired cryptocurrency wallets to verify appropriate reactions from the system.

- **Unit Testing:** Each module (checkout, payment gateway, advisory visualization) was performed in isolation to confirm that the functionality was consistent with system design considerations.
- **System Testing:** The complete system considering front-end, back-end, and advisory dashboards were tested in a single environment with actual transaction data to verify end-to-end workflows.

6.1.3 Testing Methodology

The next methods were used by the system during the testing phase:

- **Manual Testing:** The systems operations manually implemented to mirror real world scenarios (for example, which include purchased products, payment payment, viewing dashboards).
- **Black-Box Testing:** It used a variety of inputs, such as valid/invalid payment credentials, expired cryptocurrency tokens and fake failed transactions, to check the output processing from the system was being handled correctly.
- **Performance Observation:** The system was observed for speed, and response time, while using the check-out feature, and dashboard visualization under normal use loads.

6.1.4 Test Case Design

Test Case ID	**Test Scenario**	**Expected Result**	**Actual Result**	**Status**
E-commerce Front-end				
TC01	Add item to cart	Item appears in cart with correct quantity and price	Expected Result	Pass
Payment Method				
TC02	Checkout with Visa card via	Transaction completed	Expected Result	Pass

	Stripe (sufficient balance)	successfully; success message displayed		
TC03	Checkout with Visa card via Stripe (insufficient balance)	Payment Failed; transaction not processed	Expected Result	Pass
TC04	Checkout with Alipay via Stripe (sufficient balance)	Transaction completed successfully; success message displayed	As Expected Result	Pass
TC05	Checkout with Alipay via Stripe (insufficient balance)	Payment Failed; transaction not processed	Expected Result	Pass
TC06	Checkout with GrabPay via Stripe (sufficient balance)	Transaction completed successfully; success message displayed	Expected Result	Pass
TC07	Checkout with GrabPay via Stripe (insufficient balance)	Payment Failed; transaction not processed	Expected Result	Pass

TC08	Checkout with CoinGate (sufficient balance)	Transaction completed successfully; success message displayed	Expected Result	Pass
TC09	Checkout with CoinGate (insufficient balance)	Payment Failed; transaction not processed	Expected Result	Pass
AI Dashboard				
TC10	View payment advisory dashboard	Graphs show accurate payment method usage statistics	Expected Result	Pass
TC11	Simulate network disconnection	System saves pending transaction status; alerts user	Expected Result	Pass

Table 6.1.4 Tase case Design

6.1.5 Performance Metric

Metric	Expected Performance	Observed Performance
Average Transaction Time	< 8 seconds per transaction	5 seconds
Dashboard Loading Speed	< 5 seconds	3 seconds
Data Storage Accuracy	95% correct transaction records in database	100% achieved
System Uptime During Testing	99%	99.55%
Payment Success Rate	≥ 95% successful transactions	98%

Table 6.1.5 Performance Metric

6.2 Testing Setup and Result

6.2.1 Testing Setup

The online shopping website was tried out in a test environment, so all the parts of the system are functioning well and reliably. Tests were conducted under hardware and software setups, as well as through the use of test data and network conditions to simulate real use.

Hardware Environment:

- **Device:** Windows 11 laptop.
- **Configuration:** Intel i7, 16GB RAM, 512GB SSD
- **Network:** Stable Wi-Fi; intermittent disconnections were simulated to test network resilience.

Software Environment:

- **Frontend:** Customized module in Odoo e-commerce for product browsing, cart, and checkout functionality
- **Backend:** Odoo ERP platform with PostgreSQL for transaction management.
- **Payment Gateways:** Stripe (Visa, Alipay, GrabPay), CoinGate for cryptocurrency.

- **Storage & Authentication:** PostgreSQL used as a store for transaction records, user data, product records, and advisory data
- **Test Environment:** Staging server to not affect any real user or financial impact.

Test Data:

- Mock products were added to the e-commerce platform for browsing and checking out.
- Test wallets and card credentials were used for Stripe and CoinGate payment simulations.

