**Vietnam General Confederation of Labor**

**TON DUC THONG UNIVERSITY**

**FACULTY OF INFORMATION TECHNOLOGY**



**Software Engineering**

**FINAL PROJECT SOFTWARE ENGINEERING**

*Instructor*: **PHẠM THÁI KỲ TRUNG**

*Performer*: **NGUYEN HẢI BẰNG - 521H0012**

**ĐẶNG LỮ ANH KIỆT-521H0090**

Class **: 21H50201**

Course  **: 25**

**HO CHI MINH CITY, IN 2023**

**Vietnam General Confederation of Labor**

**TON DUC THONG UNIVERSITY**

**FACULTY OF INFORMATION TECHNOLOGY**



**Software Engineering**

**FINAL PROJECT SOFTWARE ENGINEERING**

*Instructor*: **PHẠM THÁI KỲ TRUNG**

*Performer*: **ĐẶNG LỮ ANH KIỆT-521H0090**

**NGUYEN HẢI BẰNG - 521H0012**

Class **: 21H50201**

Course  **: 25**

**HO CHI MINH CITY, IN 2023**

THANK YOU

First of all, I would like to say thank you to Mr. Trung, in the past time he has been dedicated to imparting new knowledge to me as well as supporting me in my studies. After being instructed by the teacher wholeheartedly, I have learned many things, but there are still many points that I have not grasped, in this final report, we have tried to fix it and there are still many mistakes. skip. I sincerely thank you and wish you good health.

PROJECT COMPLETED

AT TON DUC THONG UNIVERSITY

I hereby declare that this is the product of our project and under the guidance of Mr. Pham Thai Ky Trung. The research contents and results on this topic are honest and have not been published in any form before. The data in the tables for analysis, comments, and evaluation are collected by the author himself from different sources, clearly stated in the reference section.

In addition, the project also uses a number of comments, assessments as well as data from other authors, other agencies, and organizations, with citations and source annotations.

If I find any fraud, I will take full responsibility for the content of my project. Ton Duc Thang University is not related to copyright and copyright violations caused by me during the implementation process (if any).

*Ho Chi Minh City, 07/05/2023*

*Students1*

(signature and full name)



*Nguyen Hai Bang*

PROJECT COMPLETED

AT TON DUC THONG UNIVERSITY

I hereby declare that this is the product of our project and under the guidance of Mr. Pham Thai Ky Trung. The research contents and results on this topic are honest and have not been published in any form before. The data in the tables for analysis, comments, and evaluation are collected by the author himself from different sources, clearly stated in the reference section.

In addition, the project also uses a number of comments, assessments as well as data from other authors, other agencies, and organizations, with citations and source annotations.

If I find any fraud, I will take full responsibility for the content of my project. Ton Duc Thang University is not related to copyright and copyright violations caused by me during the implementation process (if any).

*Ho Chi Minh City, 07/05/2023*

*Students2*

(signature and full name)

*Dang Lu Anh Kiet*

**TEACHER'S CONFIRMATION AND ASSESSMENT SECTION**

**The confirmation part of the instructor**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Ho Chi Minh City, in 2023

(signature and full name)

**The evaluation part of the teacher marks the test**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Ho Chi Minh City, in 2023

(signature and full name)

SUMMARY

Requirements collection: First, the development team collected information about the customer's requirements for functionality, interface, features, and other requirements related to the application.

Design: The development team then designed the user interface, database, and other features of the application.

Development planning: The development team made a plan to develop the application, including the schedule, work assignment, and development steps.

Development: After completing the above steps, the development team started building the online phone selling application according to the approved designs.

Testing: Once the application is complete, the development team has performed tests and checked the functionality of the application to ensure that it is working properly and without errors.

Deployment: After testing is completed, the application is deployed and running on the mobile platform.

Maintenance and updates: The development team continues to maintain and update the application to ensure that it always works well and meets the requirements of the users.

Due to limited technical expertise, the report still contains several deficiencies, and the code section remains incomplete.

Overall, this development process has helped the development team to create an online phone selling application with good features and attractive interface, meeting the requirements of customers.

TABLE OF CONTENTS

Contents

[Chapter 1 - Introduction 10](#_Toc134555240)

[1.1Purpose and Scope: 10](#_Toc134555241)

[1.2Product Overview (including capabilities, scenarios for using the product, etc.) 10](#_Toc134555242)

[1.3 Structure of the Document 11](#_Toc134555243)

[1.4 Terms, Acronyms, and Abbreviations 12](#_Toc134555244)

[2.Project Management Plan 12](#_Toc134555245)

[2.1 Project Organization 12](#_Toc134555246)

[2.2. Lifecycle Model Used 12](#_Toc134555247)

[2.3 Risk Analysis 16](#_Toc134555248)

[2.4 Hardware and Software Resource Requirements 19](#_Toc134555249)

[2.5 Deliverables and Schedule 19](#_Toc134555250)

[2.6 Monitoring, Reporting, and Controlling Mechanisms 20](#_Toc134555251)

[2.7. Professional Standards 21](#_Toc134555252)

[2.8. Evidence all the artifacts have been placed under configuration management 22](#_Toc134555253)

[2.9. Impact of the project on individuals and organizations 22](#_Toc134555254)

[3. Requirement Specification 23](#_Toc134555255)

[3.1.Stakeholders for the system: 23](#_Toc134555256)

[3.2. Use case model: 24](#_Toc134555257)

[3.2.1. Graphic use case model 24](#_Toc134555258)

[*Table use case:* 27](#_Toc134555259)

[3.2.2. Textual Description for each use case: 56](#_Toc134555260)

[**3.3 Rationale for your use case model:** 57](#_Toc134555261)

[3.4 Non-functional requirements: 58](#_Toc134555262)

[**4. Architecture** 59](#_Toc134555263)

[4.1. Architectural style(s) used: 59](#_Toc134555264)

[4.2. Architectural model: 60](#_Toc134555265)

[4.3. Technology, software, and hardware used: 61](#_Toc134555266)

[4.3.1 Technology 61](#_Toc134555267)

[4.3.2. Software 61](#_Toc134555268)

[*4.3.3. Hardware*: 62](#_Toc134555269)

[4.4. Rationale for your architectural style and model: 62](#_Toc134555270)

LIST OF SYMBOLS AND ABBREVIATIONS

LIST OF TABLES, PICTURES, GRAPHS

**Tables use case:**

[*Create customer account*](#_Toc134222959)

[*Add to cart*](#_Toc134222960)

[*View receipts*](#_Toc134222961)

[*Create import receipts*](#_Toc134222962)

[*Create Export receipts*](#_Toc134222963)

[*View Analysis Data*](#_Toc134222964)

[*Print report*](#_Toc134222965)

[*Register*](#_Toc134222966)

[*Login*](#_Toc134222967)

[*View order*](#_Toc134222968)

[*Make order*](#_Toc134222969)

[*Pay by cash*](#_Toc134222970)

[*Online payment*](#_Toc134222971)

[*Make payment*](#_Toc134222972)

[*Choose Address*](#_Toc134222973)

[*Search by name*](#_Toc134222974)

[*View Phones*](#_Toc134222975)

[*Search by Manufacturer*](#_Toc134222976)

[*Choose Reseller Order*](#_Toc134222977)

[*Update Order Status*](#_Toc134222978)

[*Update payment status*](#_Toc134222979)

[*View Best-Selling product*](#_Toc134222980)

[*View Monthly Income*](#_Toc134222981)

[*View Product Data*](#_Toc134222982)Chapter 1 - Introduction

1.1Purpose and Scope:

-Purpose:

The purpose of this document is to define the requirements and specifications for the development of a website that sells phones online.

-Scope:

Product Catalog, Search Functionality, Registration and Payment Systems, Shipping and Delivery Systems, Security and Privacy Measures.

1.2Product Overview (including capabilities, scenarios for using the product, etc.)

Online Phone Web is an online shopping place for mobile phone products. Products include mobile devices such as smartphones, tablets, smart watches and phone accessories.

This website offers customers a wide range of products from well-known and reputable brands, including Samsung, Apple, Xiaomi, Oppo, Vivo, Realme, Huawei, Nokia and many more. Customers can easily search and choose the product that suits their needs.

Additionally, the site provides detailed product information, including specifications, pricing, and previous customer reviews. Customers can order online and pay online through various methods such as credit card, debit card or bank transfer.

To ensure customer satisfaction, the site offers a 30-day return policy from the date of purchase. In addition, the website also provides customer support services via phone, email or online chat to answer customer inquiries and assist customers when needed.

In short, the website that sells mobile phones online is a reliable and convenient shopping place for those who are looking for mobile phone products.

1.3 Structure of the Document

The structure of an online mobile phone store's website document may vary depending on the specific design and layout of the site. However, some common sections that are typically included in such a website are:

Homepage: The homepage is the first page that users see when they visit the website. It usually contains a banner or carousel showcasing the latest promotions or new arrivals, and may also feature some popular or recommended products.

Product Categories: This section includes a list of different categories of mobile phones such as "Smartphones," "Feature phones," "Tablets," "Accessories," etc. Users can click on these categories to browse the products available in each category.

Product Listings: This section displays a list of products available for sale on the website, typically presented as thumbnail images with brief descriptions and prices. Users can click on each product to view more detailed information, including features, specifications, and customer reviews.

Product Detail Pages: These pages provide more detailed information about each individual product, including high-resolution images, detailed specifications, customer reviews, and options for customization (such as color or storage capacity).

Shopping Cart: The shopping cart allows users to add products to their cart for purchase. It typically shows a summary of the items in the cart, the total price, and an option to proceed to checkout.

Checkout: The checkout page provides users with options to enter shipping and billing information, choose a payment method, and confirm their purchase.

Customer Service: This section typically provides information on how to contact customer support, including email, phone, and chat options. It may also include an FAQ section to answer common questions..

* 1. Terms, Acronyms, and Abbreviations

2.Project Management Plan

2.1 Project Organization

Dang Lu Anh Kiet

Nguyen Hai Bang

2.2. Lifecycle Model Used

We decided to adopt the waterfall model for our software development project because it fits our needs and goals very well. Because it is suitable for small and medium sized software projects, saving time and resources. It allows the software development steps to be done in sequence and clearly. This makes time and resource management easier.

The waterfall model is very simple and convenient for us to use. It allows us to set and document our requirements and specifications clearly at the beginning of the project. It also helps us to assign tasks and roles to our team members based on their expertise and skills. We are able to carry out the work process in a systematic and orderly manner without any mess or confusion. The waterfall model also allows us to thoroughly test and secure our systems before deploying to customers.

Feasibility study is the first stage of the software development process. It is intended to evaluate the feasibility of a software project. This process helps to determine if the project is economically, technically and business feasible. In this phase, the problem to be solved with the software is clearly defined and the various alternatives are identified. The comparison is based on criteria such as costs, benefits, risks and feasibility. If the project is deemed viable, decisions on investment, budget, schedule and resource allocation can be made.

Feasibility study is an important stage in software engineering as it helps to ensure that the software product will meet the needs and expectations of the users. This phase also helps us avoid wasting time and resources on a project that is not feasible or does not bring the expected benefits. The feasibility study will evaluate technical, financial as well as market and user research for comparison and decision making.

The requirements analysis and specification phase is one of the following important steps of software development. It consists of two main tasks: requirement generation and specification. At this stage, the customer requirements are clearly and accurately recorded and become the required functions and features of the software, meeting the requirements and expectations of the customer for the program.

