MID-TERM ASSIGNMENT REPORT

CBAD2103
SYSTEM ANALYSIS AND DESIGN

JAN 2025

HUTECH

STUDENT'S NAME: **TA DUY THANH TAI**

STUDENT ID: **2254030103**

CLASS: **22BOIT02**

EMAIL: **thanhtai12122004@gmail.com**

INSTRUCTOR: **Mr. Le Dang Khoa**

Contents

1. Introduction

In today's digital landscape, mobile shopping systems play a crucial role in enhancing business operations and consumer convenience. This report presents the analysis and design of a Mobile Shopping System for a retail company. The system is aimed at providing an intuitive shopping experience for consumers while ensuring effective inventory and order management for administrators. The analysis covers the system's objectives, feasibility, design methodology, and essential components, including diagrams representing the system's architecture.

With the rapid advancement of mobile technology, consumers now expect seamless and efficient shopping experiences right at their fingertips. Mobile shopping systems are transforming the way businesses operate by enabling real-time interactions and personalized experiences. This report delves into the comprehensive analysis and design of a mobile shopping system tailored to meet the demands of both consumers and administrators, thereby driving business growth and customer satisfaction.

The introduction of mobile shopping systems has revolutionized the retail industry by bridging the gap between physical and digital shopping experiences. These systems empower consumers with the ability to explore a wide range of products, make informed purchasing decisions, and receive personalized recommendations. For businesses, mobile shopping systems provide valuable insights into consumer behavior, enabling data-driven decision-making and targeted marketing strategies. By leveraging the power of mobile technology, businesses can enhance their operational efficiency, expand their market reach, and foster customer loyalty.

In this report, we will explore the key components of a mobile shopping system, starting with the system objectives, scope, description, and benefits. We will then discuss the chosen system development methodology and its advantages over other methodologies. The feasibility analysis will evaluate the operational, technical, and financial aspects of the system. We will also delve into the methods used to determine system requirements, including stakeholder interviews, surveys, market research, and use case analysis. The report will include detailed diagrams, such

as the context diagram and data flow diagram, to illustrate the system's architecture and data flow. Finally, we will describe the functional and non-functional requirements of the system, followed by the system architecture design and conclusion.

2. Objectives, Scope , Description and Benefits of the System

2.1 Objectives

The main objective of this system is to develop a feature-rich mobile shopping platform that enhances the e-commerce experience for consumers and optimizes retail business operations.

For consumers:

- smooth browsing, shopping and order tracking experience .
- secure transactions and multiple payment methods.
- Make personalized shopping recommendations.

For administrators:

- Simplify product and inventory management.
- Automate order processing and sales tracking .
- Enhance decisions through sales data analysis.

The system aims to bridge the gap between consumers and businesses by providing a user-friendly platform that is both easy to use and fully featured. By integrating advanced technologies such as AI-based recommendations and real-time analytics, the system ensures a competitive edge in the digital marketplace.

To achieve these goals, the system will focus on providing a user-friendly interface that allows consumers to easily browse products, add to cart, and complete purchases. The platform will support multiple payment options, including credit cards, e-wallets, and bank transfers, ensuring a smooth and secure transaction process . Additionally, the system will provide personalized recommendations based on users' browsing history and preferences, enhancing the overall shopping experience.

For administrators, the system will provide a comprehensive dashboard to manage product listings, inventory levels, and pricing. The platform will automate order processing, reducing manual intervention and minimizing errors. Sales data analytics will allow administrators to monitor performance, identify trends, and make data-driven decisions to optimize operations and improve profitability.

By meeting the needs of both consumers and administrators, the system aims to create a smooth and efficient shopping experience, promoting customer satisfaction and business growth.

2.2 Scope

The system is designed to support mobile platforms (iOS and Android) and web platforms.

Features for consumers:

- User registration and authentication.
- Search products, filter and manage shopping cart.
- Secure payments and multiple payment options.
- Order tracking and customer support.

Features for administrators:

- List products, control inventory and manage pricing.
- Order processing, shipping and customer data handling.
- Sales and performance analysis reporting tool.

The system scope includes a wide range of functions to provide a comprehensive solution for both consumers and administrators. By supporting diverse platforms, the system ensures accessibility and convenience, meeting the diverse preferences of users.