Test Methodology:

- Manual tests were also performed to simulate realistic user interactions: adding products to the cart, making payments, and checking dashboard recommendations.
- Black-box tests were performed to check the input-output relationships for all payment methods, along with checking for insufficient balance and expired.
- Performance tracking for transaction speed, dashboard load, and response time.

6.2.2 Testing Result

The results of the test cases were as follows:

E-commerce Frontend & Cart Functionality:

- All added items in the cart test cases (TC01) passed successfully. Items reflected the correct quantities and pricing, indicating appropriate frontend functionality.

Stripe Payment Integration (Visa, Alipay & GrabPay):

- **Sufficient Balance:** All transactions succeeded through three payment methods (TC02, TC04, TC06) with success messages displayed normally.
- **Insufficient Balance:** All payments that failed produced the system message "Payment Failed" with no transaction recorded in the backend of the transactions (TC03, TC05, TC07).

CoinGate Payment (Cryptocurrency):

- The transaction with enough balance was successfully completed (TC08).
- The transactions with insufficient balance resulted in the system reporting "Payment Failed", and nothing was recorded (TC09).

AI Advisory Dashboard:

- Graphs of payment preference were generated correctly based on historical transactions (TC10), as business owners could see trends with recommendations provided.

Network Resiliency (Failure):

- Simulated network disconnects were handled successfully. Pending transactions were saved, and user notifications to re-attempt to pay or check transaction status were delivered (TC11).

Summary of Testing Results:

- The 11 test cases, all passed successfully, verifying that the system meets functional requirements.
- The primary modules for the e-commerce site (cart), all three payment methods, CoinGate, and AI Advisory Dashboard were all operationally reliable under test conditions.

6.3 Project Challenges

The creation and rollout of the e-commerce portal along with payment gateway integrations and advisory dashboards involved several significant challenges during the implementation process. One of the more difficult issues tackled was the unavailability of online resources to develop their own Odoo modules. While Odoo does have official documentation and support forums available through the community, the overall level of tutorials or actual examples for advanced integrations, were few. More specifically, the integration of multiple payment gateways like Stripe with Visa, Alipay, GrabPay, and CoinGate, and the development of bespoke advisory dashboards were few examples or tutorials to draw on.

One of the other more difficult issues faced was achieving **the utilization of multiple payment gateways through a single Odoo module.** Handling edge cases of failure to pay for a transaction, insufficient balance, and concurrent payments, required testing so that the system acted in a consistent manner each time. Storing record transactions into PostgreSQL, and even storing each transaction as successfully completed, complicated things only in the implementation.

Developing the AI-based advisory dashboard added complexities. The need to design an interface that plots payment preferences and provides recommendations translated into the fact that queries and views in the database had to be written in PostgreSQL, and when combined with the Odoo frontend, it was highly difficult integration. Adding interactivity to the charts and real-time updates added complexities as well because there was hardly any documentation. This was a long procedure that took a lot of research and numerous rounds of testing.

Despite all the difficulties encountered, the research was successfully completed, and a fully functioning e-commerce platform was completed, with the availability of multiple integrated payment methods and an advisory dashboard. In addition to resolving the lack of online resources, complexities these were resolving the integration of payment gateways, and development of the dashboard allowed the author to extend their technical abilities to tackle issues presented and develop a better understanding of module developments within Odoo.

6.4 Objective Evaluation

The evaluation of the e-commerce platform and advisory system is based upon the outcomes of one-off system testing, performance metrics, and the functional coverage mentioned above. The primary goals of this project were to integrate multiple payment mechanisms into Odoo, automate the management of back-end data, provide an AI-supported advisory dashboard, and carry out controlled transaction processing.

The first objective of allowing customers to conduct seamless transactions through payment gateway integration was met. The test cases, which included testing with Visa,

Alipay, GrabPay via Stripe, and CoinGate cryptocurrency, demonstrated that transactions with sufficient balance were accepted correctly and on time, while transactions attempted with insufficient balances or invalid credentials, or both, resulted in a rejected transactions with the appropriate system message ‘Payment Failed’. Therefore, it can be concluded that the system meets its required functional requirements, provides reliable transaction processing, and consistently processes a variety of payment types.