Requirements analysis and specification is the task of collecting and analyzing customer requirements. The goal is to find and resolve any issues, ambiguities or inconsistencies in the requirements and confirm them with the User. This ensures that the requirements are accurate and relevant to the actual problems and goals the client wants to achieve.

Design: This phase involves converting the requirements specified in the analysis into a form that can be implemented in a programming language. It includes both detailed and specific design documentation for the software system, including architectural drawings, database design, user interface design, and source code. The software design document was created to document all this work.

The purpose of the design phase is to define how to implement the customer requirements identified at the requirements specification and analysis stage into more specific specifications and a logical and physical structure that computer can do. It also ensures that the software designed will meet functional and non-functional requirements. at a high design level focusing on the global aspects of the system, its interface and data flow. The low-level design details the details into specific units and the internal logic, algorithms, and data structures of each module. A Software Design Document (SDD) is a document that documents the design decisions and rationales made during this phase. It serves as a blueprint for software system development and testing and can also be used as a reference for future software improvements.

Coding and unit testing are two essential processes in software development. This process ensures that the software design is translated into executable source code that meets the desired functionality and quality. Before coding begins, developers will use the design documents generated during the design phase to match the user's requirements and desires. Unit testing is the process of testing the correctness and reliability of each module by running various tests and tests to ensure that the written code works as expected. Unit testing can be performed by the writer himself or by a separate test team using different tools and methods.

Integration and system testing are two essential stages in the software development process. These steps ensure that the software meets the functional and non-functional requirements and performs correctly in different situations.

In software development, integration testing is the process of combining separately tested modules to assess their interactions and dependencies. The approach to integration testing can be either bottom-up or top-down, depending on the order of module integration. This process is typically carried out incrementally until all modules are integrated and tested together to ensure the software functions properly.

The final testing stage before delivering the software to the customer is system testing. It aims to ensure that the software aligns with the specifications and fulfills the user's requirements.

System testing comprises three testing types: alpha testing, beta testing, and acceptance testing.

-The development team performs alpha testing in a controlled environment to identify any software issues.

-Beta testing is carried out by a selected group of potential users or customers in a real-world environment to gather feedback and suggestions for improvement.

-Acceptance testing is done by the customer after receiving the software to determine if it meets their criteria and expectations.

Software upkeep is a critical phase in the software development process. It constitutes 60% of the overall effort required to create a complete software product.

Maintenance can be categorized into three types: corrective, perfective, and adaptive.

Corrective maintenance refers to the process of addressing defects that were not discovered during the development phase.

Perfective maintenance involves enhancing the software's features based on customer feedback and requests.

Adaptive maintenance involves modifying the software to function in a new environment, such as a different operating system or hardware platform.

Software maintenance is the process of keeping software up to date with the latest customer needs and modifying the software post-implementation. This involves modifying and updating the software after it has been launched. . Enhance and improve the performance of the software. Software development continues even after it has been launched. Update software to keep up with changing customer needs. Any work done on the software after it has been deployed is considered maintenance.

Software upkeep can be demanding and expensive, as it demands experienced developers, testers, and supervisors to guarantee the software's excellence and dependability. Additionally, software upkeep entails dealing with altering requirements, developing technologies, and outdated systems. As a result, software maintenance should be methodically planned and supervised to decrease risks and maximize advantages.

2.3 Risk Analysis

Risk planning is a critical aspect of any project, as it involves identifying, evaluating and mitigating potential threats that may affect the project's objectives, scope, quality, cost and schedule. These risks include technical, operational, financial, legal and environmental factors. To address these risks, we have established a risk management plan that outlines the strategies, actions and responsibilities for each risk. The plan also specifies the criteria for monitoring and controlling the risks throughout the project lifecycle. The objective of the plan is to minimize the adverse impacts of the risks and maximize the positive opportunities that may arise from them. The plan is based on a cost-benefit analysis of the available risk management options.

Businesses are often confronted with the task of managing potential risks that could impede their goals and performance. Risk management is a structured approach that involves recognizing, evaluating, and reacting to these risks with the aim of minimizing negative consequences and maximizing potential benefits. The use of risk management in business has both advantages and disadvantages, which are outlined below:

-Advantages:

+Risk management can assist a business in mitigating losses by anticipating and planning for potential risks, as well as developing contingency plans.

+Risk management can aid a business in quantifying and prioritizing risks based on their likelihood and impact, and allocating resources appropriately.

+Risk management can assist a business in safeguarding its assets, improving its processes, and enhancing its reputation by demonstrating its commitment to quality and safety.

-Cons:

+Risk management involves a certain degree of uncertainty, as it is based on estimates and assumptions that may not always be accurate or reliable, especially for certain types of risks.

+While risk management can help a business prepare for and mitigate potential risks, it cannot guarantee protection against unforeseen events that may have severe consequences for the business, such as natural disasters or cyberattacks.

There are a number of common risks involved that need to be identified and mitigated when developing a software project. Can be classified into five main areas: people, operations, reputation, finance and technology.

People risks are issues that arise when working with a team of project members. Examples of people risks include poor communication, coordination, or collaboration among team members, mismatched skills and tasks, low motivation, commitment, or accountability, insufficient knowledge or expertise, and health issues or personal problems.

Operational risks are the obstacles that arise during the project's execution. Examples of operational risks include unclear or unrealistic allocation of roles and responsibilities, inefficient or inappropriate work methods or processes, poor control or management of resources, schedule or quality, loss or damage to project components or deliverables, and external factors that impact project performance.

In the context of a project or business, reputation risks refer to potential threats to the organization's image or credibility. These risks can arise from a variety of sources, such as conflicts among team members, unethical behavior, poor quality or reliability of products or services, low customer satisfaction or loyalty, or legal and regulatory issues.

Financial risks are uncertainties that can affect a project's budget or revenue. These risks can take various forms, such as financial instability, low project value, competition or low market demand, and difficulty in raising capital or investors.

In a project, technology risks may arise from the use or development of technology. These risks include outdated or incompatible technology, lack of functionality, frequent bugs, slow performance, unresponsiveness, and difficulty updating or improving technology.

I am feeling unsure about my ability to take on this project due to my limited qualifications. I have a misunderstanding of the topic and am uncertain if I possess the necessary knowledge and skills to successfully complete it. My initial contact with the project has left me anticipating further surprises. My lack of deep understanding in this field has made it difficult for me to comprehend customer expectations.

There are various methods to enhance risk management, including:

-Identifying high-risk areas and evaluating the system structure and subjective risks.

-Prioritizing risks based on their potential impact and addressing them accordingly.

-Conducting a risk assessment analysis to create a risk improvement plan: this may involve accepting certain risks but developing a plan to mitigate their effects when they occur, including actions to take as part of the business continuity plan.

-Employing risk management techniques such as avoidance, retention, sharing, transferring, and loss prevention and reduction.

2.4 Hardware and Software Resource Requirements

**Hardware Requirements:**

1 Gb hard free drive space

512 MB RAM

**Software Requirements:**

C#

Figma

MS SQL

Visual Diagram

MS Word 2017

Web browser: Microsoft Internet Explorer, Mozilla, Google Chrome

2.5 Deliverables and Schedule

-Deliverables

+Functional Requirements Document

+User Interface Design (User Interface Design)

+Source Code and Database

+User Manual

+System Testing (System Testing)

-Schedule:

+Week 1: Requirements gathering and development planning

+Week 2-3: UI Design and Key Features

+Week 4: Programming, Database Development and System Testing

+Week 5-6: Check the system, user manual and exploit the product.

Before handing over the product, we will have a demo presentation to get feedback from customers.

On the expected delivery date, all product functions have been completed according to customer feedback.

2.6 Monitoring, Reporting, and Controlling Mechanisms

The implementation of monitoring and evaluation (M&E) can assist an organization in measuring and enhancing the performance and effectiveness of its operations. This entails gathering and analyzing data on the progress, outcomes, and challenges of the organization's projects or programs. The implementation of M&E can benefit various stakeholders, including:

-Beneficiaries: By utilizing M&E, organizations can meet the needs and expectations of target groups more effectively, resulting in better services and outcomes.

-Employees: The implementation of M&E can instill a sense of purpose and enthusiasm in workers regarding the organization's vision and future. It can also provide opportunities for employees to learn new skills and develop innovative solutions for their work.

-Executive management: M&E implementation can enable executive management to make more informed and strategic decisions by providing accurate and timely data on project performance and impact. It can also aid management in being more adaptable and responsive by identifying emerging issues and opportunities.

- Funders: The implementation of M&E can provide evidence of an organization's results and capabilities, which can increase the confidence and satisfaction of sponsors and donors. M&E can also enhance transparency and accountability by demonstrating how resources are being used effectively and efficiently.Creating a Monitoring and Evaluation Plan.

Quantitative vs. Qualitative

Defining appropriate metrics and indicators is essential for effective monitoring and evaluation of a program. These indicators should be identified and established during the planning phase to enable accurate measurement and assessment of the program's success. This allows organizations to evaluate whether their intended outcomes and objectives are being met and make adjustments as necessary.

Supervising and monitoring project activities involve the following tasks:

Comparing the actual results of the project with the planned ones to evaluate the progress and quality of the project.

Assessing performance and analyzing any deviations from the plan to determine whether corrective or preventive actions need to be taken.

Identifying new risks, analyzing, collecting evidence, and monitoring existing risks to ensure that risks are identified, documented, and risk response plans are in place.

Maintaining an updated and accurate database of project outputs and related project documents throughout the implementation phase.

Collecting information on the effectiveness and impact of project adjustment solutions implemented to assess their outcomes..

2.7. Professional Standards

Software engineers have a responsibility to ensure that the development and maintenance of software is a valuable and respected profession. They must prioritize the safety, health, and well-being of the public and uphold the Eight Principles, which include analysis, specification, design, development, testing, and maintenance of software:

1. PUBLIC – Software engineers shall act consistently in the public interest.

2. CLIENT AND EMPLOYER – Software engineers shall act in a manner that is in the best interests of their client and employer and consistent with the public interest.

3. PRODUCT – Software engineers shall ensure that their products and related modifications meet the highest professional standards possible.

4. JUDGMENT – Software engineers shall maintain integrity and independence in their professional judgment.

5. MANAGEMENT – Software engineering managers and leaders shall subscribe to and promote an ethical approach to the management of software development and maintenance.

6. PROFESSION – Software engineers shall advance the integrity and reputation of the profession consistent with the public interest.

7. COLLEAGUES – Software engineers shall be fair to and supportive of their colleagues.

8. SELF – Software engineers shall participate in lifelong learning regarding the practice of their profession and shall promote an ethical approach to the practice of the profession.

2.8. Evidence all the artifacts have been placed under configuration management

2.9. Impact of the project on individuals and organizations

Impact on individuals

Through our participation in this project, we gained a valuable understanding of what a real project entails and were able to apply our skills and knowledge to create an effective solution for the client. This project also provided us with the opportunity to learn important skills, such as teamwork, effective communication, accountability, and the development of both hard and soft skills. Furthermore, this project helped us identify our passions, interests, strengths, and weaknesses. While we encountered some challenges and disagreements throughout the project, we were able to overcome them by valuing and comprehending each other's perspectives. Overall, this project was a memorable experience that equipped us for our future endeavors.

The societal impact of this project is significant, as it aims to create and deliver a practical product that is highly applicable, usable, and satisfying to users. The project has the potential to contribute to the betterment of society, and we are motivated to learn and work hard in this area despite our limited experience. Our team is committed to producing valuable and beneficial products through our hard work, curiosity, and dedication.

