University Name

Department Name

Gocashless: A Digital Payment System for Public Transportation in Zambia

Student Name: [Your Name]

Computer Number: [Your Computer Number]

Supervisor Name: [Your Supervisor's Name]

Date of Submission: [Date]

Declaration

I hereby declare that this thesis is my own work and has not been submitted in any form for another degree or diploma at any university or other institution of tertiary education.

Acknowledgements

[Optional: Insert acknowledgements here]

Abstract

[Write a 150–250 word abstract summarizing the project here.]

Table of Contents

[Automatically generate table of contents here in Word]

# 1. Introduction

## 1.1 Background and Motivation

The transportation sector in Zambia, particularly public transportation, heavily relies on cash-based transactions. This traditional payment method presents several challenges, including the risk of cash handling, revenue leakages for bus operators, and inconvenience for passengers who may not always have the exact change. The need for a modernized, cashless payment system is evident to enhance efficiency, transparency, and user experience. The Gocashless project aims to address these challenges by introducing a digital payment solution for public transportation in Zambia.

## 1.2 Problem Statement

The core problem is the inefficiency and lack of security in the current cash-based fare collection system in Zambia's public transport. This leads to revenue losses for bus companies, security risks for conductors, and a suboptimal experience for passengers. A digital payment system can mitigate these issues, but its implementation requires careful consideration of the local context, such as the prevalence of mobile money and the specific needs of the stakeholders (passengers, conductors, and bus companies).

## 1.3 Aim and Objectives

The aim of the Gocashless project is to design, develop, and implement a secure and user-friendly cashless payment system for public transportation in Zambia using QR code technology and mobile money.

The objectives of the project are:

- To develop a microservices-based backend to manage users, routes, fares, and payments.

- To create a web-based dashboard for bus companies to manage their fleet, routes, and conductors.

- To build a mobile application for conductors to generate QR codes for fare payment.

- To design a mobile application for passengers to scan QR codes and make payments using their mobile money accounts.

- To integrate the system with a mobile money provider (Airtel Money) to process transactions.

## 1.4 Significance of the Project

The Gocashless project is significant as it has the potential to revolutionize the public transportation payment system in Zambia. It offers a more secure, efficient, and convenient alternative to cash, benefiting all stakeholders. For bus companies, it provides better revenue management and reduces the risks associated with cash handling. For conductors, it simplifies the fare collection process and enhances their security. For passengers, it offers a seamless and hassle-free payment experience.

## 1.5 Scope and Limitations

The scope of this project is to develop a fully functional prototype of the Gocashless system. This includes the backend microservices, the web-based admin dashboard, and the conductor's mobile application. The passenger mobile application and the integration with the mobile money API are planned but will be implemented in a future phase of the project. The initial implementation will focus on a single bus company and a limited number of routes for testing and evaluation purposes.

## 1.6 Report Organization

This report is organized into seven chapters. Chapter 1 provides an introduction to the project. Chapter 2 presents a review of related literature. Chapter 3 describes the system analysis and design. Chapter 4 details the implementation of the system. Chapter 5 discusses the testing and evaluation. Chapter 6 provides a conclusion and recommendations for future work. Chapter 7 lists the references used in this report.

# 2. Literature Review / Related Work

## 2.1 Introduction

The transition from cash-based to digital fare collection systems is a global trend in the public transport sector. This chapter reviews existing literature on digital fare collection systems, with a focus on developing countries and technologies relevant to the Gocashless project. It explores the benefits, challenges, and different technological approaches, providing a context for the design and implementation of the Gocashless system.

## 2.2 Automated Fare Collection (AFC) Systems

Automated Fare Collection (AFC) systems have been identified as a key component in modernizing urban transport. According to a report by the World Bank, AFC systems can enhance efficiency, reduce revenue leakage, and improve the overall commuter experience [1]. These systems enable transport operators to gather valuable data on travel patterns, which can be used to optimize routes and schedules. Furthermore, they can facilitate the implementation of targeted subsidies for specific user groups, such as students or the elderly [1, 2].