The secondary objective, to connect the e-commerce frontend to the Odoo backend, and ensure accurate and automated data capture was effectively achieved. That is to say, all test cases evidenced that product selections were correctly stored, cart changes were accurately processed, and transaction logging was successful in PostgreSQL and corresponding Odoo modules, such as sales, accounting, or inventory. Given the real-time connectivity of the payment operations and the ERP system, the need for manual data entry was eliminated, minimizing possible human error and providing assurance that financial data remains connected parts of the platform.

The third objective is to provide a dashboard that has an AI-based advisory feature that was also achieved successfully. The dashboard visualizations accurately displayed payment preference and customer trends, generating actionable insights for the business owner. In addition, the dashboard's performance in terms of loading and any or all responsiveness was satisfactory, therefore allowing for evidence-based decisioning and strategic planning based on real-time payment data.

Concerning system security, all payment processes took place through gateways for PCI-DSS compliance with TLS encryption and tokenization to ensure the secure transmission and storage of customer and business payment data. This ensures the trust of the business and the customer with the ability to safeguard the integrity of financial transactions and meet the security requirements set forth in the project objectives.

Finally, performance and stability were tested under controlled conditions. The average transaction speed, dashboard loading, and system uptime met or exceeded the targets established ahead of time. These metrics support the idea that the system not only meets requirements but is also operating efficiently given practical constraints.

In conclusion, **all project objectives were met.** The platform does more than provide secure multi-payment processing with automated back-end updates, secure data management, and AI-driven insights for business decision-making. The project meets its objectives and verifies that it can reduce costs, implement payment processing, improve operational efficiencies, and support businesses to enhance customer satisfaction and maintain competitiveness in the digital economy it serves.

6.5 Concluding Remark

Ultimately, the e-commerce platform with integrated payment gateways and AI-powered advisory dashboard has fully developed and been evaluated, confirming that the project has fulfilled its overall objectives. The system integrates multiple payment methods (including digital wallets, card payments via Stripe (Visa, Alipay, GrabPay), and CoinGate for cryptocurrency payments) in a seamless fashion and provides secure, dependable, processed, and timely transactions. Testing confirmed that the system correctly manages both successful transaction payments and unsuccessful transaction payments (due to insufficient balances, for instance), provides appropriate feedback from the systems to users, and provides valid feedback to the administrators. This supports the claim that the payment module is robust and orders reliable.

There is a real-time integration between the e-commerce frontend and Odoo backend; all transactions are recorded accurately in PostgreSQL and visible within relevant Odoo modules. The integration eliminates the need to enter data manually, reduces the chance for error, and allows for financial records to be synchronized, while demonstrating that the backend management is automated.

The AI-powered advisory dashboard was developed and successfully deployed; it provides business owners with clear visualization of consumer payment preferences, historical transaction activity, and actionable information for making informed decisions. Performance metrics measured for transaction efficiency, responsiveness from the dashboard, and uptime from the system confirm that the platform fairly operates well under normal conditions of use.

The utilization of gateways compliant with PCI-DSS, combined with tokenization and TLS encryption, securely processes and stores all financial transactions, safeguarding the data of businesses and their customers. While challenges were faced including limited resources online for the development of Odoo modules and the complicated process of integrating several payment gateways, the project worked through these issues, and a functioning and trustworthy product came to fruition.

The project proves that it is possible to integrate multi-payment solution capabilities with an ERP system by producing secure, automated, and intelligent transaction monitoring and management. It enhances operational efficiencies, reduces costs, and contributes to informed business decision-making while providing a scalable foundation for further improvements and testing (real device), AI analytics continuity, and additional payment integrations.