3. Requirement Specification

3.1.Stakeholders for the system:

The online shopping system is a web-based application that enables various user groups to perform different functions related to online shopping. The online shopping system offers the following advantages and functionalities for each user group:

-Customers: Individuals or businesses who shop online and expect a seamless, secure, and convenient shopping experience. They also expect to receive high-quality products and timely customer support.

-Vendors: Companies or individuals who sell their products through the online shopping system. They rely on the system to reach a wider audience and manage their inventory, orders, and payments. Vendors can reach a broader audience and increase their sales with the online shopping system.

- Customer service representatives: Employees responsible for answering customer inquiries and resolving issues related to orders, payments, returns, and shipping. They rely on the system to access customer information, order history, and product details. Customer service representatives can provide high-quality customer support with the online shopping system.

- Marketing team: The marketing team can promote the online shopping system through various channels such as social media, email campaigns, and online ads. They Employees responsible for promoting the online shopping system through various channels such as social media, email campaigns, and online ads. They rely on the system to track customer behavior, analyze data, and measure the effectiveness of their marketing efforts.

- IT team: The IT team can develop, maintaining, and securing the online shopping system. They ensure that the system is up-to-date, scalable, and free of bugs and security vulnerabilities. They can also monitor the system performance and troubleshoot any technical issues in the online shopping system. IT team can provide a reliable and robust online shopping system.

- Executives: Executives can oversee the strategic direction and performance of the online shopping system. Top-level managers who oversee the strategic direction and performance of the online shopping system. They rely on the system to provide them with timely and accurate data and insights that can help them make informed decisions.

3.2. Use case model:

3.2.1. Graphic use case model

A graphic use case model is a visual representation of the use case model, typically in the form of a diagram that helps to better understand the interactions between actors and the system. It allows stakeholders to easily see how the system will behave in different scenarios.

The graphic use case model typically consists of use case diagrams, which show the use cases, actors, and relationships between them. The diagram consists of ovals representing the use cases, stick figures representing the actors, and lines representing the relationships between them. It can also include additional information, such as the conditions under which a use case can be performed, and the outcomes of the use case.

-Let's consider an ATM system. A use case diagram for this system could include use cases such as:

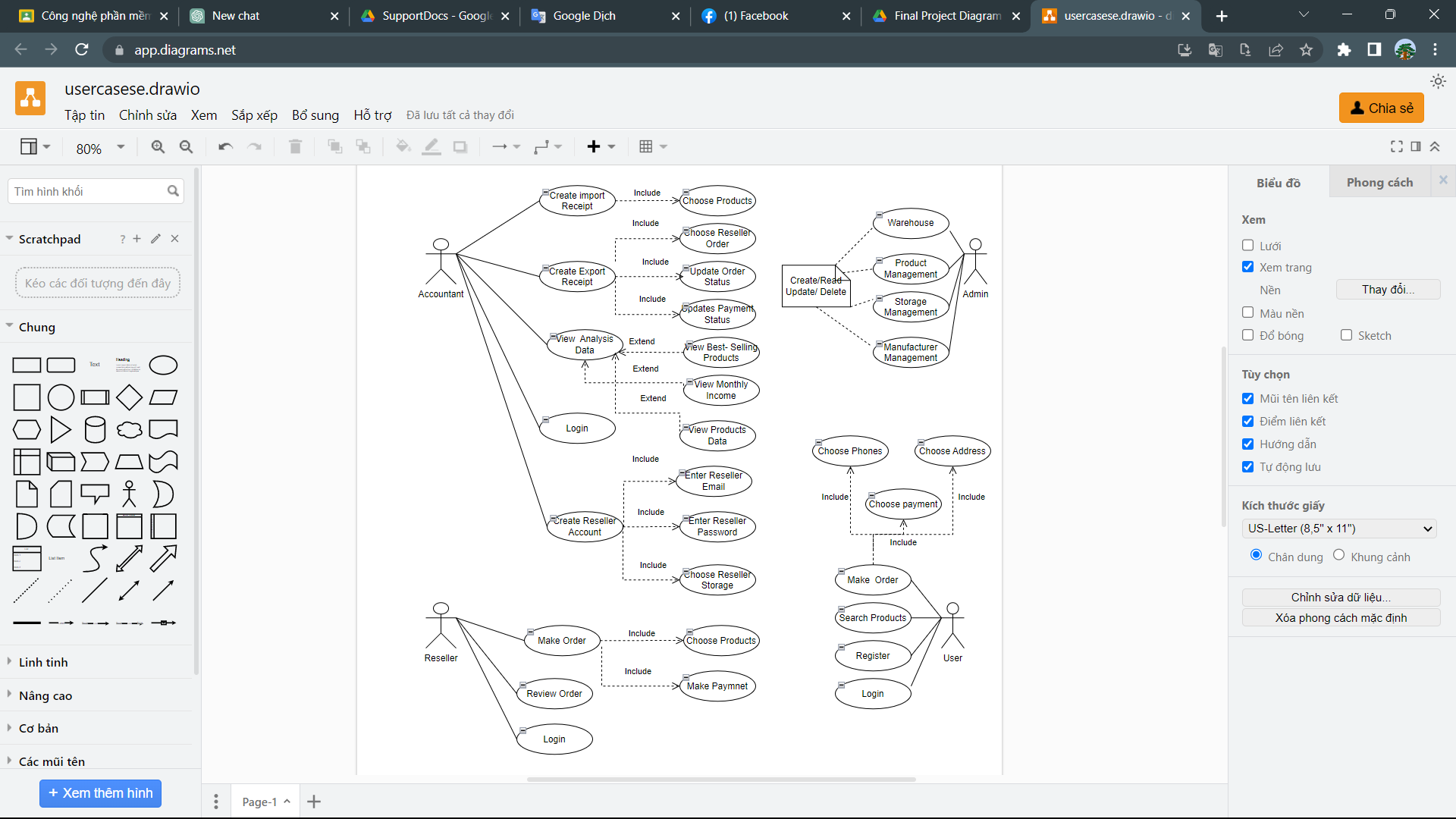
- Withdraw Cash: The actor (customer) requests to withdraw cash from their account.

- Check Balance: The customer requests to check their account balance.

- Transfer Funds: The customer requests to transfer funds between accounts.

The diagram would show the relationships between these use cases and the actors. For instance, the Withdraw Cash use case would be connected to the Customer actor, while the Transfer Funds use case would be connected to both the Customer actor and the Account System actor.

Overall, the graphic use case model provides a clear and concise overview of the system's behavior, making it an effective tool for communication and collaboration among stakeholders.



## *Table use case:*

*Create customer account*

|  |  |  |
| --- | --- | --- |
| **Use case name :** | *Create customer account.* | |
| **Scenario:** | *Create online customer account.* | |
| **Triggering event:** | *New customer to set up account online.* | |
| **Brief description:** | *Online customer creates customer account by entering basic information*  *and then following of with one or more addresses and a credit of debit card.* | |
| **Actors:** | *Customer.* | |
| **Related use cases:** | *Might be invoked by the Check out shopping cart use case.* | |
| **Stakeholders:** | *Accounting, Maketing, Sales.* | |
| **Preconditions:** | *Customer Account subsystem must be available.*  *Credit/debit authorization services must be available.* | |
| **Postconditions:** | *Customer must be create and saved.*  *One or more addresses must be create and saved.*  *Credit/debit card information must be validated.*  *Account must be created and saved.*  *Adddress and Account must be associated with Customer.* | |
| **Flow of activities:** | **Actor** | **Sytem** |
| *1.Customer indicates desire to create customer account and enters basic customer infomation.*  *2. Customer enter one more addresses.*  *3. Customer enters credit/debit card information* | *1.1 System creates a new customer.*  *1.2 System prompts for customer addresses.*  *2.1 System creates addresses.*  *2.2 System prompts for credit/debit card.*  *3.1 System creates account*  *3.2 System verifies authorization for credit/debit card.*  *3.3 System associates custoner, address and account.*  *3.4 System returns valid custom account details.* |
| **Exception**  **Conditions:** | * 1. *Basic customer data are incomplete.*   *2.1 The address isn’t valid.*  *3.2 Creadit/debit information isn’t valid* | |

*Add to cart*

|  |  |  |
| --- | --- | --- |
| **Use case name :** | *Add to cart* | |
| **Scenario:** | *Customers can add products to their cart.* | |
| **Brief description:** | *Customers can add products to their cart to purchase later. After adding products, customers can review their cart to check product information and total amount.* | |
| **Actors:** | *Customer.* | |
| **Stakeholders:** | *Accounting, Maketing, Sales.* | |
| **Preconditions:** | *Customer has logged into the account*  *Credit/debit authorization services must be available.* | |
| **Postconditions:** | *Customer must be create and saved.*  *Credit/debit card information must be validated.*  *Account must be created and saved.*  *Adddress and Account must be associated with Customer.* | |
| **Flow of activities:** | **Actor** | **Sytem** |
| *1.The customer visits the product page and selects the product to buy.*  *2.Customers choose the number of products to buy.*  *3.Customers can continue shopping or check out the cart by clicking the cart icon.*  *4.If the customer wants to remove the product from the car.* | *1.The system displays product information, including name, photo, price and product description.*  *2.The system adds products to the customer's cart.*  *2.1displays cart information, including product name, quantity, price and total amount.* |
| **Exception**  **Conditions:** | *1.1.Invalid goods quantity*  *2.1.Customer data are incomplete.*  *3.1.* *Invalid form of payment.* | |

*View receipts*

|  |  |  |
| --- | --- | --- |
| **Use case name :** | *View receipts* | |
| **Scenario:** | *User can review receipts.* | |
| **Brief description:** | *Users can review previously created invoices to check spending information.* | |
| **Actors:** | *Customer.* | |
| **Stakeholders:** | *Accounting, Maketing,* | |
| **Preconditions:** | *Customer has logged into the account.*  *The customer has paid for the order.*  *Valid invoice information.* | |
| **Postconditions:** | *Account must be created and saved.*  *Adddress and Account must be associated with Customer.*  *Successful invoice search: the user uses the search function to search for a specific invoice.*  *User can close bill view function.* | |
| **Flow of activities:** | **Actor** | **Sytem** |
| *1.Users access the application and select the "View invoice".*  *2.Users can select an invoice from the list or use the search feature to search for invoices by date,amount spent or category name.*  *3.* *If the user wants to print or issue an invoice.* | *1.The system displays a list of previously created invoices, including information about creation date, spending amount and category name.*  *2.The system displays detailed information about the selected invoice, including information about creation date, amount spent, category name and description.* |
| **Exception**  **Conditions:** | *1.* *There are no receipts to display*  *2.* *Invoice not found and invoice details not displayed.*  *3.* *Connection error, check internet connection or try again later.* | |