Consumer features are designed to make shopping easy and enjoyable. Users will be able to create an account, log in securely , and manage their personal profile. Product search and filtering will help users quickly find items they are interested in, while cart management will allow them to add or remove products and view total costs. The secure checkout process will support multiple payment methods, ensuring a smooth and hassle-free transaction. Users will also be able to track their orders and access customer support for any queries or issues.

For administrators, the system will provide powerful tools for managing product listings, including adding, updating, and deleting items. Inventory control features will help administrators monitor inventory levels and set reorder points to avoid stockouts. Pricing management tools will allow administrators to set and adjust prices, apply discounts, and manage promotions. Order processing functionality will automate tasks such as order confirmation, packaging, and shipping, ensuring timely and accurate fulfillment. Customer data processing features will help administrators manage user information and resolve customer requests. Reporting tools will provide insights into sales performance, inventory levels, and customer behavior , enabling data-driven decisions.

2.3 Description

This mobile shopping system integrates user-friendly interfaces with backend functionalities that support order fulfillment, payment processing, and real-time analytics. The system connects with external services such as logistics partners and payment gateways to ensure smooth operations.

The user interface is intuitively designed, making it easy for consumers to navigate through products, make purchases, and track their orders. A robust backend infrastructure ensures that administrators can efficiently manage inventory, process orders, and analyze sales data. Integration with external services adds a layer of convenience, allowing for seamless transactions and deliveries.

The system will be developed using modern technologies to ensure flexibility, security and performance. The front-end user interface will be built using React Native for mobile applications and React.js for the web platform, providing a consistent and responsive user experience across devices. The backend will be run on Node.js, a high-performance, flexible and event-driven server environment. MongoDB, a NoSQL database, will be used to store user data, product information, and order details. The system will be hosted on a secure cloud infrastructure , ensuring availability and reliability.

secure and convenient transactions , while logistics partners will ensure timely and accurate deliveries. Real-time analytics will be supported by tools like Google Analytics and Elasticsearch, providing administrators with valuable insights into user behavior and system performance.

API integration will allow the system to connect and communicate with external services seamlessly, ensuring that data is exchanged securely and efficiently. Using a NoSQL database like MongoDB provides great flexibility in managing complex data structures and large volumes of data.

3. System Development Methodology

3.1 Chosen Methodology: Agile Development

Reason:

- **Flexibility: Allows for** phased development and continuous user feedback.
- **Faster Deployment:** Releases are done in small chunks, ensuring quick adaptation to market trends.
- **Risk Reduction:** Early detection of problems helps reduce development risks.
- **User-centric:** Regular testing and feedback improves user satisfaction.

Agile development method is particularly suitable for mobile commerce systems due to its flexibility and focus on user engagement. By breaking the development process into short phases, Agile ensures that the system remains responsive to changing requirements and market dynamics.

Agile development involves iterative cycles called sprints, which typically last two to four weeks. Each sprint focuses on delivering a specific set of features or improvements, allowing the development team to gather feedback and make adjustments in real time. This approach fosters collaboration between developers, designers, and stakeholders, ensuring that the final product aligns with user expectations and business goals.

One of the main advantages of Agile development is the ability to respond quickly to changes and continuous improvements. This helps the system stay current and relevant to new consumer needs and market changes.

3.2 Comparison with Other Methods

Method	Strengths	Weaknesses
Waterfall	Clear structure, easy to take notes	Rigid and difficult to adapt to change
RAD	Fast development, user engagement	Requires highly skilled developers
Spiral	Minimize risk, improve every stage	High cost and complexity

The Agile methodology was chosen because it balances flexibility, efficiency, and user engagement, making it ideal for mobile commerce systems. The Agile methodology ’s iterative approach allows for continuous improvement and refinement, ensuring that the final product meets user expectations and business goals.

4. Feasibility Analysis

4.1 Operational Feasibility

- User-friendly interface ensures easy application.
- Training materials are provided to assist users.

The operational feasibility of the system is assessed based on ease of use and availability of training materials. Intuitive user interface design minimizes learning curve for consumers and administrators, promoting widespread adoption. Comprehensive training materials and support ensure that users can quickly become proficient with the system.