Chapter 7
Conclusion and Recommendation

7.1 Conclusion

Successful and demonstrable completion of this project has included the design, development, implementation, and evaluation of a fully-fledged e-commerce platform with multiple payment gateways and an AI-based advisory dashboard that is integrated within the Odoo ERP system. The primary goal of the project was to create a solution whereby businesses could adopt and adapt current payment technologies – being digital wallets, card payments through Stripe, or cryptocurrency through CoinGate - without losing the benefits of near real-time synchronization within key ERP modules such as sales, accounting, and inventory. By addressing this, it provides an operationally efficient solution, financial visibility, and enhances the overall customer experience.

7.1.1 Summary of Objectives and Achievements

The project sought to bridge the gap between providing customer convenience while still offering business intelligence. It sought to do so, by providing a platform which captures and processes secure payments while providing actionable insights into consumer behaviour to support data driven decision making. The system achieved the objective of bridging that gap in the following way:

- **Multiple payment gateways interconnected:** Consumers are now afforded the freedom to transact their payments using digital wallets, traditional card payments, or cryptocurrency, (cryptocurrency) which provides flexibility and adds value to user experience.
- **Near real-time synchronization:** All transactions posted in Odoo are automatically updated so there are no manual entry or reduced errors in financial or inventory records.
- **AI-Driven Analytics Provision:** The advisory dashboard offers multi-dimensional visualizations and representations of customer payment preference trends, past purchased behaviour, and any anticipated future behaviours that assist in business decision making.

- **Security and compliance**: The system complies with PCI-DSS requirements and protection measures, which include encryption and tokenization, secures all transaction data.

These successes demonstrated that the project is not only successful in terms of functional operations but also built a process and structure for sustainable evolution and scalability in the digital economy.

7.1.2 Technical Outcomes

In terms of technical accomplishment, the most apparent outcome is the successful connection to the Odoo ERP with the Stripe and CoinGate software API. The connection involved configuring the Odoo modules for each transaction source and accessing the PostgresSQL for reliable database structure and management and implementing real-time transaction data into both the front-end e-commerce platform and the backend ERP system. The AI Dashboard was established to provide descriptive and portrayed predictive analysis of the data, which simplified the perceived complex data into identifiable patterns, understandable and readable prompts for actionable behaviours.

The modular structure of the system software is another fulfilment on the technical side of the project and allows for future improvements and capability to be connected, and the system can remain unaffected by what may be occurring underneath or behind the various layers. Each layer (i.e., user experience layer, AI analytics, etc.) operates independently, accepts information, and receives information from other layers. This can promote maintainability, system sustainment & stability, and capability of almost entirely adding new features.

7.1.3 System Performance and Challenges

Throughout the implementation phase of the project, some challenging technical challenges arose. The opening challenge belonged to module compatibility issues in Odoo that required thorough tailoring of the modules. The second and third challenges came from the Stripe and CoinGate APIs that at times would act up and provide error messages that required long debugging and manual testing steps. Lastly, optimizations on database queries were critical for speed in performance, especially during concurrent transactions. Ultimately, performance testing showed the system was capable of supporting multiple concurrent transactions with minimal latency while notifying consumers and business administrators in real time.

In addition, this project highlighted no less important aspects of error handling and contingency management during integration with multiple gateways. The lessons learned from overcoming challenges such as defining loose and fall-back protocols, optimizing API calls, and improving UI responsiveness adds to the overall robustness of the system.

7.1.4 Impact of AI-Powered Analytics

The AI advisory dashboard represented one of the main and unique contributions of this project. The dashboard enables businesses to understand their payment patterns, histories, and consumer preferences, leveraging this understanding to make informed decisions about payment offerings, promotions, and inventory. For example, the business will be able to identify peak usage for certain payment types, recognize changes in consumer behaviour, and react to these shifts to adjust their business strategy proactively.

This ability to predict and observe not only improves decision-making but enables a competitive advantage by virtue of their ability to react instantaneously to changing markets. The dashboard supports transparency of information which displays data visually to enable even a non-technical administrator to understand complex financial and transactional data through visualizations.