*Create import receipts*

|  |  |  |
| --- | --- | --- |
| **Use case name :** | *Create import receipts* | |
| **Scenario:** | *Customer can create vouchers* | |
| **Brief description:** | *The user starts running the "Create Invoice" function, the system shows the user a web page with fields to fill in information.* | |
| **Actors:** | *Customer.* | |
| **Stakeholders:** | *Import Managers, Warehouse Managers, Suppliers* | |
| **Preconditions:** | *The import manager must be logged in to the system with appropriate access rights to create import receipts.*  *The system must be connected to the database containing information about the supplier, products, and inventory levels.*  *The system must have a valid and up-to-date list of product codes and prices.*  *The system must have a way to validate the accuracy of the product codes and quantities entered by the import manager.* | |
| **Postconditions:** | *The system displays a confirmation message to the user that the import receipt has been created successfully.*  *The system updates the database with the new import receipt information and the inventory quantity in the warehouse.* | |
| **Flow of activities:** | **Actor** | **Sytem** |
| *1.The user enters the input invoice code. This is required and must not match any other input code in the system.*  *2.The user enters the date to import the goods. This is required and must be in dd/mm/yyyy format.*  *3.User can enter more information about shipper, consignee or any other relevant information.* | *1.The system needs to provide an interface for the user to enter the information of the order entry.*  *2.The system must update the corresponding inventory levels in the warehouse when the order receipt is created.*  *3.The system must verify the accuracy of the information entered by the user.*  *4.The system needs to generate a number of unique receipts and store them.* |
| **Exception**  **Conditions:** | *1. If the user enters a duplicate import receipt code, the system displays an error message.*  *2. If the user enters an invalid date format, the system displays an error message.*  *3. If the user enters a non-existent product code or a quantity that exceeds the available inventory quantity, the system displays an error message* | |

*Create Export receipts*

|  |  |  |
| --- | --- | --- |
| **Use case name :** | *Create Export receipts* | |
| **Brief description:** | *This function allows warehouse staff to create delivery notes to confirm the release of goods to customers.* | |
| **Actors:** | *Warehouse worker, System* | |
| **Stakeholders:** | *Warehouse staff, Inventory management, customer* | |
| **Preconditions:** | *The warehouse staff has logged into the system.*  *Information about products, customers, quantity of goods to be exported has been fully updated in the system.* | |
| **Postconditions:** | *The system successfully created the delivery note and updated it in the database.* | |
| **Flow of activities:** | **Actor** | **Sytem** |
| *1.The user selects the function to create a delivery note on the system interface.*  *2.* *The user enters information about the delivery note, including the date of shipment, the dispatcher, the shipping unit, and the products shipped.*  *3.* *Users select products to export from the list of available products on the system.* | *1.* *The system displays an interface to create a delivery note, allowing users to enter information about the delivery note.*  *2.* *The system checks the delivery note information and the selected product information*  *3.* *The system displays detailed information about the delivery note.* |
| **Exception**  **Conditions:** | *If the system cannot save the delivery slip information due to a connection error or other system error, the system will notify the warehouse staff that the error has occurred and request a retry later.* | |

*View Analysis Data*

|  |  |  |
| --- | --- | --- |
| **Use case name :** | *View Analysis Data* | |
| **Brief description:** | *Allows users to view reports and analyze data related to the business operations of the enterprise.* | |
| **Actors:** | *Maketting, User management, Financial staff.* | |
| **Stakeholders:** | *User management, Financial staff, sytem.* | |
| **Preconditions:** | *The user is logged in to the system and has access to the analysis data.*  *The system must have complete and accurate analytical data to display to the user.* | |
| **Postconditions:** | *Users have seen the analytical data they require and can use it to make specific business decisions.* | |
| **Flow of activities:** | **Actor** | **Sytem** |
| *1. The user accesses the system's software or website and logs in to the system.*  *2. Users select the "View Analysis Data" function on the interface of the software or website.* | *1. The system displays reports and analytical data that the user has access to, including sales reports.* |
| **Exception**  **Conditions:** | *1.* *Users do not have access to reports or analytical data.*  *2. Analysis data does not exist or is faulty.* | |

*Print report*

|  |  |  |
| --- | --- | --- |
| **Use case name :** | *Print report* | |
| **Brief description:** | *Allows users to output reports, lists or other documents from the system.* | |
| **Actors:** | *User* | |
| **Stakeholders:** | *User, Technical support staff, Manage* | |
| **Preconditions:** | *The user is logged into the system and has access to the reports or documents to be printed.*  *The system has prepared the data to be printed and displays them on the user interface.* | |
| **Postconditions:** | *The user has printed a report or list from the system.*  *The data to be printed is not changed or deleted from the system.* | |
| **Flow of activities:** | **Actor** | **Sytem** |
| *1. The user accesses the report or list to print.*  *2. The user selects the report or list printing feature from the user interface.*  *3. The system opens the printing browser on the user's computer and displays the report content.*  *4. User selects printing options like paper size, number of prints, colors, etc.* | *1. The system opens the printing browser on the user's computer and displays the report or list content to be printed.* |
| **Exception**  **Conditions:** | *1. The user does not have access to the report or to-print list.*  *2. The system encountered a technical problem or an error during printing.* | |

*Register*

|  |  |  |
| --- | --- | --- |
| **Use case name :** | *Register* | |
| **Brief description:** | *Describes the process of registering a new user account in the system.* | |
| **Actors:** | *Customer, Sytem* | |
| **Stakeholders:** | *System administrator, User* | |
| **Preconditions:** | *The system is connected and working properly.*  *The user does not have an account in the system.* | |
| **Postconditions:** | *The new account is successfully created and stored in the system.*  *The user is successfully registered and can log into the system with a new account.*  *The system sends a registration confirmation email to the user to verify the account.*  *New account information is stored and managed in the system's database.*  *New registered accounts are subject to the rules and restrictions set forth by the system.* | |
| **Flow of activities:** | **Actor** | **Sytem** |
| *1. Users access the account registration page on the system's interface.*  *2. The user fills in the information in the registration form.* | *1. The system requires users to enter personal information, including name, email address, password and other information if necessary.*  *2.The system checks the validity of user information and saves the new account information in the database.*  *3.The system notifies the user that the account has been successfully registered.* |
| **Exception**  **Conditions:** | *If the user enters the wrong email address or password, the system will ask the user to re-enter the information.*  *If the email address is already used in the system, the system will notify the user and ask the user to re-enter another email address.* | |

*Login*

|  |  |  |
| --- | --- | --- |
| **Use case name :** | *Login* | |
| **Brief description:** | *Use case Login is used to allow users to log into the system with previously registered account information.* | |
| **Actors:** | *User, Sytem* | |
| **Stakeholders:** | *User, Sytem* | |
| **Preconditions:** | *The user already has an account registered on the system and needs to know the login information, including username and password.* | |
| **Postconditions:** | *If the credentials are valid, the user is taken to the main system screen.*  *If the credentials are not valid, the system asks the user to re-enter the credentials.* | |
| **Flow of activities:** | **Actor** | **Sytem** |
| *1.The user enters the username and password information in the login screen.* | *1. The system checks the user's login information against the database to confirm the validity of this information.*  *2. If the login information is valid, the system will bring the user to the main screen of the system.* |
| **Exception**  **Conditions:** | *If the login information is incorrect (wrong username or password), the system will display an error message and ask the user to re-enter it.* | |

*View order*

|  |  |  |
| --- | --- | --- |
| **Use case name :** | *View order* | |
| **Brief description:** | *Allows users to view the list of orders that have been placed on the system. The information related to the order includes order number, customer name, order date, product quantity, total order value.* | |
| **Actors:** | *Customer support staff, customer* | |
| **Stakeholders:** | *Customer support staff, customer* | |
| **Preconditions:** | *The user has logged into the system.*  *The user has access to the order view function.* | |
| **Postconditions:** | *The user can see the order information and related details.*  *The system stores information about whether the user has viewed the order.* | |
| **Flow of activities:** | **Actor** | **Sytem** |
| *1. The user selects the View Order function on the system interface.*  *2. User can search and filter order list.*  *3. When the user selects an order in the list, the system displays detailed information about that order.*  *4. Users can view detailed information about each product in the order by clicking the corresponding links on the interface.* | *1. The system displays a list of orders placed on the system.* |
| **Exception**  **Conditions:** | *If no orders are found on the system, the system will display a notification to the user.*  *If the user does not have permission to access the View Order function, the system displays an error message and does not allow access to this function.* | |

*Make order*

|  |  |  |
| --- | --- | --- |
| **Use case name :** | *Make order* | |
| **Brief description:** | *Customer wants to create a new order on the system.* | |
| **Actors:** | *Customer* | |
| **Stakeholders:** | *Customer, Sales agent* | |
| **Preconditions:** | *The customer has logged into the system.*  *The product you want to order is already on the system and is available for sale.* | |
| **Postconditions:** | *The system stores successful order information into the database.*  *The customer receives information about the order and the expected delivery time.*  *The salesperson receives the order information to make the delivery.* | |
| **Flow of activities:** | **Actor** | **Sytem** |
| *1. Customers access the product page on the system.*  *2. The customer searches for the product he wants to buy and adds it to the cart.*  *3. Customers choose to continue shopping if they want to buy more products, or choose to pay to proceed with the order.*  *4. The customer checks the order information and fills in the contact information to complete the order.*  *5. The customer chooses to continue to return to the product page or to complete to place an order.* | *1. The system displays order information including: product list, price, product quantity, total amount and customer information.*  *2. The system confirms the order and displays information about the order that has been placed successfully.* |
| **Exception**  **Conditions:** | *1.The product is no longer in sufficient quantity to fulfill the order.*  *2.Database connection error, cannot add orders to the system.*  *3.Error in order confirmation process* | |

*Pay by cash*

|  |  |  |
| --- | --- | --- |
| **Use case name :** | *Pay by cash* | |
| **Brief description:** | *This use case describes the process of paying for an order with cash at the point of sale.* | |
| **Actors:** | *Cashier* | |
| **Stakeholders:** | *Manage, User, Cashier* | |
| **Preconditions:** | *The customer has selected the items they wish to purchase.*  *The total cost of the items has been calculated.*  *The customer has chosen to pay with cash.* | |
| **Postconditions:** | *The payment has been recorded in the system.*  *The sale is completed and the customer receives their items.*  *The cash received from the customer has been deposited into the cash register.* | |
| **Flow of activities:** | **Actor** | **Sytem** |
| *1.The cashier selects the "Pay by cash" option in the point of sale system.*  *2.The customer hands over the cash payment to the cashier.*  *3.* *The cashier verifies that the amount received matches the total amount due.* | *1.The system displays the total amount due.*  *2.The system calculates the change due to the customer.*  *3.The cashier returns the change to the customer.*  *4.The system records the payment and finalizes the sale.* |
| **Exception**  **Conditions:** | *The payment amount exceeds the maximum amount allowed for cash transactions.*  *The cashier does not have enough cash on hand to provide change for the payment.*  *The payment is attempted after the store has closed for the day.* | |

*Online payment*

|  |  |  |
| --- | --- | --- |
| **Use case name :** | *Online payment* | |
| **Brief description:** | *Describe the online payment process when customers want to buy goods and use online payment methods.* | |
| **Actors:** | *User* | |
| **Stakeholders:** | *User, Sales agent* | |
| **Preconditions:** | *The customer has selected the product and placed an order on the website.*  *Customer has selected online payment method and provided full payment information* | |
| **Postconditions:** | *The system confirms the successful payment and provides information about the order to customers and sales staff.*  *Orders are stored in the system's database for tracking and management.* | |
| **Flow of activities:** | **Actor** | **Sytem** |
| *1.Customers choose the product they want to buy.*  *2.Customers log in or create an account if they do not have an account.*  *3.Customer enters shipping information and chooses online payment method.*  *4.Customer enters card information and confirms payment.*  *5.Customers receive order information and payment history.* | *4.1The system redirects the customer to the online payment page.*  *4.2The system processes the payment and notifies the customer of the payment result.* |
| **Exception**  **Conditions:** | *If the card information is invalid or insufficient, the system will notify the customer of the error.*  *If the online payment system has technical problems or errors* | |