Providing detailed documentation and online training courses will help users quickly become familiar with and use the system effectively. The system will provide resources such as video tutorials, PDF documents and online courses, helping users understand the features and processes of the system.

4.2 Technical Feasibility

- Developed with React Native, Node.js and MongoDB.
- secure cloud infrastructure ensuring scalability and security.

The technical feasibility of the system is assessed based on the selected technologies and infrastructure. React Native enables the development of cross-platform mobile applications, while Node.js and MongoDB provide a robust backend framework. Hosting the system on a secure cloud infrastructure ensures scalability, reliability, and security, meeting industry standards and best practices.

Using these technologies offers many benefits, including high performance, flexible scalability, and superior security. Storing the system on the cloud also allows optimizing operating costs and ensuring high availability, helping the system always operate stably and serve users continuously.

4.3 Financial Feasibility

- Estimated cost: $50,000–$70,000.
- Revenue model : commission fees, advertising and premium services.

The financial viability of the system is determined by evaluating the estimated costs and potential revenue streams . The initial investment is estimated to be between $50,000 and $70,000, which covers development, infrastructure, and marketing costs. The system’s revenue

model includes transaction commissions, targeted advertising, and premium services, ensuring a sustainable and profitable business model.

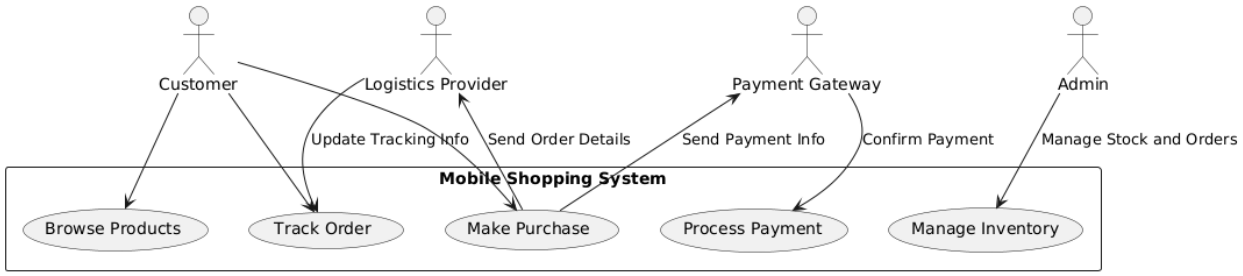
diversified revenue model helps the system have many stable sources of income, thereby ensuring sustainable financial capacity and supporting future system development and upgrades. Applying effective marketing strategies and optimizing operating costs will help achieve financial goals and bring economic benefits to the business.

5. Method of Determining System Requirements

- **Stakeholder Interviews:** Engage potential users to gain insights.
- **Surveys & Questionnaires:** Collect customer preferences and challenges.
- **Market Research:** Analyze competitor strategies and industry trends.
- **Use case analysis:** Map out the user's interactions with the system.

System requirements definition involves a combination of qualitative and quantitative methods to gather comprehensive information. Stakeholder interviews provide valuable insights from potential users, while surveys and questionnaires collect customer preferences and pain points. Market research helps identify industry trends and competitor strategies, ensuring the system stays competitive. Use case analysis maps user interactions, helping guide the design of user-friendly and functional features.

6. Context Diagram



Context Diagram

A Context Diagram is a high-level, simplified view of the system that shows its boundaries and interactions with external entities. It provides an overview of how the system interacts with users, external services, and other systems. Below is a detailed description of the Context Diagram for the Mobile Shopping System:

External Entities:

