SLP (Scan Link Pay)

by

Jose Carlos El Khoury

Submitted to the

School of Engineering

In partial fulfillment of the requirement

For the degree of

Bachelor of Engineering –

Computer Engineering

at the

Holy Spirit University of Kaslik (USEK)

Kaslik, Lebanon

SLP (Scan Link Pay). A Seamless Payment Solution for Large-Scale Businesses and Malls.

by

Jose Carlos El Khoury

APPROVAL:

Dr. Ayman Khalil

Advisor / Supervisor

Lecturer [Signature]

Eng. Jean Chamoun

Internal/External Examiner

Lecturer [Signature]

Dr. Issam Moussallem

Internal/External Examiner

Lecturer [Signature]

Date defended:
[December 18, 2023]

Plagiarism Statement

I confirm that this final year project is my own work, is not copied from any other person's work (published or unpublished) and was not previously submitted for assessment either at the Holy Spirit University of Kaslik (USEK) or elsewhere.

I confirm that I have read and understood the Academic Integrity regulations on plagiarism in the *Academic Rules and Student Life Handbook*.

December 5, 2023

Signature(s) Date

Holy Spirit University of Kaslik (USEK)
Library

Certificate:
Plagiarism Check with Turnitin

LEVEL: Final Year Project

Thesis title:

SLP (Scan Link Pay). A Seamless Payment Solution for Large-Scale Businesses and Malls.

Student name: Jose Carlos El Khoury

Turnitin Paper ID No: 2245246889

Last Date of check: 02/12/23, 13:56

Supervisor's declaration:

This paper has been evaluated using Turnitin:
I have thoroughly analyzed the report produced by the system for unoriginal work and plagiarism.
I certify that the references in this paper are in accordance with good academic practice.
I approve and accept the percentage stated in the Originality Report herein attached.

Supervisor name: Ayman Khalil

Supervisor title: Dr.

Signature:

Date:

Final Year Project
Release Form

I, the undersigned, hereby submit this final year project to the Holy Spirit University of Kaslik (USEK)

as partial fulfillment of the requirements for a Bachelor of Engineering's degree.

By signing and submitting this agreement: I grant USEK Library the right to

(a) reproduce electronic copies of my final year project for the purpose of preservation

(b) include the scholarly material final year project in the archives and digital repositories at the Holy Spirit University of Kaslik "USEK Digital Gate"

(c) release my final year project or dissertation to ProQuest/UMI

(d) keep more than one copy of my final year project or dissertation for purposes of security and backup

(e) reproduce, publicly display, and distribute the material to users world-wide at no cost for academic purposes

I should inform the Main Library at the University if I intend to publish or post my final year project before the transaction is completed and listing the University as my affiliation.

I request an embargo of this final year project for 24 months from the date of submission of the final year project.

USEK will clearly identify your name as the author or owner of the final year project, and will not make any alteration, other than as allowed by this license, to your submission.

December 5, 2023

Signature Date

ACKNOWLEDGEMENT

First, I want to express my sincere appreciation and gratitude to Holy Spirit University of Kaslik for providing me with the necessary resources to complete this final year project report.

I also want to extend my deepest thanks to my final year project supervisor at USEK, Dr. Ayman KHALIL, for his invaluable guidance, unwavering support, and assistance in preparing this report.

Furthermore, I would like to acknowledge the assistance provided by all the professors who helped me during my time at USEK, particularly Prof. Joseph ZALAKET for his exceptional support.

Lastly, I want to express my heartfelt appreciation to my family and colleagues for their unwavering support and motivation throughout my life. Their encouragement was crucial in making this project a reality.

ABSTRACT

The increasing demand for efficient and contactless payment methods has led to the development of innovative solutions that cater to the needs of both customers and businesses. SLP is a cutting-edge web application designed to streamline the payment process for large-scale businesses and locations, such as cinemas, by integrating a third-party payment solution. The web app not only allows customers to make swift and secure transactions using QR codes, which significantly reduces waiting times and queues and enhances the overall customer experience, but it also provides large businesses with valuable statistics and data about their operations.

Keywords:
Contactless payment methods, Web application, Large-scale businesses, Third-party payment solution, Waiting times, Customer experience, Statistics and data.

RÉSUMÉ

La demande croissante pour des méthodes de paiement efficaces et sans contact a conduit au développement de solutions innovantes répondant aux besoins des clients et des entreprises. SLP est une application web de pointe conçue pour simplifier le processus de paiement pour les entreprises et les lieux de grande envergure, tels que les cinémas, en intégrant une solution de paiement tiers. L'application web permet non seulement aux clients d'effectuer des transactions rapides et sécurisées à l'aide de codes QR, ce qui réduit considérablement les temps d'attente et les files d'attente et améliore l'expérience client globale, mais elle fournit également aux grandes entreprises des statistiques et des données précieuses sur leurs opérations.

Mots-clés:
Méthodes de paiement sans contact, Application web, Entreprises de grande envergure, Solution de paiement tiers, Temps d'attente, Expérience client, et Statistiques et données.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMES

- SLP: Scan Link Pay.
- QR: Quick Response.
- JSON: JavaScript Object Notation.
- REST: Representational State Transfer.
- API: Application Programming Interface.
- XML: Extensible Markup Language.
- MVC: Model View Controller.
- HTTP: Hypertext Transfer Protocol.
- DB: Database.
- ACID: Atomicity, Consistency, Isolation, and Durability.
- POM: Project Object Model.
- DSL: Domain-Specific Language.
- ISO: International Organization for Standardization.
- IEC: International Electrotechnical Commission.
- ISMS: Information Security Management System.

I. INTRODUCTION

The rapid evolution of digital technology and the rise of e-commerce have led to an increased demand for more efficient and user-friendly payment methods. Traditional payment systems often result in long waiting times and queues, creating a suboptimal experience for customers and negatively impacting businesses. To address this issue, there is a need for research into innovative solutions that can streamline the payment process and significantly enhance user satisfaction. This report focuses on the research, development and implementation of SLP, a web application designed to provide seamless payment solutions for large-scale businesses and locations, such as malls and cinemas.

The primary motivation behind this project is to create a comprehensive and user-centric payment system that not only reduces waiting times and queues but also ensures secure and contactless transactions. SLP aims to revolutionize the way customers interact with businesses by offering a highly efficient and versatile platform that caters to a wide range of services and products.

The main objectives of this research and implementation are as follows:

1. To analyze the current state of payment systems and identify the challenges faced by large-scale businesses and customers.
2. To propose an innovative solution using existing technologies that addresses the identified challenges and enhances the overall user experience.
3. To design and develop a web application using Java with Spring Boot, MySQL, and JSON technologies to implement the back-end of the proposed solution.
4. To evaluate the performance and usability of the developed application through a series of tests after implementation.

Throughout this report, detailed information will be provided on the research methodologies employed, the design and structure of the back-end and the final results. Additionally, diagrams will be used to illustrate the application's architecture and data flow, offering a clear visual representation of the project's components and their interactions.

In addition to addressing the challenges faced by customers and businesses in the current payment landscape, this report delves into the practical aspects of implementing and testing the SLP system. It demonstrates the step-by-step process of bringing the SLP concept into reality, covering the nuances of development strategies, the tools employed, and the rigorous testing protocols undertaken to ensure reliability and efficiency.

The SLP system is envisioned not just as a technological solution but as a transformative approach with the potential to reshape the payment industry dynamics, enhancing the experience for both businesses and their customers.

II. STATE OF THE ART AND RELATED WORK

1. State of the Art:

In recent years, numerous solutions have emerged in the digital payment landscape, leveraging advanced technologies to streamline transactions and improve customer experiences. This section will explore the current state of the art in payment systems, highlighting notable innovations and the challenges that remain unaddressed.

a. Digital Payment Apps:

Digital payment apps, such as Apple Pay, Google Pay, and PayPal, have revolutionized the payment industry by enabling users to make contactless payments using their smartphones. These solutions typically involve linking a user's bank account or credit card to the app, allowing them to make payments at participating businesses. While these apps have significantly enhanced the payment experience for users, they primarily focus on individual transactions rather than addressing the challenges faced by large-scale businesses and venues.

b. QR Code-based Payment Systems:

QR code-based payment systems, such as Alipay and WeChat Pay, have gained popularity due to their ease of use and versatility. Users can simply scan a QR code displayed by the business to complete a transaction quickly and securely. However, these systems often require users to download a separate app, also it is tailored to single businesses, which may not be feasible for temporary visitors or tourists in large venues such as malls or cinemas.

c. Integrated Point of Sale (POS) Systems:

Integrated POS systems, such as Square and Clover, combine hardware and software to process transactions, manage inventory, and generate sales reports. These systems provide businesses with a comprehensive solution to manage their sales and streamline the payment process. However, they may not be suitable for large-scale businesses and venues, as they can be expensive to implement and maintain.

d. Cryptocurrency and Blockchain-based Payment Solutions:

Cryptocurrencies have introduced decentralized payment systems based on blockchain technology. These systems offer secure and transparent transactions, with lower fees

compared to traditional payment methods. However, the volatile nature of cryptocurrencies and the lack of widespread adoption make them less practical for everyday use in large-scale businesses and venues.