*Make payment*

|  |  |  |
| --- | --- | --- |
| **Use case name :** | *Make payment* | |
| **Brief description:** | *where the user initiates the payment process by providing the requested payment details and payment confirmation.* | |
| **Actors:** | *Customer, Payment gateway system.* | |
| **Stakeholders:** |  | |
| **Preconditions:** | *The user must have selected the "Online payment" option for the purchase.*  *The user must have entered the valid payment details such as card number, expiration date, CVV, etc.*  *The user must have confirmed the payment.* | |
| **Postconditions:** | *The payment information is recorded in the system.*  *The user receives the payment confirmation message.* | |
| **Flow of activities:** | **Actor** | **Sytem** |
| *1.The user selects the "Online payment" option during checkout.*  *2.The user enters the valid payment details such as card number, expiration date, CVV, etc.*  *3.The user confirms the payment.* | *1.* *The system sends the payment information to the payment gateway for processing.*  *2.The bank authorizes or declines the payment and sends the response back to the payment gateway.*  *3.The payment gateway receives the authorization response from the bank and sends the response to the system.*  *4.The system displays the payment confirmation message to the user.* |
| **Exception**  **Conditions:** | *If the payment details entered by the user are invalid, the system displays an error message*  *If the payment gateway encounters any technical issues or errors during the payment processing.* | |

*Choose Address*

|  |  |  |
| --- | --- | --- |
| **Use case name :** | *Choose Address* | |
| **Brief description:** | *The use case allows users to choose a shipping address when placing an order on an e-commerce system.* | |
| **Actors:** | *User* | |
| **Stakeholders:** | *Customer, Sytem* | |
| **Preconditions:** | *User logged in to the e-commerce system.*  *User has placed an order and selected a shipping method.* | |
| **Postconditions:** | *The shipping address of the order has been updated successfully.* | |
| **Flow of activities:** | **Actor** | **Sytem** |
| *1.The user selects one of the saved addresses or adds a new one.*  *2.User enters new address information.* | *1.The system displays a list of the user's saved addresses.*  *2. system asks the user to enter new address information if the user chooses to add a new address.*  *3.The system saves the new address information in the database.*  *4.The system updates the shipping address for the order.* |
| **Exception**  **Conditions:** | *The user cannot connect to the database.*  *The user did not enter enough new address information.* | |

*Search by name*

|  |  |  |
| --- | --- | --- |
| **Use case name :** | *Search by name* | |
| **Brief description:** | *Product search function by name.* | |
| **Actors:** | *Customer, Sytem* | |
| **Stakeholders:** | *User* | |
| **Preconditions:** | *User has posted on the system.*  *The user wants to search for a product by name.* | |
| **Postconditions:** | *The system displays a list of products matching the search keyword.* | |
| **Flow of activities:** | **Actor** | **Sytem** |
| *1.The user selects the product search function by name.*  *2.The user enters the search keyword in the search box.* | *1.The system displays the product search page.*  *2.The system displays a list of products matching the search keyword.* |
| **Exception**  **Conditions:** | *If no product is found with a similar name to the search term, the system will notify the user.*  *If the database cannot be connected, the system will notify the user.* | |

*View Phones*

|  |  |  |
| --- | --- | --- |
| **Use case name :** | *View Phones* | |
| **Brief description:** | *User can see the list of mobile phones currently in the system.* | |
| **Actors:** | *User* | |
| **Stakeholders:** | *Users: want to search and view information about existing mobile phones in the system.*  *Admin: want to update and manage the list of existing mobile phones in the system.* | |
| **Preconditions:** | *Logged in to the system.* | |
| **Postconditions:** | *The user has seen the list of mobile phones.* | |
| **Flow of activities:** | **Actor** | **Sytem** |
| *1.The user accesses the system's website.*  *2.User logs in to the system.*  *3.The user selects the "View Phones" item from the menu.*  *4.User can see detailed information about each mobile in the list.* | *3.1.The system displays the list of mobile phones currently in the system to the user.* |
| **Exception**  **Conditions:** | *There are no mobile phones in the system. The system will display an error message and no list will be displayed.* | |

*Search by Manufacturer*

|  |  |  |
| --- | --- | --- |
| **Use case name :** | *Search by Manufacturer* | |
| **Brief description:** | *It allows customers to search for phone products from different manufacturers.* | |
| **Actors:** | *User* | |
| **Stakeholders:** | *Customer* | |
| **Preconditions:** | *The system displayed the home page with the product search option.*  *The customer has visited the product search page.* | |
| **Postconditions:** | *A list of phone products from the selected manufacturer is displayed.*  *Customers can continue to search or choose to buy products.* | |
| **Flow of activities:** | **Actor** | **Sytem** |
| *1.The customer visits the product search page.*  *2.Customers choose to search by manufacturer.*  *3.The customer chooses the manufacturer to look for.*  *4.Customers can continue to search or choose to buy products.* | *1.1.The system displays search options, including search by name, manufacturer, price, ...*  *2.1.The system displays a list of available manufacturers.*  *3.1.The system displays a list of products of the selected manufacturer.* |
| **Exception**  **Conditions:** | *If the product of the selected manufacturer is not found, the system will notify the customer and ask the customer to choose another manufacturer.*  *If the system encounters an error during the search.* | |

*Choose Reseller Order*

|  |  |  |
| --- | --- | --- |
| **Use case name :** | *Choose Reseller Order* | |
| **Brief description:** | *This use case allows the user to choose a reseller order from a list of available orders.* | |
| **Actors:** | *User* | |
| **Stakeholders:** | *Supplie, Warehouse staff, Sytem* | |
| **Preconditions:** | *The user must be logged in to the system.*  *There must be at least one reseller order available.* | |
| **Postconditions:** | *If the user chooses to add the reseller order to the cart, the system adds the selected order to the cart and updates the cart total.*  *If the user chooses to return to the list of available orders, the system displays the list of available orders again.* | |
| **Flow of activities:** | **Actor** | **Sytem** |
| *1.The user selects a reseller order from the list.*  *2.The user can choose to add the reseller order to the cart or return to the list of available orders.* | *1.The system displays a list of available reseller orders.*  *2.* *The system displays the details of the selected reseller order.* |
| **Exception**  **Conditions:** | *If there are no available reseller orders, the system displays a message indicating that there are no orders available.*  *If the user is not logged in to the system, the system displays a message.* | |

*Update Order Status*

|  |  |  |
| --- | --- | --- |
| **Use case name :** | *Update Order Status* | |
| **Brief description:** | *Describes the operations required to update the status of an order in the system.* | |
| **Actors:** | *Customer support staff, Store manager* | |
| **Stakeholders:** |  | |
| **Preconditions:** | *Order has been created in the system.*  *Permission is required for this feature.* | |
| **Postconditions:** | *The status of the order has been successfully updated in the database.* | |
| **Flow of activities:** | **Actor** | **Sytem** |
| *1.Customer support staff or store managers access the order status update feature.*  *2.The customer support staff or the store manager chooses a new status for the order.* | *1.The system displays a list of orders that have been created in the system.*  *2.The system displays detailed information of the order, including the current status of the order.*  *3.The system updates the new status for the order in the database.*  *4.The system notifies the customer of the new status of the order* |
| **Exception**  **Conditions:** | *If the order does not exist in the system.*  *If the order status cannot be updated due to a system error or a network error.* | |

*Update payment status*

|  |  |  |
| --- | --- | --- |
| **Use case name :** | *Update payment status* | |
| **Brief description:** | *This use case describes the steps involved in updating the payment status of an order.* | |
| **Actors:** | *Admin* | |
| **Stakeholders:** | *Customer, Administrator* | |
| **Preconditions:** | *The admin must be logged in to the system.*  *An order must exist in the system.*  *The payment for the order must have been made.* | |
| **Postconditions:** | *The payment status of the order is updated in the system.* | |
| **Flow of activities:** | **Actor** | **Sytem** |
| *1.The admin navigates to the "Orders" section of the system.*  *2.The admin selects the order for which the payment status needs to be updated.*  *3.The admin opens the order details and looks for the payment status field.*  *4.The admin saves the changes.* | *1.system retrieves the payment information associated with the order.*  *2.System verifies the payment status of the order.*  *3.System updates the payment status of the order based on the payment information.* |
| **Exception**  **Conditions:** | *If the admin is not logged in to the system, they cannot update the payment status of an order.*  *If the order does not exist in the system.*  *the payment for the order has not been made.* | |

*View Best-Selling product*

|  |  |  |
| --- | --- | --- |
| **Use case name :** | *View Best-Selling product* | |
| **Brief description:** | *View information about the best-selling products for a certain period of time, providing the user with a list of products sorted in descending order of the number of products sold.* | |
| **Actors:** | *Manager* | |
| **Stakeholders:** | *Customers, Marketing* | |
| **Preconditions:** | *The system has collected sufficient data on the sales and inventory of products.* | |
| **Postconditions:** | *The best-selling products are displayed to the manager.*  *The manager can use the information to make decisions on inventory management and product promotion.* | |
| **Flow of activities:** | **Actor** | **Sytem** |
| *1.Users access the sales management system.*  *2.The user chooses the function to view the best-selling products.*  *3.Users can choose to see details about each product in the list.* | *1.The system updates the list of best-selling products if there is a change.* |
| **Exception**  **Conditions:** | *Store Manager, Salesperson* | |

*View Monthly Income*

|  |  |  |
| --- | --- | --- |
| **Use case name :** | *View Monthly Income* | |
| **Brief description:** | *Allows the user to see the total sales of the store for each month.* | |
| **Actors:** | *Shop steward, Cashier Manager* | |
| **Stakeholders:** | *Store owner, Client, Staff* | |
| **Preconditions:** | *User has logged into the system.* | |
| **Postconditions:** | *Users can view the store's total sales for each month.* | |
| **Flow of activities:** | **Actor** | **Sytem** |
| *1.Users access the View Monthly Income feature on the system interface.*  *2.The user selects the month he wants to see the revenue.* | *1.2.* *The system displays the list of months and the total sales of the store in each month.*  *2.1.* *The system displays the total sales of the store in the selected month.* |
| **Exception**  **Conditions:** | *If there is not any revenue in the selected month, the system will display a message that there is no corresponding data.* | |

*View Product Data*

|  |  |  |
| --- | --- | --- |
| **Use case name :** | *View Product Data* | |
| **Brief description:** | *Allows users to view detailed product information, including price, description, inventory, customer ratings and comments.* | |
| **Actors:** | *Customer* | |
| **Stakeholders:** | *Client, Store owner* | |
| **Preconditions:** | *The user has logged into the system.* | |
| **Postconditions:** | *Users can view detailed product information.* | |
| **Flow of activities:** | **Actor** | **Sytem** |
| *1.The user visits the store's website.*  *2.Users searching for products need to see detailed information.* | *2.1.The system displays detailed product information, including price, description, inventory, customer ratings and comments.* |
| **Exception**  **Conditions:** | *If the product does not exist or is no longer in stock, the system will notify the user and not display the product information.*  *If the user is not logged in, the system will ask the user to log in to continue viewing product information.* | |