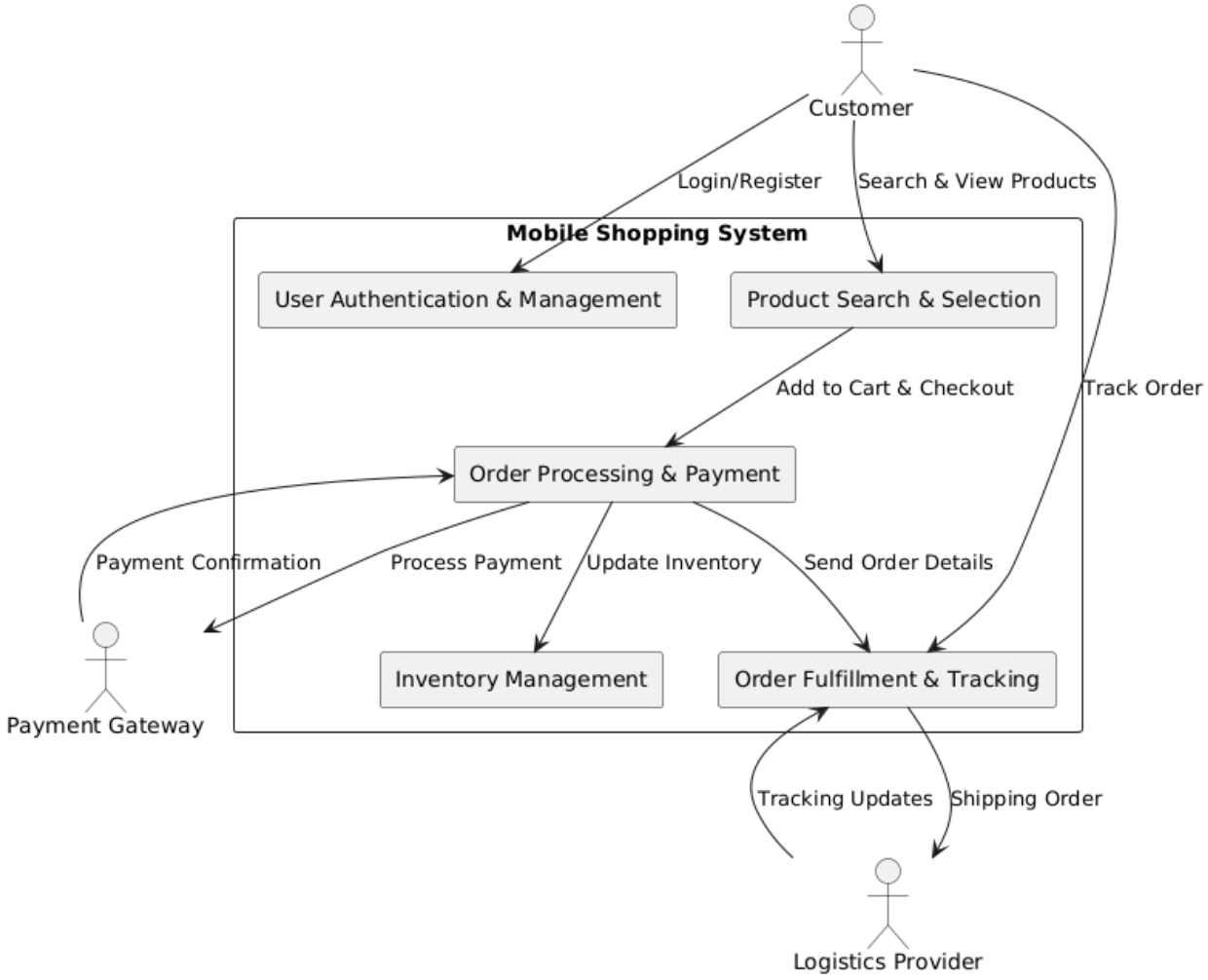
1. **Customers**: Individuals who use the mobile shopping system to browse products, make purchases, and track their orders.
2. **Administrators**: Users who manage the product listings, inventory, and order processing within the system.
3. **Payment Gateway**: External service providers (e.g., Stripe, PayPal) that handle payment processing for transactions made within the system.
4. **Logistics Providers**: External services responsible for delivering the products to customers. They provide tracking information and update the system on the delivery status.
5. **Customer Support**: External or internal service that assists customers with inquiries, complaints, and other support-related issues.

Interactions:

- **Customers** interact with the system to register, log in, browse products, add items to the cart, make purchases, and track orders.
- **Administrators** interact with the system to add, update, and manage product listings, monitor inventory levels, process orders, and generate reports.
- **Payment Gateway** interacts with the system to process payments securely and provide confirmation of successful transactions.
- **Logistics Providers** interact with the system to receive order details, update delivery status, and provide tracking information.
- **Customer Support** interacts with the system to receive customer inquiries, resolve issues, and provide support.