Despite the advancements in digital payment solutions, there is still a need for a comprehensive and user-centric payment system that caters to the unique challenges faced by large-scale businesses and malls. SLP aims to address these challenges by offering an innovative platform that combines the best aspects of existing technologies to create a seamless payment experience for both customers and businesses.

2. Related Work:

In the rapidly evolving landscape of digital payment solutions, it is crucial to understand the existing technologies and applications that have shaped the market. The "Related Work" section of this report aims to explore and analyze the work of competitors and similar solutions in the domain of QR-based payment systems.

So, let's explore some key differences between SLP and the competitors:

a. AirPay:

AirPay is a mobile payment solution that can be used to pay for goods and services. Users can link their credit card to make purchases. However, AirPay is not specifically designed for businesses with multiple stores or locations. Additionally, while AirPay offers some features as in SLP, such as the ability to purchase digital vouchers, it doesn’t offer the same level of functionality.

b. Boost Mobile Wallet:

Boost Mobile Wallet is a mobile payment solution that users also (as AirPay) pay for goods by linking their credit card. However, like AirPay, Boost Mobile Wallet is not specifically designed for businesses with multiple stores or locations. Also, Boost Mobile Wallet lacks certain features that will be used in SLP such as the ability to buy items and services from multiple stores at once or to exchange funds with other users.

c. **LevelUp:**

LevelUp is a mobile payment platform that is primarily designed for restaurants. The app allows users to order, make payments, and earn rewards. However, LevelUp does not offer the same level of functionality as SLP as it is specifically designed for restaurants.

d. **Qkr!:**

Qkr! is a mobile payment app (developed by Mastercard), which as LevelUp, designed specifically for restaurants and has the same features (view menus, place orders, make payments and split bills with users). However, like all the competitors listed above, Qkr! is not specifically designed for businesses with multiple stores or locations.

e. **ABC Mall App (Specific to Lebanon):**

The ABC App represents an online mall shopping solution in Lebanon with delivery services. However, it is only limited to basic online shopping with delivery services.

The following comparison table highlights the key features of SLP and its competitors, emphasizing the unique functionalities that differentiate SLP from AirPay, Boost Mobile Wallet, LevelUp, Qkr!, and ABC:

Table 1. Comparison of SLP and Competitors

Feature	SLP	AirPay	Boost Mobile Wallet	LevelUp	Qkr!	ABC
QR code for item details	Yes	No	No	No	No	No
Exchange funds with users	Yes	No	Yes	No	No	No
Split payments	Yes	No	No	Yes	Yes	No
Transaction history	Yes	Yes	Yes	Yes	Yes	Yes
Save payment information	Yes	Yes	Yes	Yes	Yes	Yes
Search for items/services	Yes	No	No	No	No	Yes
Add items to favorites	Yes	No	No	No	No	No
Designed for multi-store payments*	Yes	No	No	No	No	Yes (Delivery Only)

Note:

Designed for multi-store payments: A feature that allows users to make purchases and payments across multiple merchants within a designated location, offering a unified and streamlined payment experience.

Overall, while the competitors we've discussed and SLP have similarities, SLP is more generalized and is tailored for bigger businesses with multiple stores or locations. Additionally, it offers a wider range of features such as the ability to redeem paid items/services, to exchange funds with other users, and to buy items and services from multiple stores at once.

III. SYSTEM REQUIREMENTS AND SPECIFICATIONS

The SLP web app aims to provide a seamless and efficient payment experience for customers and businesses in various settings such as cinemas, malls, and other large venues. To fully realize the potential of the SLP system, a comprehensive set of system requirements and specifications has been meticulously defined. The requirements encompass a range of factors, from user interaction and transaction processing to security protocols and system adaptability. By adhering to these specifications, SLP aspires to deliver an unparalleled payment experience, characterized by its user-friendliness, and robust security measures.

1. Comprehensive and User-centric Approach

- **Introduction to the Challenge:** In the rapidly evolving commercial landscape, large-scale businesses such as malls face significant challenges in managing payment processes. These challenges include long waiting times for customers, inefficient transaction processing, and the pressing need for effective customer service in high-traffic commercial areas. Traditional payment methods often fall short in addressing these demands, leading to a suboptimal experience for both businesses and their patrons.
- **SLP as a Solution:** The 'SLP (Scan Link Pay)' system emerges as a tailored solution to these prevalent issues. It leverages advanced digital technology to significantly enhance the payment process. Central to SLP's approach are QR code scanning and contactless transactions, technologies renowned for their efficiency and positive impact on user experience. This innovative combination is especially beneficial in environments where quick and secure transactions are paramount.
- **User-Centric Design Philosophy:** At the core of the SLP system is a user-centric design philosophy. Every aspect, from the ease of account creation to the efficiency of transaction processing, is crafted with the customer's experience in mind. SLP aims to markedly reduce waiting times and streamline payments, thereby elevating customer satisfaction and potentially boosting business efficiency.
- **Combining Existing Technologies Innovatively:** SLP stands out by integrating the best aspects of existing payment technologies into one cohesive platform. Beyond basic payment processing, the system's novel use of QR codes extends to functionalities like detailed item information access, integration with existing

payment solutions, and payment across multiple merchants, adding layers of utility and engagement for users.

- **Ensuring Security and Efficiency:** A paramount concern in the design of SLP is the security of user data and transactional integrity. The system incorporates robust security measures to safeguard sensitive information. Simultaneously, SLP maintains a high level of transactional efficiency, crucial for user acceptance and operational effectiveness in busy commercial settings.
- **Adaptability to Various Settings:** Finally, the adaptability of the SLP system to diverse commercial environments is a key strength. Whether it is in malls, cinemas, or other large venues, SLP's versatility makes it an ideal solution for a variety of business models and customer interaction scenarios. This adaptability ensures that SLP is not just a payment solution but a comprehensive, flexible tool for enhancing the commercial experience.

2. User Requirements

- User can scan a QR code, create an account and add funds.
- User can purchase directly from website and redeem (example: cinema ticket and can redeem it and use it as a ticket entry).
- User can scan QR code for more details about an item.
- User can show generated personal QR code to pay with funds.
- User can exchange funds with other users.
- User can split payments (for single or multiple items in a single transaction).
- User can view transaction history and account balance.
- User can request a refund.
- User can save payment information.
- User can reset password.
- User can report bugs or unauthorized transactions.
- User can search for items/services.
- User can add item to favorites.

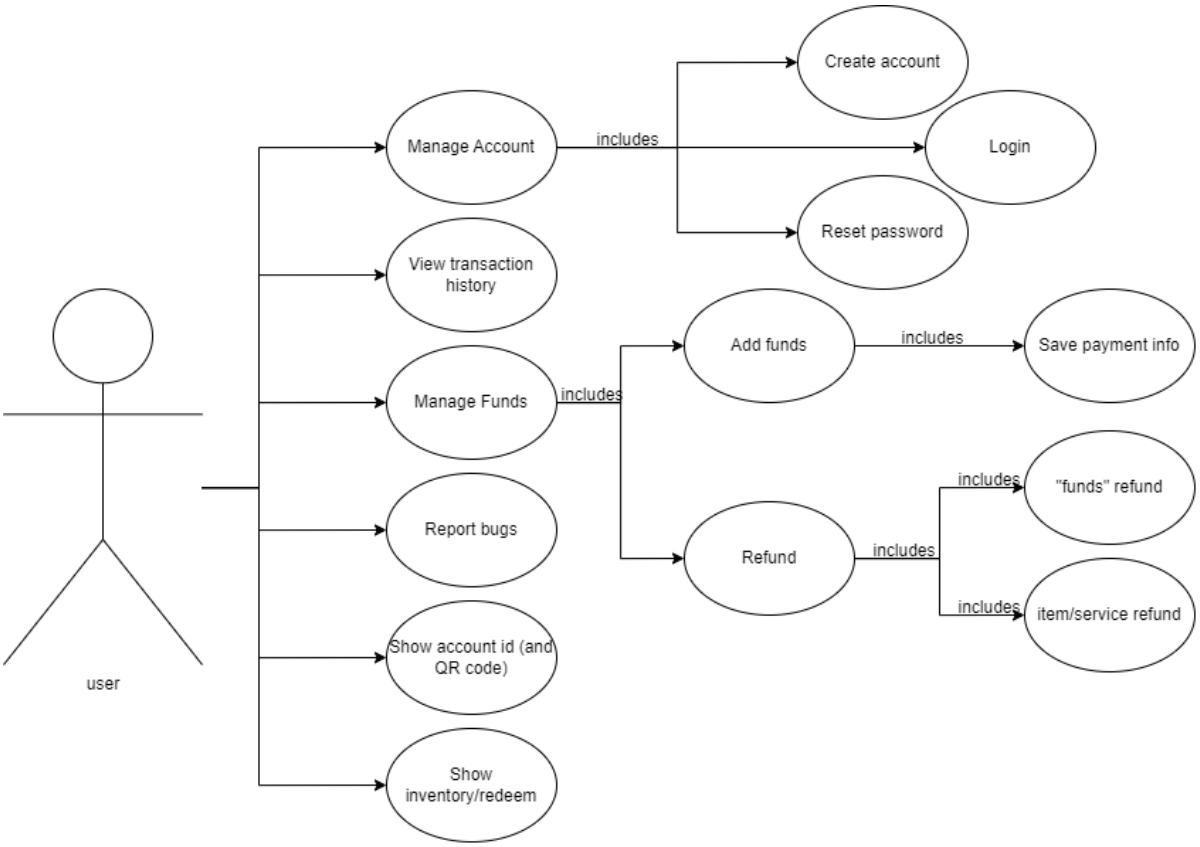

Figure 1. User Requirements 1.