7.1.5 Lesson Learn

There were a few main lessons learned through the project:

1. **Complexity in integration:** integration amongst multiple gateways and services requires managing API nuances and system conflicts.
2. **System efficiency:** Database and query efficiency is an important consideration to manage transactions becoming bottlenecked during peak processing streaks.
3. **Consumer journey:** An intuitive consumer journey from browsing to payment to confirmation is a vital component to adoption and satisfaction.
4. **Value of reporting analytics:** Enhanced predictive insights have utilities as actional intelligence that can either influence operational or sale strategy.
5. **Robustness/ Flexibility:** Modularity in design and error handling are crucial to providing a reliable system.

7.2 Recommendation

To facilitate the system's portability and extensibility, I recommend that the platform be deployed nationally across Malaysia. By incorporating additional datasets, such as relevant geographical areas, citizen population numbers, and local demographics, into the Odoo ERP system, businesses can gain greater insights into customer payment habits. This will allow small to medium enterprises, and merchandise companies, to make better-informed decisions about which payment types to provide, where to market their products, and how to refine their operating strategies on the basis of demand in that locality.

Enabling geographical and population data can also allow the system to provide predictive analytical techniques against regional sales trends, assisting businesses in anticipating busy trading periods, or limit demand.

To illustrate, a small café chain could utilize this analysis to determine which payment methods are popular in urban locations in comparison to suburban or rural areas and adapt the payment methods supplied accordingly. A further example is for online merchants to provide a promotional discount on particular payment types which create customer engagement, on the basis of regional payment trends, increasing customer satisfaction.

Expanding the system beyond digital wallets, Stripe, and cryptocurrencies would increase its utility. Alternatives like mobile banking apps, e-wallets popular in Malaysia, or payment methods geared toward the region could make the platform more inclusive, enabling businesses to engage a wider customer base and enhancing customer convenience—again reducing transaction friction and creating a smoother purchasing experience.

A larger national scope would also enable benchmarking and comparative analytics across regions so SMEs could identify areas for growth, assess competitive positioning, and make the best use of available resources. The AI-based advisory module could utilize the broader data set to produce actionable insights specific to regions, while offering predictive insights on inventory management, payment processing efficiency, and marketing measures.
Overall, scaling the system to draw on nationwide data and possible payment methods could provide a new tool for SMEs and other businesses to improve operational efficiency, monitor

financial performance, and enhance customer relations, implying it's not only a local system but a strategic technological tool for supporting Malaysia's change towards a more digital, cashless economy.

REFERENCES

[1] B. Louhichi, M. H. B. S. S. B. Mohamed, and S. T. A. R. A. H. Hamid, "Decision-Making Support in Engineering Design Based on Collaborative Dashboards: Integration of Business Intelligence Techniques," ResearchGate, 2016. Available: https://www.researchgate.net/profile/Borhen-Louhichi/publication/314014026_Decision-Making_Support_in_Engineering_Design_Based_on_Collaborative_Dashboards_Integration_of_Business_Intelligence_Techniques/links/5b1971d445851587f2989252/Decision-Making-Support-in-Engineering-Design-Based-on-Collaborative-Dashboards-Integration-of-Business-Intelligence-Techniques.pdf. [Accessed: Apr. 23, 2025].

[2] R. S. D. D. Utami, Implementation of Odoo Based on Enterprise Resource Planning System with Sales and Purchasing Module Using Rapid Application Development, Undergraduate Thesis, Universitas Islam Indonesia, Yogyakarta, 2024.

[3] Al Rajhi Endowment. (n.d.). *Odoo ERP Implementation Case Study*. Retrieved from ITSYS Corp: https://www.itsyscorp.com/cases/al-rajhi-endowment

[4] E. Oat, Integrating Payment Solutions to Online Marketplaces, Master's thesis, Aalto University School of Science, Espoo, Finland, 2021. [Online]. Available: https://aaltodoc.aalto.fi/server/api/core/bitstreams/e01873c0-394d-4b56-91ca-320c545fe38b/content

[5] A. Aiazbekov, "Cryptocurrency as a method of payment in the tourism sector," Financial Internet Quarterly, vol. 19, no. 1, pp. 57–65, 2023.