The Context Diagram visually represents these interactions, highlighting the flow of information between the system and external entities.

7. Data Flow Diagram (Level 0)



The Level 0 Data Flow Diagram (DFD) provides a high-level overview of the Mobile Shopping System, showing the main processes and how data flows between them and external entities.

Processes:

1. **User Authentication & Management**: Handles user registration, login, and management of user accounts.
2. **Product Search & Selection**: Allows customers to search for and select products from the catalog.
3. **Order Processing & Payment**: Manages the shopping cart, checkout process, and payment transactions.
4. **Inventory Management**: Tracks product stock levels and updates inventory based on sales and restocking.
5. **Order Fulfillment & Tracking**: Manages order fulfillment, shipping, and tracking.

External Entities:

- **Customer**: Interacts with the system to browse products, make purchases, and track orders.
- **Payment Gateway**: Processes payment transactions and provides payment confirmation.
- **Logistics Provider**: Handles the shipping and delivery of orders and provides tracking updates.

Data Flows:

- **Login/Register**: Customers provide login credentials or registration details, which are processed by the User Authentication & Management system.
- **Search & View Products**: Customers search for and view product information, which is managed by the Product Search & Selection system.
- **Add to Cart & Checkout**: Customers add products to their cart and proceed to checkout, initiating the Order Processing & Payment process.
- **Process Payment**: Payment details are sent to the Payment Gateway, which processes the payment and sends confirmation back to the system.
- **Update Inventory**: The Inventory Management system updates stock levels based on sales and restocking information.
- **Send Order Details**: Order details are sent to the Order Fulfillment & Tracking system for processing.
- **Shipping Order**: The Logistics Provider receives shipping orders and updates tracking information, which is sent back to the system.
- **Track Order**: Customers track their orders through the Order Fulfillment & Tracking system.

Modified Level 0 Data Flow Diagram (DFD)

Below is the modified Level 0 DFD based on the provided description:

1. **User Authentication & Management**
    - Input: Login/Register (from Customer)
    - Output: User Details (to Customer), Authentication Status (to Product Search & Selection, Order Processing & Payment)
2. **Product Search & Selection**
    - Input: Search Request (from Customer), Authentication Status (from User Authentication & Management)
    - Output: Product List, Product Details (to Customer)
3. **Order Processing & Payment**
    - Input: Add to Cart & Checkout (from Customer), Authentication Status (from User Authentication & Management)
    - Output: Order Details (to Inventory Management, Order Fulfillment & Tracking), Payment Request (to Payment Gateway)
4. **Inventory Management**

- Input: Order Details (from Order Processing & Payment), Restocking Info (from Admin)
- Output: Updated Inventory Levels (to Order Processing & Payment, Product Search & Selection)

5. **Order Fulfillment & Tracking**
    - Input: Order Details (from Order Processing & Payment), Shipping Order (from Logistics Provider)
    - Output: Order Status, Tracking Info (to Customer), Shipping Request (to Logistics Provider)
6. **Payment Gateway**
    - Input: Payment Request (from Order Processing & Payment)
    - Output: Payment Confirmation (to Order Processing & Payment)
7. **Logistics Provider**
    - Input: Shipping Request (from Order Fulfillment & Tracking)
    - Output: Tracking Updates (to Order Fulfillment & Tracking)

8. Functional and Non-Functional Requirements

8.1 Functional Requirements

1. Securely authenticate and manage user accounts .
2. Search, filter and purchase product functions.
3. Order processing, tracking and payment integration.
4. Customer support and feedback management.