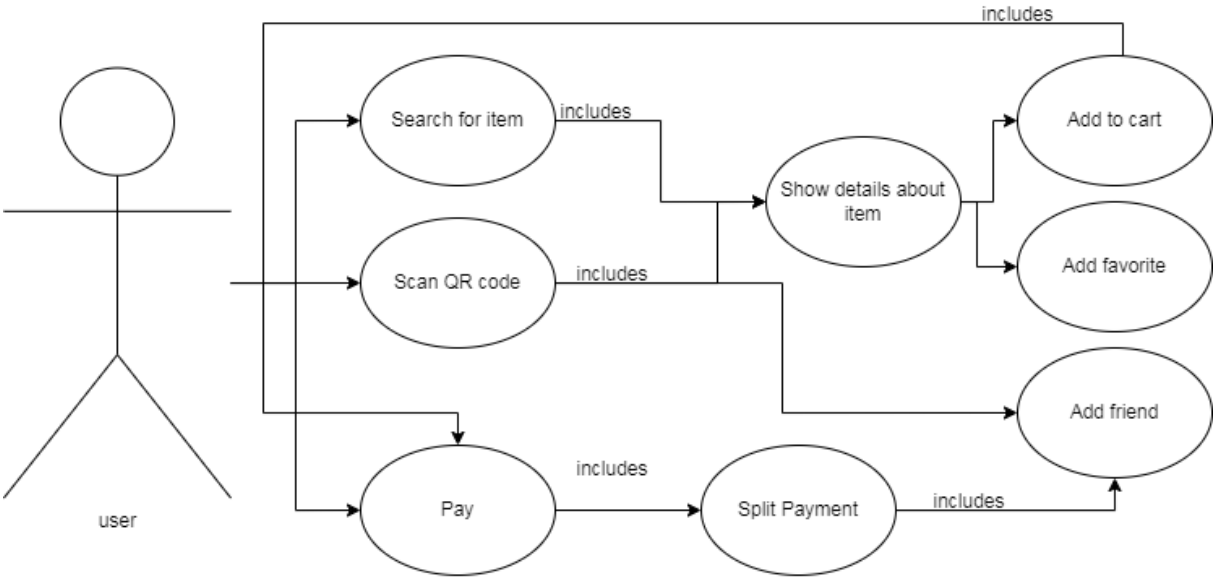

Figure 2. User Requirements 2.

3. Admin Requirements

a. Owner

Has access to all admins, can set admins, and has access to all features.

b. Section manager (store owner/manager) can:

i. Add/remove items to store.

ii. Change item prices.

iii. Make refunds to clients.

iv. View items and quantities.

v. View selling reports and statistics.

vi. Redeem coupon/ticket.

vii. Sell item to user.

c. Seller can:

i. View items and quantities.

ii. Redeem coupon/ticket for user.

iii. Sell item to user.

` Note: Admin requirements are dynamic and can be changed with each customer.

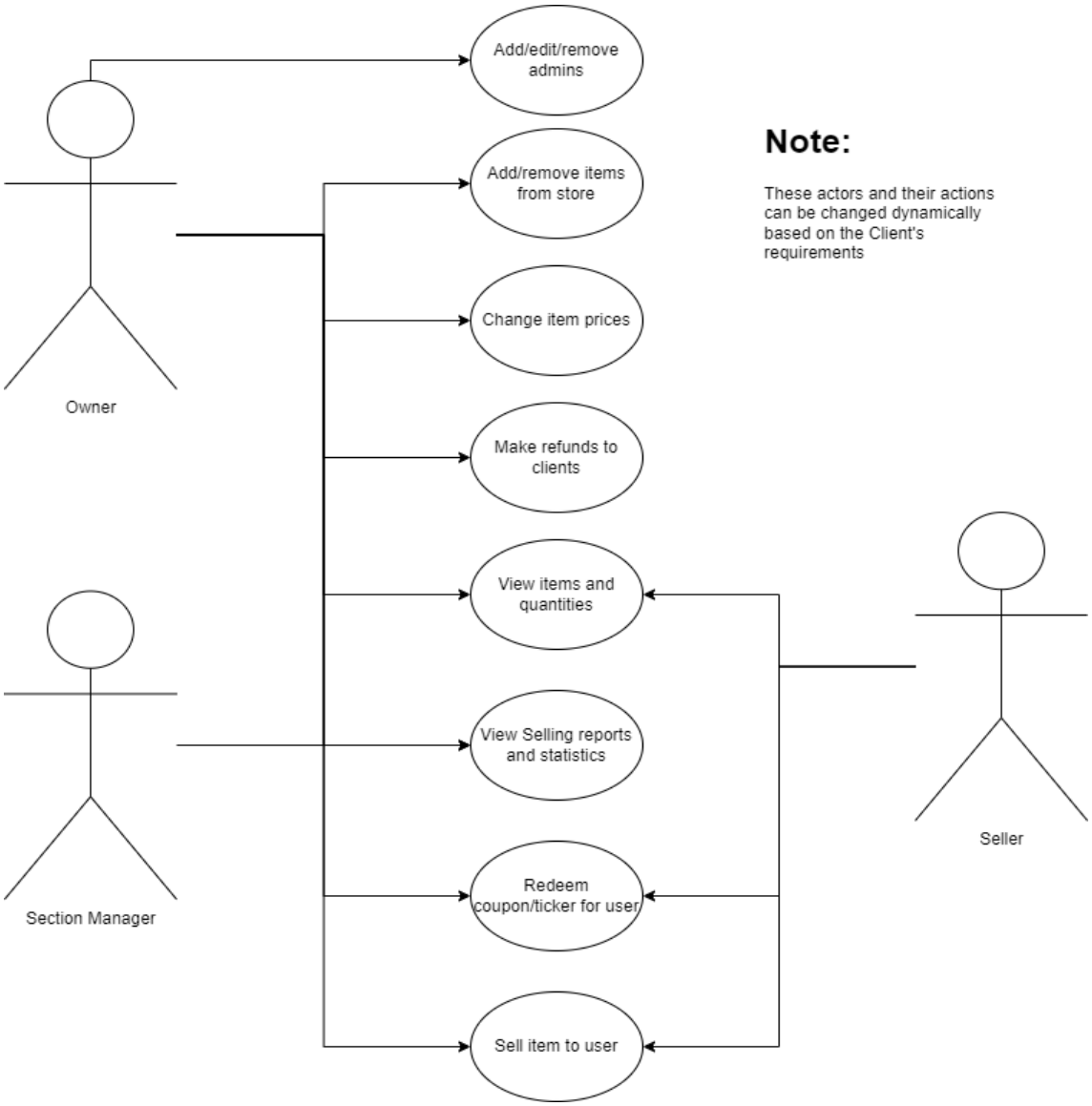

Figure 3. Admin Requirements.

IV. BACK-END RESEARCH AND IMPLEMENTATION

1. Programming Languages and Frameworks

There are several backend programming languages and frameworks that can be used to develop SLP, depending on the requirements given previously, here are some of the good and popular choices:

a. Python with Django or Flask: Python is a popular language for web development due to its simplicity, readability, and extensive libraries. Django and Flask are two popular web frameworks that make it easy to build scalable and secure web applications. Django includes features like authentication, database modeling, and admin interface, while Flask is a lightweight framework that gives more flexibility and control.

b. Java with Spring Boot: Java is a robust and mature language that has been used for enterprise web development for decades. Spring Boot is a popular Java-based framework that makes it easy to create web applications with features like security, caching, and database access. Spring Boot has a large community and many plugins and libraries available.

c. Node.js with Express.js: Node.js is a popular runtime for building scalable and fast web applications using JavaScript. Express.js is a popular framework for building web applications with Node.js, “The Express philosophy is to provide the minimal layer between your brain and the server”[1]. It is lightweight and flexible, making it easy to create REST APIs and handle HTTP requests and responses.

d. Ruby on Rails: Ruby on Rails is a popular web development framework that follows the Model-View-Controller (MVC) architecture. It includes many features out of the box, such as database access, authentication, and caching. Ruby on Rails is known for its simplicity and convention over configuration approach, making it easy to develop and deploy web applications.