*Warehouse manager*

|  |  |  |
| --- | --- | --- |
| **Use case name :** | *Warehouse manager trong use case* | |
| **Brief description:** | *Responsible for managing inventory, making sure all items are properly stored and delivered to the correct locations. In the process of using the system.* | |
| **Actors:** | *Warehouse manager* | |
| **Stakeholders:** | *Warehouse manager, admin* | |
| **Preconditions:** | *The warehouse manager must be logged in to the system.*  *There must be inventory data available in the system.*  *There must be an existing order placed by a customer that needs to be fulfilled.* | |
| **Postconditions:** | *The inventory data will be updated based on the order placed by the customer.*  *The order status will be updated to "fulfilled".*  *The warehouse manager can view the updated inventory data.* | |
| **Flow of activities** | **Actor** | **Sytem** |
| *1.The warehouse manager selects an order to fulfill.*  *2.The warehouse manager verifies the inventory data to ensure that there is sufficient stock to fulfill the order.*  *3.* *The warehouse manager can view the updated inventory data to ensure that the stock has been updated correctly.* | *1.The system displays a list of pending orders that need to be fulfilled.*  *2.The system displays the details of the selected order, including the product name, quantity, and delivery address.* |
| **Exception**  **Conditions:** | *If there is not sufficient stock to fulfill the order, the warehouse manager must inform the customer about the delay and update the order status accordingly.*  *If the warehouse manager encounters any technical issues while updating the inventory data or the order status.* | |

*Product management*

|  |  |  |
| --- | --- | --- |
| **Use case name :** | *Product management* | |
| **Brief description:** | *Describe the activities related to product management in the system. This use case can include activities like adding a new product, editing product information, deleting a product, viewing a product listing, searching for a product, and more.* | |
| **Actors:** | *Admin, Product Manager.* | |
| **Stakeholders:** |  | |
| **Preconditions:** | *The user is logged in to the system and has access to the product management functionality.* | |
| **Postconditions:** | *New products have been added to the system.*  *Product information has been successfully updated.*  *The product has been removed from the system.* | |
| **Flow of activities** | **Actor** | **Sytem** |
| *1.User access to product management functionality.*  *2.Users choose to add new products, edit product information, delete products, view product listings, search for products, and more.*  *3.If the user chooses to delete the product, they will delete the product from the system.* | *1.The system displays a list of products currently in stock.*  *2.The system displays detailed product information, including name, description, price, quantity in stock, product image, and other specifications.*  *3.The system saves the changes and updates the product's information in the database.* |
| **Exception**  **Conditions:** | *If the user does not have access to the product management functionality, they will not be able to access this use case.*  *If the user does not enter all the required information when adding or editing a product, they will receive an error message.* | |

Storage management

|  |  |  |
| --- | --- | --- |
| **Use case name :** | *Storage management* | |
| **Brief description:** | *Describe the warehouse manager's inventory management process. When an order is received from a customer, the warehouse manager will confirm receipt of the order, make a list of necessary products to store, and arrange storage locations for each product.* | |
| **Actors:** | *Storage manager* | |
| **Stakeholders:** | *Warehouse manager,*  *Inventory manager,*  *IT department* | |
| **Preconditions:** | *Products that need to be stored in the warehouse have been ordered and confirmed by the customer.*  *The storage locations in the warehouse have been prepared and marked.* | |
| **Postconditions:** | *The products have been arranged to be stored in the warehouse.*  *Product information has been completely and accurately updated.* | |
| **Flow of activities** | **Actor** | **Sytem** |
| *1.The user accesses the warehouse management software.*  *2.User logs into his account to perform inventory management activities.*  *3.The user selects the repository to be managed and accesses it.*  *4.Users can search for information about goods, orders, import and export slips in the warehouse.*  *5.The user logs out of his account after completing the inventory management operations.* | *1.The system requires the user to define the warehouse to be managed.*  *2.The system updates information about the warehouse after each user performs a warehouse management operation.*  *3.The system allows users to add, edit and delete information about goods, orders, import and export slips in the warehouse.* |
| **Exception**  **Conditions:** | *The product cannot be added to the inventory because it is out of stock or no longer in production.*  *A product cannot be removed from the inventory because it has been sold and no longer exists in stock.* | |

*Manufacturing management*

|  |  |  |
| --- | --- | --- |
| **Use case name :** | *Manufacturing management* | |
| **Brief description:** | *Is a process that describes how the manufacturing process is managed in a system* | |
| **Actors:** | *Manufacturing Manager, Production Supervisor* | |
| **Stakeholders:** | *Manufacturing Manager, Production Supervisor* | |
| **Preconditions:** | *The manufacturing process is designed and set up in the system.*  *The necessary raw materials are available.* | |
| **Postconditions:** | *The manufacturing process is completed successfully.*  *The finished product is ready for distribution.* | |
| **Flow of activities** | **Actor** | **Sytem** |
| *1.Receive orders: The manufacturing manager receives orders for products from the sales department.*  *2.Review orders: The manufacturing manager reviews the orders to determine the quantity and type of products to be manufactured.*  *3.Monitor production: The manufacturing manager monitors the production process to ensure that it is running smoothly and on schedule.*  *4.* *Store finished products: The manufacturing manager stores finished products in the warehouse until they are ready to be shipped to customers.* | *1.Production planning: The system provides an interface for the user to plan production.*  *2.Place work orders: After planning, the system needs to generate work orders for the products listed in the plan.*  *3.Material warehouse management: The system needs to provide inventory management features, including inventory entry, exit and inventory update.* |
| **Exception**  **Conditions:** | *Insufficient raw materials are available, which may result in delays or cancellation of the production run.*  *Equipment failure or breakdown, which may result in delays or cancellation of the production run.* | |

*Reseller management*

|  |  |  |
| --- | --- | --- |
| **Use case name :** | *Reseller management* | |
| **Brief description:** | *Reseller management is a formulaic description of how to manage and interact with product distributors.* | |
| **Actors:** | *Administrator,*  *Reseller* | |
| **Stakeholders:** | *Producer, User,*  *Sales agent* | |
| **Preconditions:** | *The administrator has logged into the system*  *Registered Distributors and Admin Accepted*  *The information related to the distributors has been stored in the database of the system.* | |
| **Postconditions:** | *The information of distributors has been successfully updated or added to the system.* | |
| **Flow of activities** | **Actor** | **Sytem** |
| *1.The administrator logs into the system.*  *2.Admin accesses the distributor management page.*  *3.The administrator selects a distributor from the list to view details.*  *4.Administrators can add new or edit distributor information by selecting the corresponding options on the interface.*  *5.The administrator logs out of the system.* | *1.The system displays a list of distributors currently managed by the system.*  *2.The system displays the details of the distributor: Name and address, Phone number and email,...*  *3.The system stores the changes to the database and displays a successful update message.*  *4.The system requires confirmation from the administrator and deletes distributor-related information from the database.* |
| **Exception**  **Conditions:** | *If no distributor is found in the database, the system displays a message that no data was found.* | |

3.2.2. Textual Description for each use case:

Textual description for each use case is a written description that provides a detailed explanation of each step or action involved in a use case. It describes what the system does in response to a user's request, and how it interacts with the user and other entities involved in the use case.

The textual description typically includes a brief summary of the use case, the main actors involved, the preconditions and postconditions of the use case, and a step-by-step description of the activities involved.

Let's consider a use case for an online shopping system where a customer places an order:

Use Case: Place Order

Summary: The customer places an order for items they want to purchase.

Actors: Customer, Order Processing System, Payment System.

Preconditions: The customer has logged into their account and has added items to their cart.

Postconditions: The order is saved in the Order Processing System and the customer is redirected to a confirmation page.

Description:

1. The customer clicks on the "Checkout" button to place the order.

2. The system displays the customer's shipping and billing information.

3. The customer confirms the shipping and billing information and proceeds to the payment page.

4. The system redirects the customer to the Payment System.

5. The Payment System processes the payment and redirects the customer back to the Order Processing System.

6. The Order Processing System saves the order and generates an order confirmation number.

7. The system displays a confirmation page with the order confirmation number and the estimated delivery date.

8. If the payment fails, the Payment System sends an error message to the Order Processing System and displays an error message to the customer, asking them to try again or choose another payment method.

9. The Order Processing System receives the confirmation or error message from the Payment System and updates the order status accordingly.

10. The system redirects the customer to a confirmation page, showing the order number, status, and details.

11. The system sends an email confirmation to the customer with the same information as above.

**3.3 Rationale for your use case model:**

There are several reasons to use a use case model, including:

Directing the development process: Use case models help to direct the development process of a product or service by identifying the necessary functions and features required to meet user needs.

Optimizing product design: Using a use case model helps to define the features, usage scenarios, and interaction processes between users and the product, allowing for the optimization of product design to better meet user needs.

Minimizing risks and errors: Use case models help to detect and resolve risks and errors in the product or service development process. By identifying usage scenarios and interaction steps, developers can detect risks and errors earlier, minimizing costs associated with repairs and product improvements in the future.

Facilitating training and user support: Use case models help to effectively train and support users of a product or service. Usage scenarios and interaction processes are clearly defined, helping users better understand how to use the product or service and reducing user issues and concerns.

Increasing efficiency and productivity: Use case models help to increase the efficiency and productivity of the product or service development process, saving time and costs while ensuring quality and meeting user needs.

3.4 Non-functional requirements:

Performance: A web application should have a maximum response time of 2 seconds when a user visits a website and loads content on mobile devices. This is because users expect fast and smooth interactions with web applications, and they may abandon the site if it takes too long to load . To achieve this performance goal, a web application should use various techniques such as caching, compression, minification, and optimization of images, scripts, and stylesheets .

Security: A customer management system (CRM) must be secured using SSL encryption to protect customer data during remote access. SSL encryption ensures that the data transmitted between the client and the server is encrypted and authenticated, preventing unauthorized access, modification, or interception.

Scalability: An application with scalability in mind is crucial for handling increasing website traffic or data. The use of cloud computing and web services technologies can aid in scalability by providing on-demand resources and modular components that can be distributed across various servers or platforms. Scalability ensures that the application can handle higher loads without affecting performance or functionality.

Compatibility: For a mobile app to be successful, it needs to function well on both iOS and Android operating systems. This means that the app should run smoothly and without errors or crashes on a variety of devices and platforms. Achieving this requires adhering to the standards and guidelines of each operating system, utilizing responsive design methods to adapt to different screen sizes and orientations, and thoroughly testing the app on a range of devices and emulators.

Reliability: To ensure that a network system is reliable, it must be able to recover from faults or errors and continue to function correctly even after a node failure or disconnection. Redundancy techniques, such as replication, backup, or failover, should be used to provide an alternative node that can take over in case of a failure and ensure that the network system can self-heal.

Availability: Ensuring high availability is crucial for a web application, which means it should be accessible and responsive at all times, regardless of the number of concurrent users or any other external factors. Load balancing techniques can be used to distribute traffic evenly among multiple servers or clusters to prevent a single point of failure. Additionally, monitoring tools should be in place to keep track of the performance and health of the servers, with alerts set up to notify administrators of any issues.

Manageability: In order to improve the warehouse management process and support management decisions, the warehouse management system should be capable of monitoring and reporting on stock inventory, sales ratio, and order information. Additionally, the system should be easily controlled, configured, updated, and maintained by administrators or users. To achieve this, a warehouse management system should feature a user-friendly interface for easy data access and manipulation, a database system that securely and efficiently stores and organizes data, and a reporting tool that provides valuable insights and statistics from the data.

**4. Architecture**

4.1. Architectural style(s) used:

Layered architecture: This style divides the system into layers, with each layer responsible for a specific aspect of the system.

There are several reasons why Layered architecture:

+Modularity: Layered architecture promotes modularity by dividing the system into layers, with each layer responsible for a specific aspect of the system. This makes it easier to manage and maintain the system, as changes can be made to one layer without affecting the other layers.

+Separation of concerns: Layered architecture promotes separation of concerns by separating the system into layers, with each layer responsible for a specific concern. This helps to reduce complexity and makes the system easier to understand and modify.

+Encapsulation: Each layer in Layered architecture encapsulates its implementation details, which provides a level of abstraction that allows other layers to interact with it without having to know its internal workings. This improves the system's overall maintainability and reduces dependencies between layers.