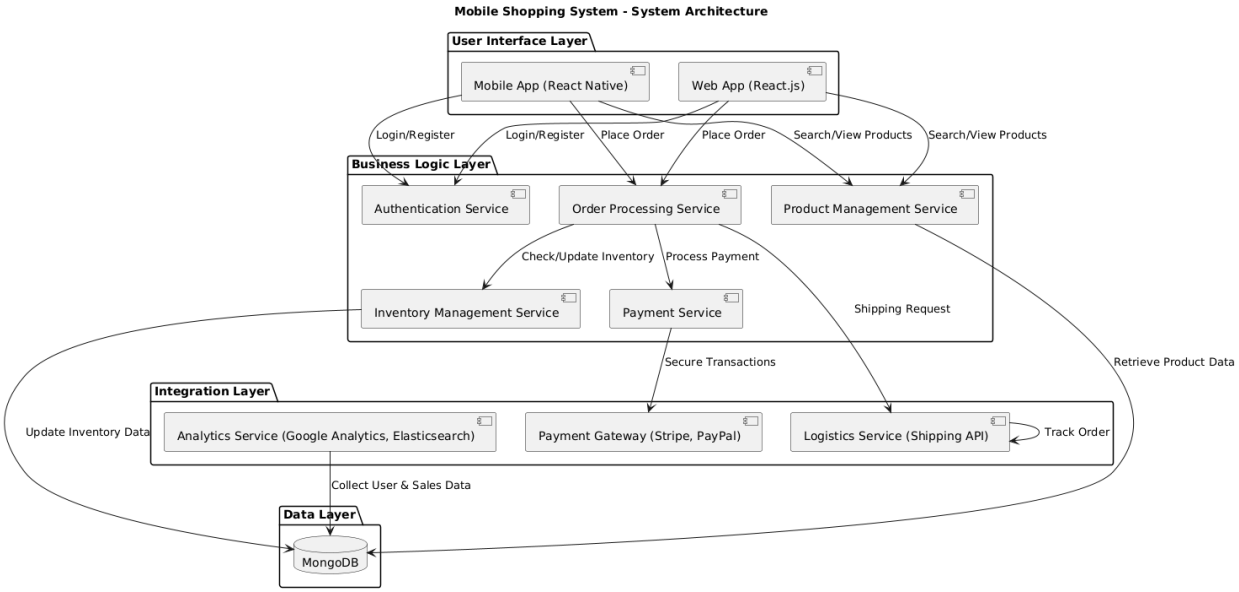
Functional requirements define the core features and capabilities of the system. These requirements ensure that the system meets the basic needs of consumers and administrators, providing a safe and efficient shopping experience.

8.2 Non-Functional Requirements

1. The system must support 10,000 concurrent users.
2. Must comply with PCI-DSS security standards.
3. Fast response time, under 2 seconds per transaction.

Non-functional requirements address the performance, security, and scalability aspects of the system. By setting high standards for concurrent user support, security compliance, and response time, the system ensures a reliable and secure user experience .

9. System Architecture Design



Description:

1. **User Interface Layer**:
    - Includes a **Mobile App (React Native)** and a **Web App (React.js)**.
    - Users interact with the system through these interfaces to browse products, place orders, and track shipments.
2. **Business Logic Layer**:
    - Manages core functionalities such as **authentication**, **product management**, **order processing**, **inventory management**, and **payments**.
3. **Data Layer**:
    - Uses **MongoDB** for storing user data, product information, and order details.
4. **Integration Layer**:
    - Connects with external services such as:
        - **Payment Gateway (Stripe, PayPal)** for secure transactions.
        - **Logistics Service (Shipping API)** for handling deliveries and tracking.
        - **Analytics Service (Google Analytics, Elasticsearch)** for gathering user insights.

System architecture design outlines the structural components and their interactions. The user interface layer provides the front-end experience for consumers and administrators, while the business logic layer handles core functionality. The data layer ensures secure storage and retrieval of information , and the integration layer supports communication with external services.

10. Conclusion

The development of the Mobile Shopping System is a significant step forward in enhancing the e-commerce experience for both consumers and administrators. This system is designed to address the evolving needs of the digital marketplace by offering a robust, user-friendly, and secure platform that streamlines the entire shopping process.

One of the key benefits of the Mobile Shopping System is its ability to provide a seamless and intuitive shopping experience for consumers. The system's user interface is designed to be easy to navigate, allowing users to effortlessly browse products, add items to their cart, and complete purchases. The integration of AI-driven recommendations enhances the shopping experience by providing personalized product suggestions based on users' browsing history and preferences. This not only makes the shopping process more enjoyable for consumers but also increases the likelihood of repeat purchases and customer loyalty.

For administrators, the system offers a comprehensive suite of tools for managing product listings, inventory, and orders. The automation of routine tasks such as order processing and inventory management reduces the manual workload, allowing administrators to focus on more strategic activities. The system's reporting tools provide valuable insights into sales performance, inventory levels, and customer behavior, enabling data-driven decision-making that can optimize operations and improve profitability. By streamlining these processes, the system helps businesses operate more efficiently and effectively in the competitive e-commerce landscape.