Ultimately, Java with Spring Boot were chosen for developing SLP for several reasons:

Figure 4. Java Logo[2]

Figure 5. Spring Boot Logo[3]

a. Scalability: Java is known for its ability to handle large-scale projects, and Spring Boot is no exception. It provides built-in support for handling multiple concurrent requests, which makes it well-suited for high-traffic applications like payment systems.
b. Security: Security is critical for any payment system, and Spring Boot provides several security features out of the box. For example, it supports secure communication over HTTPS, encryption of sensitive data, and access control to protect against unauthorized access and attacks.
c. Modular design: Spring Boot is designed to be modular, which makes it easy to add or remove components as needed. This modular design also allows the application to scale vertically or horizontally by adding more instances of the app or by adding more servers.
d. Community support: Spring Boot has a large and active community of developers who contribute to its development and provide support through forums, documentation, and code libraries. This community support can be very valuable when developing complex systems like payment apps.
e. Database access: Spring Boot provides support for accessing databases like MySQL, PostgreSQL, and MongoDB. This makes it easy to store and manage data related to transactions, user accounts, and other information needed by the payment system.

Overall, opting with Java and Spring Boot is a reliable and scalable choice for developing a payment service. It provides many built-in features and has a large community of developers and resources available, making it a popular choice for enterprise web applications.

2. Database Environment

There are several database servers that could work well with SLP:

a. MySQL: MySQL is a popular open-source relational database management system. It is widely used for web applications and can handle large amounts of data. It is also known for its fast performance, scalability, and reliability.
b. PostgreSQL: PostgreSQL is another popular open-source relational database management system. It is known for its ability to handle complex queries, its support for JSON data, and its reliability.
c. MongoDB: "MongoDB is a document-oriented database, not a relational one"[4]. It is well-suited for applications with complex data structures, and its flexible schema allows for easier development and changes over time.
d. Oracle: Oracle is a commercial relational database management system that is known for its scalability and security. It is commonly used in large enterprises and can handle large amounts of data.

To choose a specific database server from the list above, some factors must be taken into consideration:

a. Data volume: Depending on the size of the mall and the number of stores and transactions, a database that can handle a large volume of data is needed. In this case, MySQL or Oracle are good options.
b. Data complexity: If the app requires complex data structures or supports unstructured data, (which is not the case), MongoDB may be a good fit.
c. Transactions: If the app involves a high volume of transactions and requires ACID compliance, MongoDB cannot be used.
d. Security: All the databases listed have strong security features, but Oracle is known for its advanced security capabilities (followed by MySQL), which is important for a payment app.
e. Cost: Depending on the database server, there are different costs to consider in the selection of the database environment.

Ultimately, MySQL was chosen for developing SLP for these specific reasons:

Figure 6. MySQL Logo[5]

a. First, MySQL is a widely used open-source, free relational database management system, which makes it a popular choice for many web applications. It has a large and active community, which means that anyone can easily find support, documentation, and tutorials online with no additional costs.
b. Second, MySQL is known for its reliability and scalability. It can handle large amounts of data and is able to scale horizontally to accommodate more users and data. This makes it a good fit for a payment app that will likely experience a high volume of transactions.
c. Third, MySQL is ACID-compliant, which means that it ensures data consistency and reliability. This is important for a payment app, where data integrity and security are critical.
d. Fourth, MySQL has strong security features, including encryption, access controls, and auditing. This is particularly important for a payment app, where the security of user data and financial information is of the utmost importance.
e. Last, MySQL is easy to integrate with other technologies and programming languages (specifically with Java and Spring Boot). This makes it a good fit for a payment app that may need to integrate with other systems or applications.

Overall, MySQL's reliability, scalability, security, and ease of integration make it a strong candidate for SLP.

3. Maven VS Gradle Architecture with Spring Boot

Figure 7. Gradle VS Maven[6].

3.1. What are Maven and Gradle?

Maven and Gradle are build automation tools used in the Java ecosystem for managing the build lifecycle of projects. They handle tasks such as compiling source code, managing dependencies, running tests, and packaging applications for deployment. These tools are essential for modern software development as they help streamline the build process, enforce consistency, and improve productivity.

- Maven, released in 2004, is a widely adopted build tool that uses XML for its build configuration through Project Object Model (POM) files. It follows the convention-over-configuration paradigm, which provides a well-defined project structure and sensible defaults to reduce the need for extensive configuration.
- Gradle, introduced in 2009, is another popular build tool that uses a Domain-Specific Language (DSL) based on Groovy or Kotlin for its build configuration. It is known for its flexibility, extensibility, and improved performance compared to Maven, thanks to features like incremental builds and the Gradle daemon.

3.2. Why Maven?

While both Maven and Gradle have their merits, in this project, Maven will be used with Spring Boot. The following reasons outline the advantages of using Maven over Gradle and provide a comparison between the two tools:

1. Convention over configuration: “Systems, libraries, and frameworks should assume reasonable defaults”[7]. Maven follows the convention-over-configuration paradigm, which means that it provides a well-defined project structure and sensible defaults. This approach reduces the need for

extensive configuration and makes it easy to get started with new projects, especially for developers who are familiar with the standard Maven project layout.

2. Wide adoption and community support: Maven has been around since 2004 and has gained widespread adoption in the Java community. This extensive user base has led to a wealth of resources, such as plugins, libraries, and tutorials, which can be beneficial when working on the project. While Gradle has been gaining popularity, Maven's long-standing presence in the community still provides an advantage in terms of available resources and support.
3. Consistency and predictability: Maven's declarative approach to defining project dependencies and build processes ensures that the build process remains consistent across different environments and team members. Gradle, on the other hand, uses a Groovy or Kotlin-based DSL, which allows more flexibility but can also result in less predictable and more complex build scripts.
4. Centralized dependency management: Maven's dependency management system is centralized and based on the Project Object Model (POM) file, making it easy to manage and maintain dependencies across the entire project. Gradle, while offering flexible dependency management, can sometimes require more effort to achieve the same level of organization and consistency.
5. Built-in support for Spring Boot: Spring Boot provides excellent built-in support for Maven, making it simple to set up and configure a Spring Boot project with Maven. While Spring Boot also supports Gradle, the out-of-the-box support and familiarity with Maven can contribute to a smoother development experience.

In conclusion, while both Maven and Gradle have their merits, the choice to use Maven for this project is based on its convention-over-configuration approach, wide adoption, community support, consistency, and built-in support for Spring Boot. These factors contribute to a more streamlined development experience, which is valuable in delivering a robust and efficient payment solution like SLP.

4. Backend Architecture:

In the backend implementation of the SLP app, the Spring Boot framework is employed to create a highly scalable and flexible system. To ensure that the application can handle varying levels of demand and requests, a load balancer is utilized to distribute incoming traffic across multiple instances of the services. The architecture follows a microservices approach, "Microservices are small, autonomous services that work together"[8], this architecture breaks down the application into smaller, more manageable components, improving maintainability and facilitating independent deployment of each service.