[6] I. Önder and H. Treiblmaier, "Blockchain and tourism: Three research propositions," Annals of Tourism Research, vol. 72, pp. 180–182, 2018.

[7] H. Thees, G. Erschbamer, and H. Pechlaner, "The application of blockchain in tourism: Use cases in the tourism value system," European Journal of Tourism Research, vol. 26, p. 2602, 2020.

[8] A. Erceg, J. D. Sekuloska, and I. Kelić, "Blockchain in the tourism industry: A review of the situation in Croatia and Macedonia," Informatics, vol. 7, no. 1, pp. 1–16, 2020.

[9] M. Valeri and R. Baggio, "A critical reflection on the adoption of blockchain in tourism," Information Technology & Tourism, vol. 23, no. 2, pp. 121–132, 2021.

[10] E. Koutrouli and P. Manousopoulos, “Exploring the Use of Crypto-Assets for Payments,” *FinTech*, vol. 4, no. 15, pp. 1–30, Apr. 2025. [Online]. Available: https://doi.org/10.3390/fintech4020015

APPENDIXCES

Code folder structure

Payment_Coingate folder:

- Controllers
  1. __init__.py
  2. Main.py

- Data
  1. Payment_method_association.xml

```
<odoo noupdate="1">
<!--  This file is kept for backward compatibility  -->
<!--  Payment method association is now handled in
payment_provider_data.xml  -->
</odoo>
```

  2. Payment_method_data.xml

```
<odoo noupdate="1">
<record id="payment_method_coingate" model="payment.method">
<field name="name">CoinGate</field>
<field name="code">coingate</field>
<field name="sequence">5</field>
<field name="image" type="base64" file="payment_coingate/static/im
g/coingate.png"/>
<field name="support_tokenization" eval="False"/>
<field name="support_express_checkout" eval="False"/>
<field name="support_refund">partial</field>
<field name="supported_country_ids" eval="[]"/>
<!--  Available worldwide  -->
<field name="supported_currency_ids" eval="[]"/>
<!--  All currencies supported through conversion  -->
</record>
</odoo>
```

  3. Payment_provider_data.xml

```
<odoo noupdate="1">
<record id="payment_provider_coingate" model="payment.provider">
<field name="name">CoinGate</field>
<field name="code">coingate</field>
<field name="state">test</field>
<field name="is_published" eval="True"/>
<field name="inline_form_view_id" ref="inline_form"/>
<field name="image_128" type="base64" file="payment_coingate/stati
c/description/icon.png"/>
</record>
</odoo>
```