+Scalability: Layered architecture is highly scalable, as new layers can be added or existing layers can be modified without affecting the other layers. This allows the system to grow and evolve over time as requirements change.

+Standardization: Layered architecture promotes standardization by separating the system into well-defined layers, each with its own set of responsibilities and interfaces. This helps to ensure that the system is consistent and maintainable over time.

4.2. Architectural model:

The Three-Tier Architecture is a widely used software architecture that separates applications into three distinct computing tiers: the presentation tier, the application tier, and the data tier. The presentation tier is responsible for handling user interaction and communication, while the application tier performs data processing using a set of predefined business rules. Finally, the data tier is where information is stored and managed by the application.

The Three-Tier Architecture provides several benefits such as:

+ Scalability: Each layer can be scaled independently, allowing for more efficient resource utilization.

+ Modularity: Each layer can be developed and tested independently, which reduces the risk of errors and facilitates maintenance.

+ Security: The Three-Tier Architecture provides a clear separation between user interface and data storage layers, which can enhance security by preventing unauthorized access to sensitive data.

The Three-Tier Architecture is the most popular implementation of a multi-tier architecture and can be used for large applications on the web or for serverless.

4.3. Technology, software, and hardware used:

4.3.1 Technology

- Internet: Using the Internet to connect users' computers and devices to the website.

- SSL encryption technology: Using SSL encryption technology to ensure the security and safety of customers' personal information when making purchases on the website.

- C# programming language: Use the C# programming language to develop web applications on the website.

4.3.2. Software

- Content Management System (CMS): Using CMS software to manage and update website content, including products and other information about the website.

- Database management system: Using database management system to store and manage information about products, orders and customers.

- C# Framework: Use the C# framework to develop web applications on the website.

### 4.3.3. Hardware:

- Computer: Use a computer to develop and manage the website.

- Communication devices: Use communication devices to connect the website to the Internet and other users' devices.

Developing an online phone selling website with C# offers numerous advantages, including cross-platform compatibility, simplified maintenance and development, and robust support for website data processing and management.

4.4. Rationale for your architectural style and model:

Layered architecture divides the application into multiple layers, where each layer has its own specific responsibility and communicates only with the layers directly above and below it. It also helps in ensuring that changes made to one layer do not affect other layers, thus reducing the risk of unintended side effects.

In a layered architecture, the application is divided into horizontal layers that depend on each other from top to bottom. Each layer provides services to the layer above it and consumes services from the layer below it.

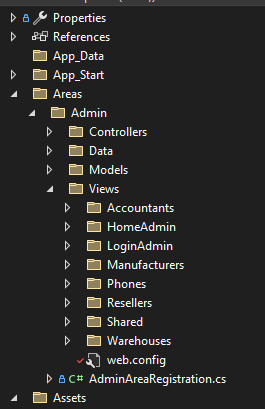
In a three-tier architecture, the application is divided into three vertical tiers that communicate with each other through well-defined interfaces. Each tier can be deployed independently on different machines or servers, which improves scalability and performance. The three tiers are: client, server, and database. The client tier is responsible for presenting the user interface and interacting with the user. The server tier contains the application logic and processes the requests from the client tier. The database tier stores and manages the data for the application.

Layered architecture is a logical way of organizing the code into different layers, such as the presentation layer, business layer, and data layer. Each layer has its specific responsibility and communicates only with the layers directly above and below. This helps in ensuring that changes made to one layer do not affect other layers, thus reducing the risk of unintended side effects.

Three-tier architecture is a physical way of deploying the code into different tiers, such as the presentation tier, application tier, and data tier. The presentation tier deals with user interface components, the application tier handles the application logic, and the data tier is responsible for handling data storage and retrieval. Each tier can run on a separate machine or process, allowing for better performance and security. This also enables developers to make changes to one tier without affecting the other tiers, allowing for easier updates and upgrades.

Both layered and three-tier architectures provide benefits for software development. Layered architecture is suitable for applications with complex business logic, while three-tier architecture is more appropriate for applications with a focus on data storage and retrieval, such as database-driven applications. However, they are not mutually exclusive. A three-tier application can also follow a layered design within each tier, or a layered application can be deployed across multiple tiers. The choice of architecture depends on the requirements and constraints of each project.

5. Source Code:



**Login Admin:**

@model FinalProject.Areas.Admin.Models.LoginViewModelAdmin

@{

ViewBag.Title = "Login\_Admin";

}

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

<meta name="description" content="">

<meta name="author" content="">

<title>Login Admin</title>

<!-- Custom fonts for this template-->

<link href="~/Areas/Admin/Data/vendor/fontawesome-free/css/all.min.css" rel="stylesheet" type="text/css">

<link href="https://fonts.googleapis.com/css?family=Nunito:200,200i,300,300i,400,400i,600,600i,700,700i,800,800i,900,900i"

rel="stylesheet">

<!-- Custom styles for this template-->

<link href="~/Areas/Admin/Data/css/sb-admin-2.min.css" rel="stylesheet">

</head>

@using (Html.BeginForm("Index", "LoginAdmin", FormMethod.Post, new { id = "loginForm" }))

{

@Html.AntiForgeryToken()

<body class="bg-gradient-primary">

<div class="container">

<!-- Outer Row -->

<div class="row justify-content-center">

<div class="col-xl-10 col-lg-12 col-md-9">

<div class="card o-hidden border-0 shadow-lg my-5">

<div class="card-body p-0">

<div class="row">

<div class="col-lg-12">

<div class="p-5">

<div class="text-center">

<h1 class="h4 text-gray-900 mb-4">Login Admin</h1>

</div>

<form class="user">

<div class="form-group">

@Html.LabelFor(model => model.Email)

@Html.TextBoxFor(model => model.Email, new { @class = "form-control" })

@Html.ValidationMessageFor(model => model.Email)

</div>

<div class="form-group">

@Html.LabelFor(model => model.Password)

@Html.PasswordFor(model => model.Password, new { @class = "form-control" })

@Html.ValidationMessageFor(model => model.Password)

</div>

@if (TempData["Error"] != null)

{

<div class="alert alert-danger">@TempData["Error"]</div>

}

<div class="form-group">

<div class="custom-control custom-checkbox small">

<input type="checkbox" class="custom-control-input" id="customCheck">

<label class="custom-control-label" for="customCheck">

Remember

Me

</label>

</div>

</div>

<button type="submit" class="btn btn-primary btn-user btn-block">Login</button>

</form>

</div>

</div>

</div>

</div>

</div>

</div>

</div>

</div>

<!-- Bootstrap core JavaScript-->

<script src="~/Areas/Admin/Data/vendor/jquery/jquery.min.js"></script>

<script src="~/Areas/Admin/Data/vendor/bootstrap/js/bootstrap.bundle.min.js"></script>

<!-- Core plugin JavaScript-->

<script src="~/Areas/Admin/Data/vendor/jquery-easing/jquery.easing.min.js"></script>

<!-- Custom scripts for all pages-->

<script src="~/Areas/Admin/Data/js/sb-admin-2.min.js"></script>

</body>

}

</html>

@{

TempData.Remove("Error");

}

@section scripts {

<script>

window.addEventListener('beforeunload', function () {

var form = document.getElementById('loginForm');

form.reset();

});

</script>

}

**Home Admin:**

@{

ViewBag.Title = "Index";

Layout = "~/Areas/Admin/Views/Shared/\_LayoutAdmin.cshtml";

}

<div class="form-group">

<div class="col-md-offset-2 col-md-10 text-center">

<a href="@Url.Action("Index","Accountants")" class="btn btn-outline-primary">Accountants Management</a>

<a href="@Url.Action("Index","Manufacturers")" class="btn btn-outline-primary">Manufacturers Management</a>

<a href="@Url.Action("Index","Phones")" class="btn btn-outline-primary">Phones Management</a>

<a href="@Url.Action("Index","Resellers")" class="btn btn-outline-primary">Resellers Management</a>

<a href="@Url.Action("Index","Warehouses")" class="btn btn-outline-primary">Warehouses Management</a>

</div>

</div>

**LayoutAdmin:**

@{

Layout = null;

}

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<link rel="preconnect" href="https://fonts.gstatic.com/">

<link rel="canonical" href="~/Areas/Admin/Views/HomeAdmin/Index.cshtml" />

<title>Admin Page</title>

<link href="~/Areas/Admin/Data/vendor/fontawesome-free/css/all.min.css" rel="stylesheet" type="text/css">

<link href="~/Areas/Admin/Data/css/sb-admin-2.min.css" rel="stylesheet" type="text/css">

<!-- END SETTINGS -->

</head>

<body data-theme="default" data-layout="fluid" data-sidebar-position="left" data-sidebar-layout="default">

<div id="wrapper">

<!-- Sidebar -->

<ul class="navbar-nav bg-gradient-primary sidebar sidebar-dark accordion" id="accordionSidebar">

<!-- Sidebar - Brand -->

<a class="sidebar-brand d-flex align-items-center justify-content-center" href="~/Admin/HomeAdmin">

<div class="sidebar-brand-text mx-3">BK's Admin</div>

</a>

<!-- Divider -->

<hr class="sidebar-divider my-0">

<!-- Nav Item - Dashboard -->

<li class="nav-item">

<a class="nav-link" href="@Url.Action("Index", "Accountants")">

<i class="fas fa-money-bill"></i>

<span>Accountant</span>

</a>

</li>

<!-- Divider -->

<hr class="sidebar-divider">

<li class="nav-item">

<a class="nav-link" href="@Url.Action("Index", "Warehouses")">

<i class="fas fa-solid fa-warehouse"></i>

<span>Warehouse</span>

</a>

</li>

<hr class="sidebar-divider">

<li class="nav-item">

<a class="nav-link" href="@Url.Action("Index", "Phones")">

<i class="fas fa-solid fa-mobile"></i>

<span>Phone</span>

</a>

</li>

<hr class="sidebar-divider">

<li class="nav-item">

<a class="nav-link" href="@Url.Action("Index", "Resellers")">

<i class="fas fa-solid fa-store"></i>

<span>Reseller</span>

</a>

</li>

<hr class="sidebar-divider">

<!-- Nav Item - Charts -->

<li class="nav-item">

<a class="nav-link" href="@Url.Action("Index", "Manufacturers")">

<i class="fas fa-solid fa-truck"></i>

<span>Manufacturer</span>

</a>

</li>

<hr class="sidebar-divider">

<!-- Nav Item - Tables -->

<li class="nav-item">

<a class="nav-link" href="@Url.Action("Index", "Receipts")">

<i class="fas fa-solid fa-receipt"></i>

<span>Reiceipts</span>

</a>

</li>

<!-- Divider -->

<hr class="sidebar-divider d-none d-md-block">

<!-- Sidebar Toggler (Sidebar) -->

<div class="text-center d-none d-md-inline">

<button class="rounded-circle border-0" id="sidebarToggle"></button>

</div>

</ul>

<!-- End of Sidebar -->

<!-- Content Wrapper -->

<div id="content-wrapper" class="d-flex flex-column">

<!-- Main Content -->

<div id="content">

<!-- Topbar -->

<nav class="navbar navbar-expand navbar-light bg-white topbar mb-4 static-top shadow">

<!-- Sidebar Toggle (Topbar) -->

<button id="sidebarToggleTop" class="btn btn-link d-md-none rounded-circle mr-3">

<i class="fa fa-bars"></i>

</button>

<!-- Topbar Navbar -->

<ul class="navbar-nav ml-auto">

<!-- Nav Item - Alerts -->

<li class="nav-item dropdown no-arrow mx-1">

<a class="nav-link dropdown-toggle" href="#" id="alertsDropdown" role="button"

data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">

<i class="fas fa-solid fa-bell"></i>

<!-- Counter - Alerts -->

<span class="badge badge-danger badge-counter">3+</span>

</a>

<!-- Dropdown - Alerts -->

<div class="dropdown-list dropdown-menu dropdown-menu-right shadow animated--grow-in"

aria-labelledby="alertsDropdown">

<h6 class="dropdown-header">

Alerts Center

</h6>

<a class="dropdown-item d-flex align-items-center" href="#">

<div class="mr-3">

<div class="icon-circle bg-primary">

<i class="fas fa-file-alt text-white"></i>

</div>

</div>

<div>

<div class="small text-gray-500">December 12, 2019</div>

<span class="font-weight-bold">A new monthly report is ready to download!</span>

</div>

</a>

<a class="dropdown-item d-flex align-items-center" href="#">

<div class="mr-3">

<div class="icon-circle bg-success">

<i class="fas fa-donate text-white"></i>

</div>

</div>

<div>

<div class="small text-gray-500">December 7, 2019</div>

$290.29 has been deposited into your account!