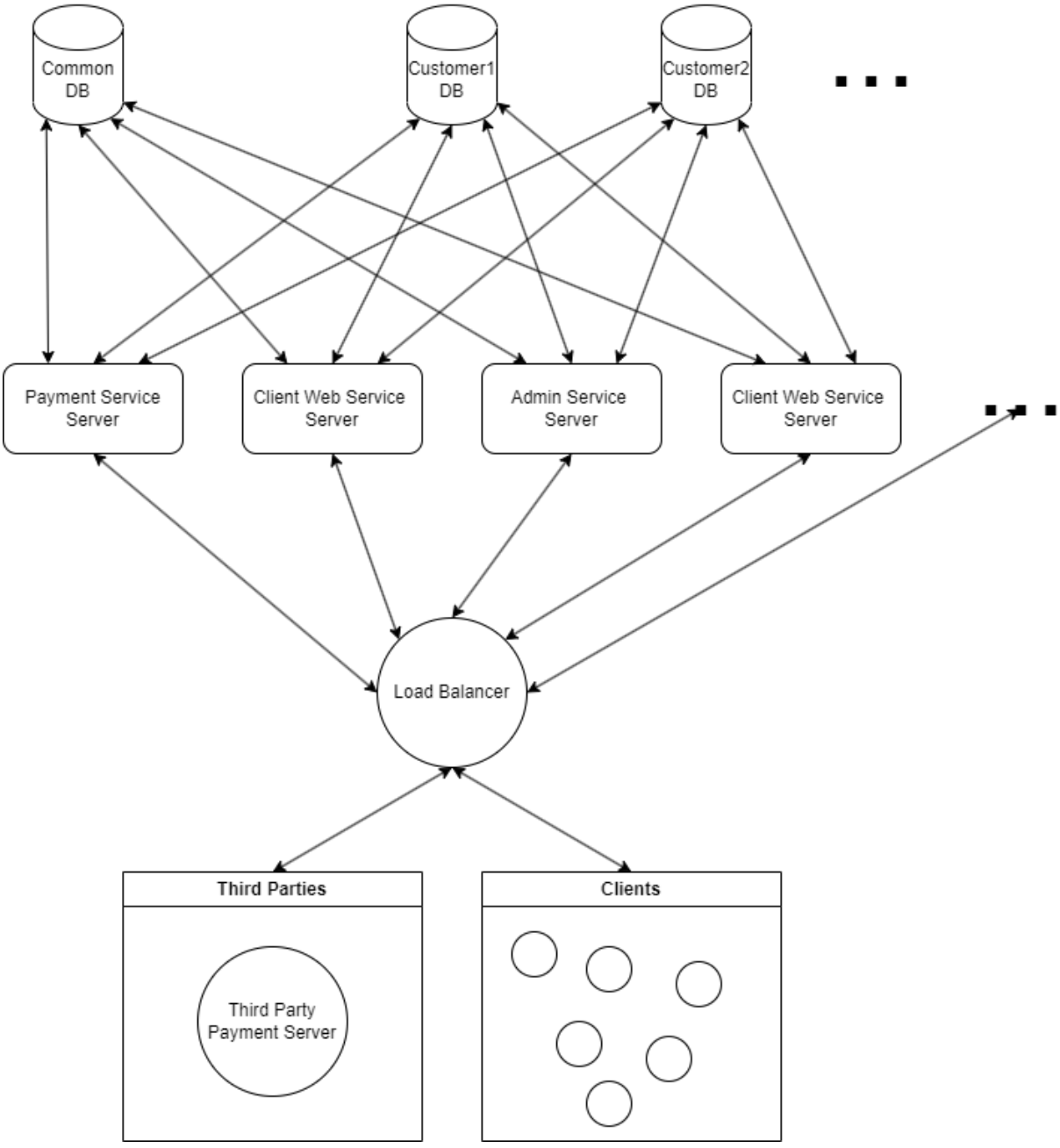

Figure 8. Back-End Architecture.

The microservices architecture in this app consists of three main services:

a. Web App Service: This service is responsible for handling all the user interactions and requests coming from the web app. It processes the client's requests, communicates with other services as needed, and sends appropriate responses back to the client.
b. Third-Party Service: This service manages the integration with third-party payment solutions, handling tasks such as processing payments, refunds, and account balance updates.
c. Authentication Service: This service ensures secure user logins and account management, utilizing robust methods like password verification and two-factor authentication to protect against unauthorized access.
d. Admin Service: This service caters to the needs of administrators and staff, enabling them to manage and monitor the application's features and transactions. It provides interfaces for tasks such as user account management, transaction tracking, and refund approvals.
e. Inventory Service: This service enables seamless integration with existing store APIs or the creation of new stock databases, catering to businesses of all sizes and technical capabilities while streamlining product management and sales.

   Notes:

   - All microservices here will be using standardized JSON request and response formats to ensure seamless communication.
   - There can be many types of third-party services since every client can use a different third party than the other.
   - Any service can be added dynamically based on the client’s requirements.

An essential aspect to highlight is that the QR-based payment system should be designed to be highly flexible and adaptable to various customer needs. In scenarios where individual stores within a mall or complex have their own APIs for managing items, fetching product details, tracking inventory, and handling sales, the system can seamlessly integrate with these existing systems through a separate dedicated service that can convert each store’s APIs to a standardized one that will be used in other microservices.

Moreover, for stores that do not have an existing API or inventory management system in place, the app offers the capability to create and integrate a new database for stock management. This ensures that businesses of all sizes and levels of technological adoption can benefit from the app's streamlined payment and inventory management functionalities.

To handle data storage efficiently, the SLP employs a multi-database approach which is called multitenancy. Each customer (e.g., a business, cinema, or mall) has a dedicated database to store its transaction data, user accounts, and other relevant information. This separation of databases helps isolate and manage each customer's data more effectively, ensuring data integrity, privacy, and security.

Dedicated Database Structure:

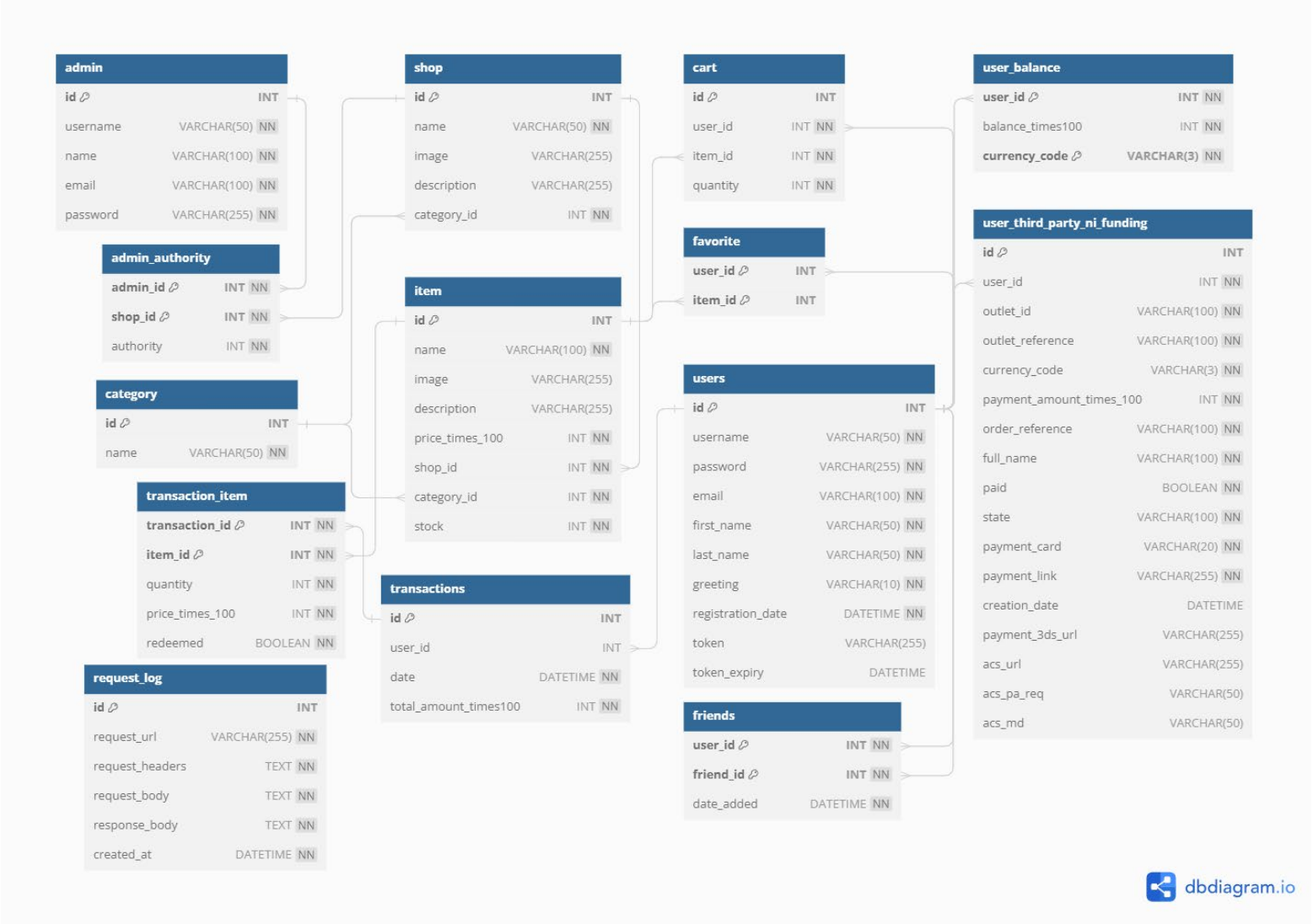

Figure 9. Dedicated Database

In addition to the individual databases, a common database is used for storing customer information. This database serves as a central repository for customer profiles, contact details, and other relevant metadata. The common database enables the app to easily retrieve and

manage customer data when needed and facilitates efficient communication between the different microservices.