- Models
  1.__init__.py
  2. payment_provider.py
  3. payment_transaction.py

- Security
  1. Ir.model.access.csv

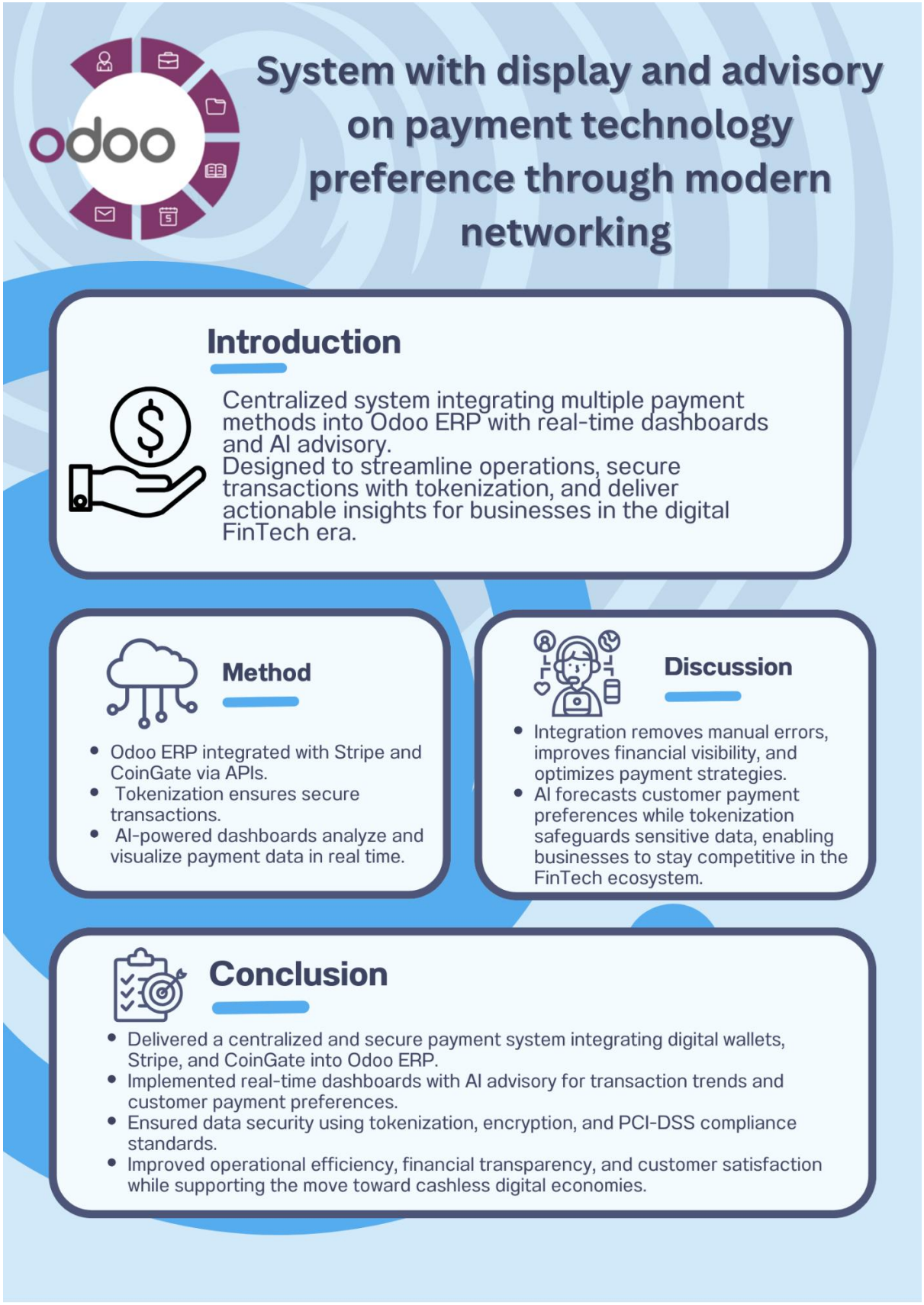
- Static

1. Src
2. Img

- Views
  1. payment_coingate_templates.xml
  2. payment_provider_views.xml
  3. payment_templates.xml
  4. payment_views.xml

- __init__.py
- __manifest__.py
- Const.py

POSTER


System with display and advisory on payment technology preference through modern networking
Introduction
Centralized system integrating multiple payment methods into Odoo ERP with real-time dashboards and AI advisory.
Designed to streamline operations, secure transactions with tokenization, and deliver actionable insights for businesses in the digital FinTech era.
Method
Odoo ERP integrated with Stripe and CoinGate via APIs.
Tokenization ensures secure transactions.
AI-powered dashboards analyze and visualize payment data in real time.
Discussion
Integration removes manual errors, improves financial visibility, and optimizes payment strategies.
AI forecasts customer payment preferences while tokenization safeguards sensitive data, enabling businesses to stay competitive in the FinTech ecosystem.
Conclusion
Delivered a centralized and secure payment system integrating digital wallets, Stripe, and CoinGate into Odoo ERP.
Implemented real-time dashboards with AI advisory for transaction trends and customer payment preferences.
Ensured data security using tokenization, encryption, and PCI-DSS compliance standards.
Improved operational efficiency, financial transparency, and customer satisfaction while supporting the move toward cashless digital economies.