The adoption of Agile development methodology in the creation of the Mobile Shopping System ensures that the system is both flexible and adaptive to changing market needs. Agile's iterative approach allows for continuous improvement and refinement, incorporating user feedback at every stage of development. This user-centric approach ensures that the final product aligns with the expectations and needs of its users, both consumers and administrators. By delivering regular updates and enhancements, the system remains relevant and competitive in the fast-paced digital marketplace.

In terms of technical feasibility, the system leverages modern technologies such as React Native, Node.js, and MongoDB to ensure scalability, security, and performance. These technologies provide a robust foundation for the system, allowing it to handle high volumes of transactions and user interactions seamlessly. The secure cloud infrastructure further enhances the system's reliability and availability, ensuring that users can access the platform at any time without disruptions. The integration with external services such as payment gateways and logistics providers ensures that transactions and deliveries are handled efficiently and securely.

Financially, the Mobile Shopping System is designed to be a sustainable and profitable business model. The estimated development costs are balanced by the potential revenue streams from commission fees, targeted advertisements, and premium services. By opening up new sales channels and expanding the customer base, the system provides businesses with opportunities for increased revenue and growth. The data-driven insights generated by the system enable businesses to optimize their sales and marketing strategies, further enhancing their profitability and competitive edge.

Operationally, the system's user-friendly interface and comprehensive training materials ensure that both consumers and administrators can quickly and easily adopt the platform. The intuitive design minimizes the learning curve, while the training resources provide support for users to become proficient with the system's features. This ease of adoption promotes widespread use of the system, driving higher engagement and satisfaction among users.

In conclusion, the Mobile Shopping System represents a significant advancement in the e-commerce landscape. By addressing the needs of both consumers and administrators, the system creates a holistic and efficient shopping experience that drives customer satisfaction and business growth. The use of Agile development methodology, modern technologies, and a user-centric approach ensures that the system remains flexible, scalable, and responsive to market trends. The financial and operational feasibility of the system further reinforces its potential for long-term success and sustainability. As businesses continue to navigate the digital marketplace, the Mobile Shopping System offers a powerful tool for enhancing their operations, expanding their reach, and achieving their strategic objectives.

11. References

Domestic References (Vietnamese Sources):

- Denning, S. (2024). *Effective work management using the Agile methodology* (Lê Hồng Phương Hạ, Trans.). National Economics University Publishing House.
- Nguyễn, H. (2022). *Agile Y: Software development process*. Self-published.
- Base.vn. (n.d.). *What is Agile? Everything you need to know about the Agile model in project management*. https://base.vn/blog/agile-la-gi/

International References:

1. Mobile Application Development Technologies (React Native, Node.js, MongoDB):

- Bush, E. (2018). *Node.js, MongoDB, React, React Native: Full-stack fundamentals and beyond*. Blue Sky Productions Inc.
- Nabendu, B. (2020). *React Native Project with NodeJS and MongoDB - Part 1*. Medium. https://nabendu82.medium.com/react-native-project-with-nodejs-and-mongodb-part-1-443cc5b65b46
- MongoDB. (n.d.). *React Native*. MongoDB Developer Center. https://www.mongodb.com/developer/technologies/react-native/

2. Agile Software Development Methodology:

- Ambler, S. W., & Holitza, M. (n.d.). *Agile for dummies*. IBM Limited Edition.
- TechTarget. (2024). *What is Agile software development?*. https://www.techtarget.com/searchsoftwarequality/definition/agile-software-development
- Atlassian. (n.d.). *What is Agile?*. https://www.atlassian.com/agile

3. Online Shopping Systems and User Experience:

- Freniche, D. (2023, June 9). *Build an offline-first React Native mobile app with Expo and Realm React Native*. MongoDB Developer Center. https://www.mongodb.com/developer/technologies/react-native/
- Nabendu, B. (2020). *React Native Project with NodeJS and MongoDB - Part 3*. Medium. https://nabendu82.medium.com/react-native-project-with-nodejs-and-mongodb-part-3-6a8128d08b0e