Common Database Structure:

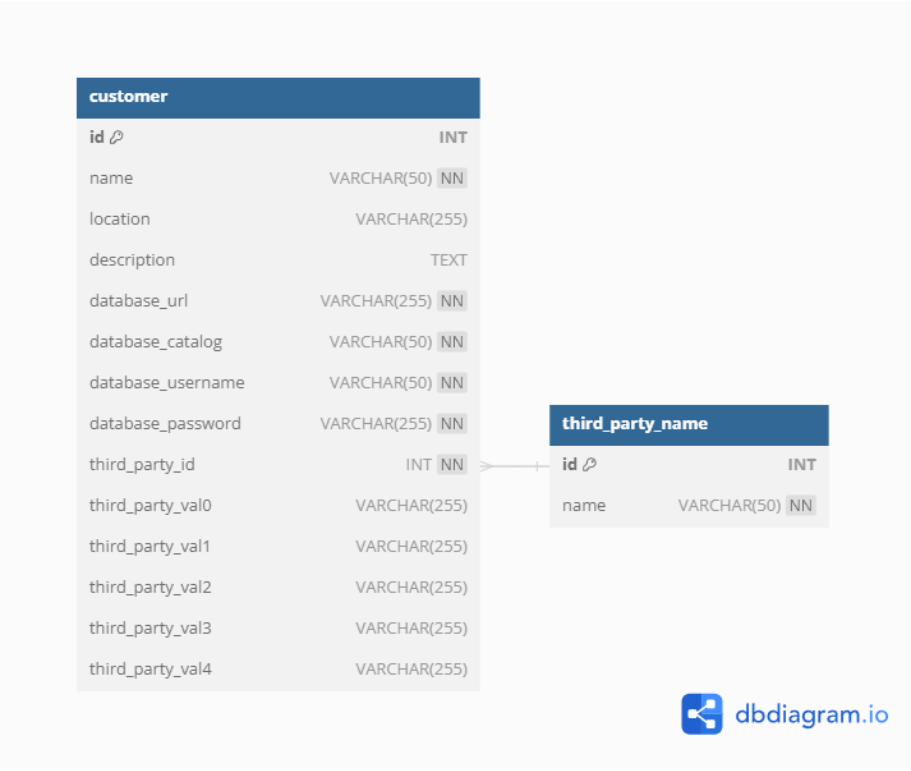

Figure 10. Common Database

In summary, the SLP's backend implementation leverages the Spring Boot framework and a microservices architecture to create a scalable, maintainable, and flexible system. The use of a load balancer and multiple databases ensures that the application can effectively handle varying demand and provide excellent performance. This robust and efficient architecture plays a critical role in delivering a seamless and enjoyable experience for customers, staff, and administrators alike.

5. Implementation Process:

a. Initial Planning and Analysis:

The genesis of the 'SLP' project was marked by a critical observation: Existing payment technologies in the market lacked comprehensive functionalities, especially for large-scale environments like malls, where each store operates as an independent entity. This gap presented an opportunity for a unified system that could integrate the diverse needs of such settings into a single, streamlined payment solution. This realization drove the initial planning phase, where the focus was on devising a system that could overcome these market limitations and offer a more cohesive experience.

b. System Design:

The design of the SLP system was grounded in established technologies but approached with innovation. Microservices and multitenancy, supported by multi-database structures and load balancers were chosen as the base system architecture. This design framework was selected for its proven scalability and responsiveness, with an added emphasis on facilitating ease of debugging. By adopting this approach, the system was poised to handle the dynamic demands of large-scale commercial environments, ensuring both robust performance and maintainability.

c. Development Process:

The development journey began with a rudimentary design sketch of the website, laying the foundation for what would become the user interface. This was followed by a meticulous planning phase for the backend architecture and database design, ensuring that each component was strategically aligned with the overall objectives. The website's implementation came next, providing a tangible interface for user interactions. The subsequent focus was on building the backend and database, the core engines driving the system's functionality.

d. Prioritizing Core Features:

In prioritizing features, significant importance was given to elements central to the user experience and operational efficiency. This included the integration of purchasing capabilities with Network International for payment processing, establishing a comprehensive item catalog, and implementing robust user authentication and security protocols. These features were essential in delivering a seamless and secure transaction experience, aligning with SLP’s goals of efficiency and user trust.

e. **Testing Methodologies:**

Testing was a critical component of the development process. Each API request to the backend was thoroughly examined through various testing methodologies. This rigorous testing regime was crucial in identifying and mitigating potential security vulnerabilities, ensuring the system's resilience against malicious activities. The backend architecture was fortified with robust authentication mechanisms, safeguarding the system from unauthorized or unintended requests.

f. **User Interface Design:**

The user interface was designed with a focus on simplicity and minimalism. This approach was aimed at enhancing user accessibility and ease of navigation, ensuring that the interface is intuitive and user-friendly.

g. **Challenges and Local Testing:**

The implementation phase was not without its challenges, particularly in the realm of multitenancy. This complex process presented unique hurdles, but experience with Java and Spring Boot helped in navigating these issues with minimal disruptions. The system was rigorously tested in a local environment, which allowed for detailed debugging and refinement without the need for an initial deployment.

6. Website Showcase and Effectiveness:

The development and testing of the 'SLP (Scan Link Pay)' system have led to several important results and insights crucial for evaluating the project's effectiveness and potential impact.

It is important to highlight the development and features of the SLP website. Although not deployed for public use, the website in its test environment exhibited several promising aspects:

- **Design and Layout:** The website features a simple and user-friendly design, emphasizing ease of navigation and transaction efficiency. The layout is intuitive, allowing users to quickly understand and use the system with minimal learning curve.
- **Functionality:** Key functionalities, such as account creation, item selection, and payment processing, were implemented and tested. The website's responsiveness and speed during these processes were notable, indicating a potential positive user experience.

- **Visual Presentation:**

The pages below represent the core components of the SLP system:

- Catalog Page:

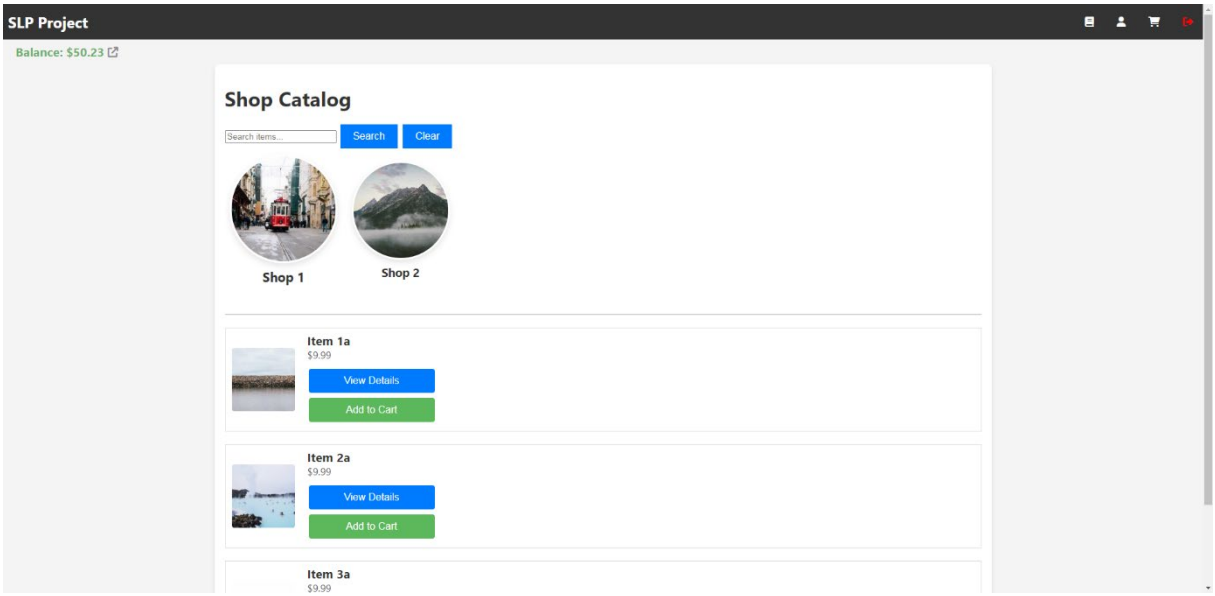

Figure 11. Catalog Page

The catalog page features a selection of shops, each leading to its range of items upon selection. A search bar allows users to look for specific items quickly. Items displayed come with an 'Add to Cart' button for immediate purchase and a 'Show Details' link leading to a more detailed page about the item.

- Shopping Cart:

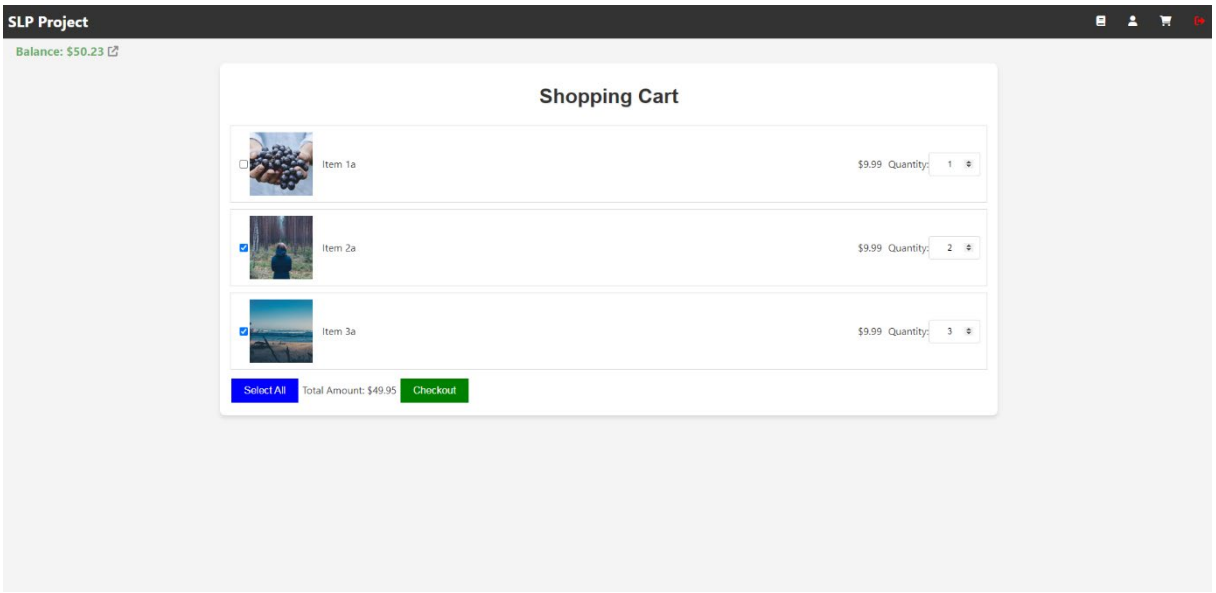

Figure 12. Shopping Cart

This page displays the items added to the user's cart. Users can select items for checkout, and upon confirming sufficient balance, the system redirects them to the transaction history page.