</div>

</a>

<a class="dropdown-item d-flex align-items-center" href="#">

<div class="mr-3">

<div class="icon-circle bg-warning">

<i class="fas fa-exclamation-triangle text-white"></i>

</div>

</div>

<div>

<div class="small text-gray-500">December 2, 2019</div>

Spending Alert: We've noticed unusually high spending for your account.

</div>

</a>

<a class="dropdown-item text-center small text-gray-500" href="#">Show All Alerts</a>

</div>

</li>

<!-- Nav Item - Messages -->

<li class="nav-item dropdown no-arrow mx-1">

<a class="nav-link dropdown-toggle" href="#" id="messagesDropdown" role="button"

data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">

<i class="fas fa-solid fa-envelope"></i>

<!-- Counter - Messages -->

<span class="badge badge-danger badge-counter">7</span>

</a>

<!-- Dropdown - Messages -->

<div class="dropdown-list dropdown-menu dropdown-menu-right shadow animated--grow-in"

aria-labelledby="messagesDropdown">

<h6 class="dropdown-header">

Message Center

</h6>

<a class="dropdown-item d-flex align-items-center" href="#">

<div class="dropdown-list-image mr-3">

<img class="rounded-circle" src="img/undraw\_profile\_1.svg"

alt="...">

<div class="status-indicator bg-success"></div>

</div>

<div class="font-weight-bold">

<div class="text-truncate">

Hi there! I am wondering if you can help me with a

problem I've been having.

</div>

<div class="small text-gray-500">Emily Fowler · 58m</div>

</div>

</a>

<a class="dropdown-item d-flex align-items-center" href="#">

<div class="dropdown-list-image mr-3">

<img class="rounded-circle" src="img/undraw\_profile\_2.svg"

alt="...">

<div class="status-indicator"></div>

</div>

<div>

<div class="text-truncate">

I have the photos that you ordered last month, how

would you like them sent to you?

</div>

<div class="small text-gray-500">Jae Chun · 1d</div>

</div>

</a>

<a class="dropdown-item d-flex align-items-center" href="#">

<div class="dropdown-list-image mr-3">

<img class="rounded-circle" src="img/undraw\_profile\_3.svg"

alt="...">

<div class="status-indicator bg-warning"></div>

</div>

<div>

<div class="text-truncate">

Last month's report looks great, I am very happy with

the progress so far, keep up the good work!

</div>

<div class="small text-gray-500">Morgan Alvarez · 2d</div>

</div>

</a>

<a class="dropdown-item d-flex align-items-center" href="#">

<div class="dropdown-list-image mr-3">

<img class="rounded-circle" src="https://source.unsplash.com/Mv9hjnEUHR4/60x60"

alt="...">

<div class="status-indicator bg-success"></div>

</div>

<div>

<div class="text-truncate">

Am I a good boy? The reason I ask is because someone

told me that people say this to all dogs, even if they aren't good...

</div>

<div class="small text-gray-500">Chicken the Dog · 2w</div>

</div>

</a>

<a class="dropdown-item text-center small text-gray-500" href="#">Read More Messages</a>

</div>

</li>

<div class="topbar-divider d-none d-sm-block"></div>

<!-- Nav Item - User Information -->

<li class="nav-item dropdown no-arrow">

<a class="nav-link dropdown-toggle" href="#" id="userDropdown" role="button"

data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">

<img class="img-profile rounded-circle"

src="~/Areas/Admin/Data/img/avatars/null\_profile.png">

</a>

<!-- Dropdown - User Information -->

<div class="dropdown-menu dropdown-menu-right shadow animated--grow-in"

aria-labelledby="userDropdown">

<a class="dropdown-item" href="#">

<i class="fas fa-user fa-sm fa-fw mr-2 text-gray-400"></i>

Profile

</a>

<a class="dropdown-item" href="#">

<i class="fas fa-cogs fa-sm fa-fw mr-2 text-gray-400"></i>

Settings

</a>

<div class="dropdown-divider"></div>

<a class="dropdown-item" href="#" data-toggle="modal" data-target="#logoutModal">

<i class="fas fa-sign-out-alt fa-sm fa-fw mr-2 text-gray-400"></i>

Logout

</a>

</div>

</li>

</ul>

</nav>

<!-- End of Topbar -->

<main class="content">

<div class="container-fluid p-8">

<div>

@RenderBody()

</div>

</div>

</main>

<!-- Footer -->

<footer class="sticky-footer bg-white">

<div class="container my-auto">

<div class="copyright text-center my-auto">

<span>Copyright &copy; Your Website 2021</span>

</div>

</div>

</footer>

<!-- End of Footer -->

</div>

<!-- End of Content Wrapper -->

</div>

</div>

@Scripts.Render("~/bundles/jquery")

@Scripts.Render("~/bundles/bootstrap")

@RenderSection("scripts", required: false)

<div class="modal fade" id="logoutModal" tabindex="-1" role="dialog" aria-labelledby="exampleModalLabel"

aria-hidden="true">

<div class="modal-dialog" role="document">

<div class="modal-content">

<div class="modal-header">

<h5 class="modal-title" id="exampleModalLabel">Ready to Leave?</h5>

<button class="close" type="button" data-dismiss="modal" aria-label="Close">

<span aria-hidden="true"></span>

</button>

</div>

<div class="modal-body">Select "Logout" below if you are ready to end your current session.</div>

<div class="modal-footer">

<button class="btn btn-secondary" type="button" data-dismiss="modal">Cancel</button>

<a class="btn btn-primary" href="@Url.Action("Logout", "LoginAdmin")">Logout</a>

</div>

</div>

</div>

````</div>

<!-- Bootstrap core JavaScript-->

<script src="~/Areas/Admin/Data/vendor/jquery/jquery.min.js"></script>

<script src="~/Areas/Admin/Data/vendor/bootstrap/js/bootstrap.bundle.min.js"></script>

<!-- Core plugin JavaScript-->

<script src="~/Areas/Admin/Data/vendor/jquery-easing/jquery.easing.min.js"></script>

<!-- Custom scripts for all pages-->

<script src="~/Areas/Admin/Data/js/sb-admin-2.min.js"></script>

<!-- Page level plugins -->

<script src="~/Areas/Admin/Data/vendor/chart.js/Chart.min.js"></script>

</body>

</html>

**Login Controllers:**

using FinalProject.Areas.Admin.Models;

using FinalProject.Models;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.Mvc;

namespace FinalProject.Areas.Admin.Controllers

{

public class LoginAdminController : Controller

{

private FinalDatabaseEntities db = new FinalDatabaseEntities();

// GET: Admin/Login

public ActionResult Index()

{

return View();

}

[HttpPost]

public ActionResult Index(LoginViewModelAdmin model)

{

if (ModelState.IsValid)

{

var Admin = db.Admins.FirstOrDefault(u => u.AdminEmail == model.Email && u.AdminPassword == model.Password);

if (Admin != null)

{

// Đăng nhập thành công, lưu thông tin người dùng vào session và chuyển hướng đến trang quản trị

Session["AdminUser"] = Admin;

return RedirectToAction("Index", "HomeAdmin");

}

else

{

// Đăng nhập không thành công, đặt thông báo lỗi vào TempData và chuyển hướng về trang login view

TempData["Error"] = "Email hoặc mật khẩu không đúng.";

return RedirectToAction("Index", "LoginAdmin");

}

}

return View(model);

}

// GET: Admin/Logout

public ActionResult Logout()

{

Session.Abandon();

return RedirectToAction("Index", "LoginAdmin");

}

}

}

**REFERENCES**

**English:**

1. Agile vs. Waterfall. (n.d.). Sunbytes. Retrieved from <https://sunbytes.io/blog/agile-vs-waterfall/>
2. Hardware & Software Requirements. (n.d.). Mt. Hood Community College. Retrieved from <https://www.mhcc.edu/OLHardwareSoftwareRequirements/>
3. Schedule of Deliverables. (n.d.). Law Insider. Retrieved from <https://www.lawinsider.com/dictionary/schedule-of-deliverables#:~:text=Schedule%20of%20Deliverables%20means%20the,for%20each%20Milestone%20and%20Deliverable>.
4. Know Your Transactions: KYT Explained. (n.d.). Sumsub. Retrieved from (<https://sumsub.com/kyt/?utm_source=google&utm_medium=cpc&utm_campaign=sem_g_srch_kyt_general_ww&utm_content=pid|kwd-317068797101|cid|19651752622|aid|647248346772|gid|151474673928|pos||src|g_|dvc|c|reg|1028581|rin||fid|&utm_term=transactional%20monitoring&gclid=CjwKCAjw3ueiBhBmEiwA4BhspGIQita1pEOiMJsI5yBOs5rMQmCEyi1b9_4s4lK7cQZ0cSqkZbY0pxoCi6YQAvD_BwE>)
5. Professional Standards for Educational Leaders. (2015). National Policy Board for Educational Administration. Retrieved from <https://www.npbea.org/wp-content/uploads/2017/06/Professional-Standards-for-Educational-Leaders_2015.pdf>
6. Configuration Management Plan. (n.d.). National Archives and Records Administration. Retrieved from <https://www.archives.gov/files/era/recompete/config-mgmt-plan.pdf>
7. Requirements vs Specifications: Create a Shared Vocabulary. (2021, March 30). Argon Digital. Retrieved from <https://argondigital.com/blog/product-management/requirements-vs-specifications-create-a-shared-vocabulary/#:~:text=A%20requirement%20specification%20is%20a,and%20maintenance%20of%20the%20product>.
8. Non-Functional Requirements: What They Are, Why They Matter, How to Write Them. (n.d.). AltexSoft. Retrieved from <https://www.altexsoft.com/blog/non-functional-requirements/#:~:text=Non%2Dfunctional%20requirements%20or%20NFRs,reliability%2C%20data%20integrity%2C%20etc>.
9. The Most Recognised Architectural Styles. (n.d.). Barker Associates. Retrieved from <https://www.barker-associates.co.uk/service/architecture/most-recognised-architectural-styles/>
10. IT (information technology). (n.d.). TechTarget. Retrieved from <https://www.techtarget.com/searchdatacenter/definition/IT>
11. A Model to Represent Architectural Design Rationale. (2003). ResearchGate. Retrieved from <https://www.researchgate.net/publication/224606007_A_Model_to_Represent_Architectural_Design_Rationale>