- Transaction History:

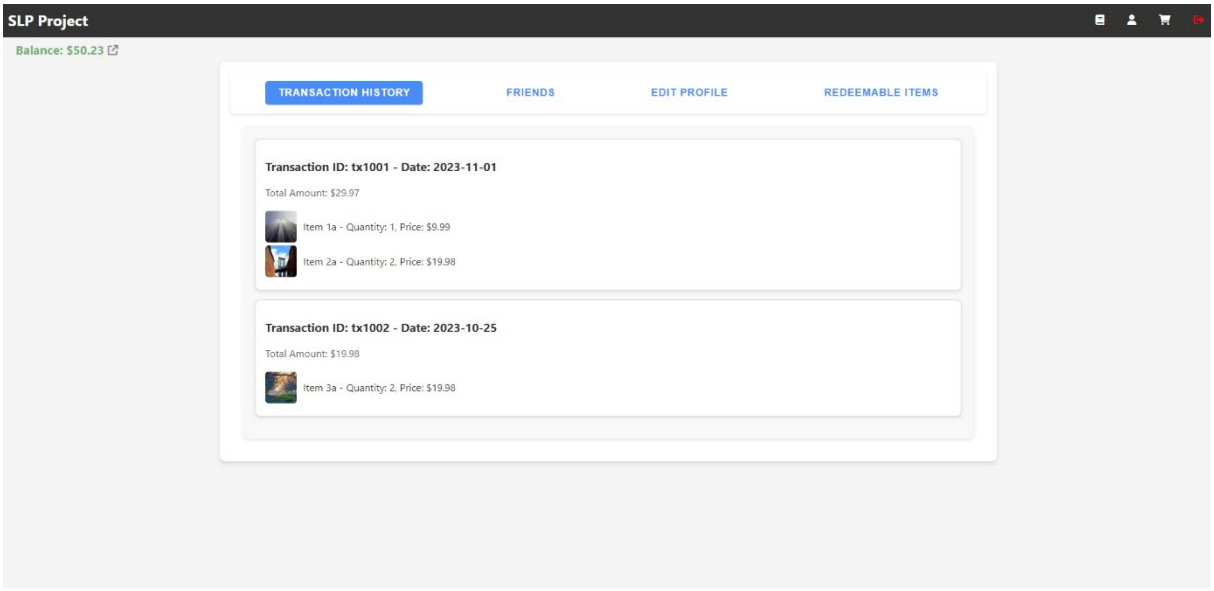

Figure 13. Transaction History

The transaction history page lists all transactions made by the user, with each transaction potentially including multiple items. This page is crucial for users to track their purchase history.

- Redeemable Items:

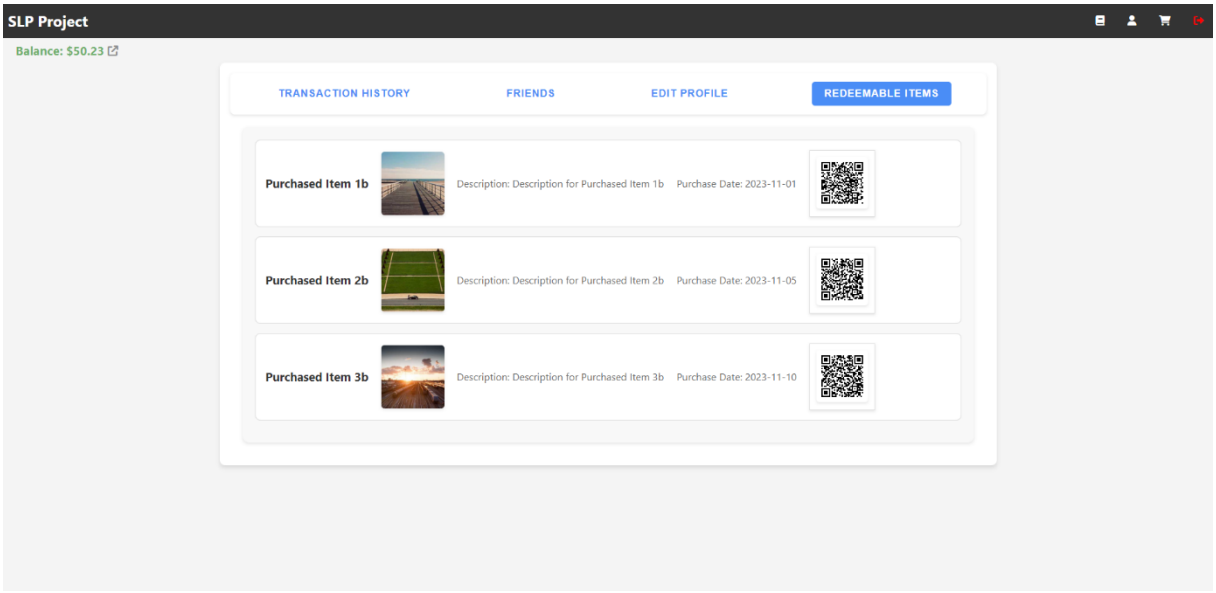

Figure 14. Redeemable Items

This section showcases items that have yet to be redeemed. Each item is associated with a displayed QR code, which shop owners can scan to redeem the item.

- Payment Screen:

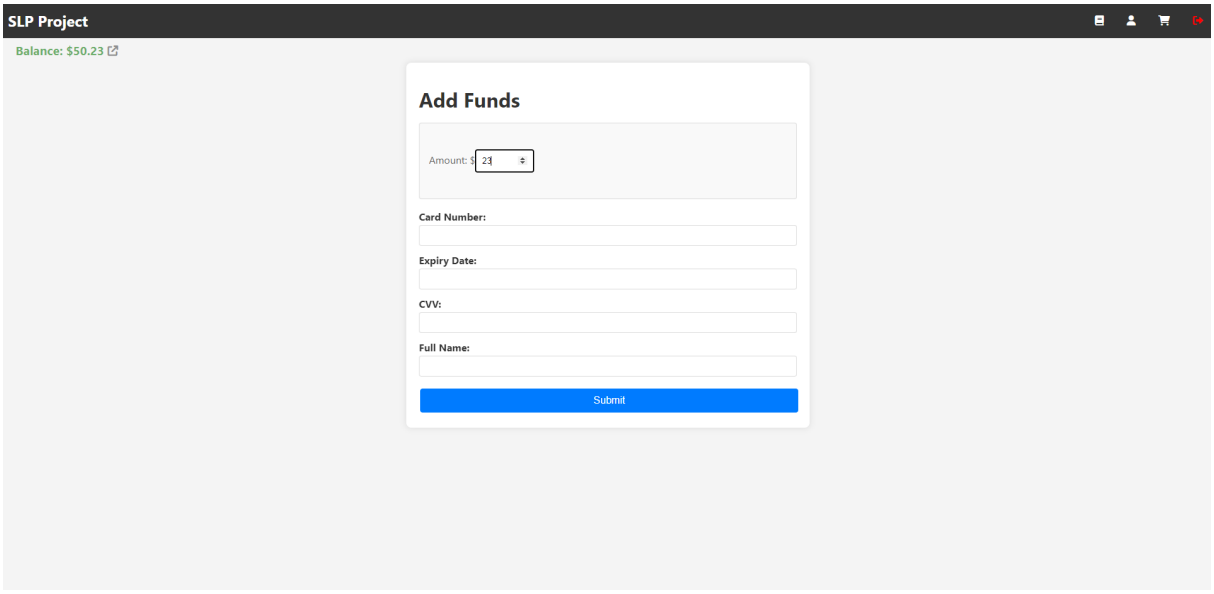

Figure 15. Payment Screen

The payment screen is where users can add funds to their account. It includes fields for entering credit card information, ensuring a secure and convenient

way for users to manage their finances within the system. This screen plays a key role in the system's functionality, facilitating easy top-ups and maintaining a seamless payment experience.

Each of these images and their functionalities contribute significantly to the SLP system's overall user experience. They collectively form a user-friendly interface that streamlines the shopping process, ensuring ease of use and efficiency for both customers and business owners.

7. Discussion of How Results Meet Objectives or Reveal New Insights:

The outcomes from the SLP system align with the project's objectives and offer new perspectives on digital payment solutions development.

- **Alignment with Objectives:** The project's results closely align with the initial goals. The system's design and implementation strategies have shown the potential to create an effective and user-friendly payment platform tailored towards large venues and malls.
- **New Insights:** The project highlighted the effectiveness of innovatively merging established technologies such as microservices and multitenancy, tailoring them to meet specific market requirements. The system's performance in test scenarios provided valuable insights into the scalability and robustness essential for payment systems in large-scale commercial settings.

V. SECURITY AND COMPLIANCE

In the process of developing a system like SLP, there is a chance of accessing sensitive data or resources including personal information of users, financial data, and third-party system credentials. To ensure compliance with relevant regulations, industry best practices, and ethical standards, it is imperative to handle and manage this data with extreme caution.

1. Security Measures Implemented in the System

In the highly sensitive domain of digital payments, the SLP system adheres to stringent security protocols to ensure the safety and confidentiality of user data.

a. **Encryption:** All sensitive data transmitted within the SLP system, including financial and personal user details, is encrypted using industry-standard protocols. This encryption ensures that data remains secure and unreadable during transmission, providing protection against interception or unauthorized access.

b. **Password Hashing:** The SLP system employs strong password hashing techniques to safeguard user accounts. When users create or update their passwords, these are hashed using advanced algorithms before being stored in the database, meaning actual passwords are never kept in a readable format. This significantly reduces the risk of data compromise in the event of a security breach.

c. **Request Logging and Monitoring:** To enhance security, the SLP system incorporates detailed request logging and continuous monitoring. Each user request is logged with timestamps and metadata, enabling the tracking of unusual activities and patterns that may indicate a security threat. This proactive approach allows for the swift identification and remediation of potential security incidents.

2. Deployment Security Measures

As part of the SLP system's deployment, great importance should be placed on security and data protection. Some of the critical measures that should be taken:

a. **Access Control:** Implement strict access control policies, granting access to sensitive data and resources only to authorized personnel on a need-to-know basis.

b. **Data Security Audits:** Conduct regular audits to assess the security of sensitive data and identify any potential vulnerabilities or breaches.

c. **Staff Training:** Provide training to all team members who handle sensitive data, ensuring they are aware of the importance of data privacy and the proper procedures for managing sensitive information.

3. Compliance with Relevant Standards and Regulations:

Adherence to legal and industry standards is a critical aspect of the SLP system's development and operation.

In the development and implementation of SLP, adherence to the ISO/IEC 27001, ISO/IEC 27002, and ISO/IEC 29100 standards will be maintained to ensure the highest level of security, privacy, and compliance.

a. **ISO/IEC 27001:**

This standard provides a framework for establishing, implementing, and maintaining an Information Security Management System (ISMS) that safeguards sensitive information, including financial transactions and personal data. By following this standard, security controls and risk management processes are implemented, ensuring that SLP consistently maintains a secure environment for its users and businesses.

b. **ISO/IEC 27002:**

This standard offers guidelines and best practices for implementing information security controls within the ISMS. By adhering to this standard, SLP employs appropriate security measures, such as access control, data encryption, and secure communication protocols, to protect against unauthorized access, data breaches, and other potential threats. This approach will ensure that the application maintains a high level of security, thereby instilling trust and confidence in its users.

c. **ISO/IEC 29100:**

This standard focuses on privacy framework, which establishes a foundation for protecting the privacy of personal information processed by SLP. By following this standard, user data will be collected, stored, and processed in a manner that respects privacy principles, such as data minimization, purpose limitation, and transparency. Implementing these standards in the design and development of SLP will not only help meet regulatory requirements but also contribute to a more secure and privacy-centric payment solution that caters to the needs of users and businesses alike.

VI. FUTURE WORK

The development of the 'SLP (Scan Link Pay)' system, while achieving significant milestones, also identified areas that warrant further attention and enhancement in future iterations of the project.

1. **Current Limitations:**
   - **Basic Admin Panel:** Due to time constraints, the admin panel in its current form is relatively basic. Enhancing its capabilities to offer more comprehensive control and analytics features is a priority for future updates.
   - **Lack of Split Payment Feature:** The split payment feature, an aspect for enhancing user convenience in transactions, is currently not implemented. This is an area that will be addressed in subsequent versions of the system.
   - **Integration with other existing systems:** Currently, the SLP system does not include integration with external systems, such as POS. This integration is complex, requiring additional time for development and testing. Moreover, it necessitates access to existing POS systems, which can involve considerable costs. This integration is crucial for a more seamless operation between SLP and physical retail environments and will be a focus in future development efforts.
   - **Time-Intensive Backend Development:** The backend structure, pivotal for the system's overall performance, required significant time for design, implementation, and debugging. This intensive development process impacted the timeline for other features.
2. **Proposed Enhancements for Future Work:**
   - **Mobile App Development:** Future development plans include creating a mobile app to complement the web-based SLP system. This app will provide users with the flexibility to manage transactions and access services on their mobile devices, enhancing accessibility and user engagement.
   - **Machine Learning for Personalized Experiences:** Implementing machine learning algorithms is envisaged to tailor the user experience more closely to individual preferences. This personalization can significantly enhance user satisfaction.
   - **Blockchain Integration for Payments:** The integration of blockchain technology will explore the potential use of cryptocurrencies as an alternative payment method within the SLP ecosystem.
   - **Chatbot for Item Discovery:** To improve user interaction and assist in discovering items and services, the addition of a chatbot is proposed. This AI-driven feature will

facilitate easier navigation and provide instant assistance, enriching the user experience.

In conclusion, while the SLP system in its current state has laid a strong foundation, there is room for advancements and additions. The focus will not only be on refining existing functionalities but also incorporating cutting-edge technologies to keep the system at the forefront of digital payment solutions. The implementation of these enhancements will further strengthen the system's capabilities, making it more aligned with emerging trends in technology.

VII. CONCLUSION

The 'SLP (Scan Link Pay)' project marks a notable advancement in the digital payment solutions landscape, specifically catering to the demands of large-scale commercial settings such as malls and cinemas. This project has illustrated not only its potential to streamline and enhance the efficiency of payment processes but also its commitment to implementing cutting-edge technologies and maintaining high standards of security and compliance.

Throughout its development, the SLP system has successfully integrated advanced technologies like microservices, multitenancy, and load balancing. These technological choices have been pivotal in creating a scalable, efficient, and robust framework, ensuring the system's readiness to handle varying demands and performance pressures. Preliminary testing primarily emphasized the security and robustness of the backend, ensuring that the system is resilient and secure against potential vulnerabilities.
Additionally, the website was designed to facilitate easy navigation and interaction for users, aligning with the project's core objectives of enhancing operational workflows and user engagement.

A key aspect of the SLP project is its unwavering focus on security and compliance. Adhering to stringent standards, the system has been designed to ensure the security and privacy of user data. This extends to implementing robust encryption methodologies and sophisticated password hashing techniques, thereby fortifying the system against potential cyber threats and data breaches. The system's adherence to international security and data protection regulations exemplifies its dedication to not only providing a seamless payment experience but also safeguarding user trust and integrity.

Despite its accomplishments, the project recognizes certain limitations due to time constraints, which have led to some features like the advanced admin panel and split payment functionality remaining underdeveloped. Moreover, the intensive process involved in the backend development has been a challenging yet crucial part of ensuring the system's effectiveness and reliability.

As the SLP project looks to the future, it is set to undergo significant enhancements. The roadmap includes the development of a mobile application, the integration of machine

learning for personalized user experiences, the adoption of blockchain technology for enhanced transaction security, and the implementation of a chatbot for improved service discovery.

In conclusion, the SLP system stands as a testament to the power of technology in reshaping the world of digital payments. It balances the drive for innovation with a strong commitment to security and compliance, ensuring that the system is not only functional and forward-looking but also trustworthy and resilient. As the digital landscape continues to evolve, the SLP system is well-positioned to lead the way in transforming payment experiences in complex commercial environments, meeting the demands of both today and tomorrow.

Notes:

- While some of the information in this report are based on books, most of the research for the back-end services/frameworks are based on the official documentation of the services.
- Database images are drawn using https://dbdiagram.io/

VIII. REFERENCES

1. [1]E. Brown, “Web Development with Node and Express: Leveraging the JavaScript Stack”, 2014.
2. [2]Java Logo https://www.oracle.com/a/ocom/docs/java-licensing-logo-guidelines-1908204.pdf
3. [3]Spring Boot Logo https://spring.io/
4. [4]S. Bradshaw, E. Brazil, and K. Chodorow, “MongoDB: The Definitive Guide”, 2013.
5. [5]MySQL Logo https://www.mysql.com/about/legal/logos.html
6. [6]Gradle VS Maven https://www.educba.com/gradle-vs-maven/
7. [7]T. O'Brien, “Maven: The Complete Reference”, 2009.
8. [8]S. Newman, “Building Microservices: Designing Fine-Grained Systems”, 2015.