## 2.3 Technological Landscape

The technological landscape of digital fare collection is diverse and evolving. While traditional systems have relied on proprietary smartcards, there is a growing shift towards more open and interoperable solutions [4].

### 2.3.1 Smartcards

Smartcard-based systems, like the Oyster card in London, have been widely adopted. However, these are often "closed-loop" systems, meaning the card can only be used for transport. This can be a limitation for users who have to carry multiple cards for different purposes. The implementation and maintenance of such systems can also be costly [3].

### 2.3.2 Mobile Payments and QR Codes

With the proliferation of smartphones, mobile payment solutions are becoming increasingly popular for fare collection. These include Near-Field Communication (NFC) payments and QR code-based systems. Mobile money services, such as M-Pesa in Kenya, have seen massive adoption in Africa and represent a significant opportunity for fare collection [3, 5]. QR codes offer a low-cost and easily deployable alternative to traditional AFC systems, as they do not require specialized hardware for the user.

## 2.4 Challenges in Developing Countries

The implementation of digital fare collection systems in developing countries faces a unique set of challenges. These include the high cost of implementation, the need for a reliable power supply and internet connectivity, and the informal nature of the public transport sector in many countries [3]. There can also be resistance from stakeholders, such as conductors and drivers, who may be accustomed to the cash-based system. User adoption can also be a hurdle, as some individuals may not have access to smartphones or may not be comfortable with digital payments [3].

## 2.5 Case Studies in Africa

Several African cities have embarked on the journey of digitizing their transport fare collection systems. Cities like Kigali, Lagos, and Nairobi have implemented various forms of cashless payment systems, with varying degrees of success [3, 5]. These case studies highlight the importance of a phased approach, strong political will, and the need to address the specific needs and concerns of all stakeholders. For instance, in Nairobi, the initial attempt to introduce a mandatory cashless system faced resistance and was eventually made optional.

## 2.6 Gaps and Opportunities

The literature reveals a clear trend towards mobile-first and open-loop payment systems for public transport. While there are case studies from several African cities, there is a lack of specific research on the implementation of such systems in the Zambian context. The Gocashless project aims to fill this gap by developing a system that is tailored to the local environment, leveraging the high mobile money penetration in Zambia. The use of QR codes as the primary payment mechanism is a cost-effective and scalable approach that has the potential to overcome some of the challenges faced by other systems.

# 3. System Analysis and Design

## 3.1 Introduction

This chapter outlines the system analysis and design of the Gocashless project. It details the functional and non-functional requirements, the system architecture, the technology stack, and the database design. The design is based on a microservices architecture to ensure scalability, flexibility, and maintainability.

## 3.2 Requirements

### 3.2.1 Functional Requirements

- \*\*User Management:\*\* The system shall be able to register and authenticate different user roles, including passengers, conductors, bus company administrators, and system administrators.

- \*\*Route and Fare Management:\*\* The system shall allow administrators to create, update, and manage bus routes, stops, and fares.

- \*\*QR Code Generation:\*\* The system shall enable conductors to generate unique QR codes for each transaction based on the selected route and fare.

- \*\*Payment Processing:\*\* The system shall integrate with a mobile money API to process payments when a passenger scans a QR code.

- \*\*Transaction History:\*\* The system shall maintain a history of all transactions for auditing and reporting purposes.

- \*\*Admin Dashboard:\*\* The system shall provide a a web-based dashboard for administrators to manage the system and view reports.

### 3.2.2 Non-Functional Requirements

- \*\*Security:\*\* The system must ensure the security of all transactions and user data.

- \*\*Scalability:\*\* The system must be able to handle a large number of concurrent users and transactions.

- \*\*Reliability:\*\* The system must be highly available and resilient to failures.

- \*\*Usability:\*\* The user interfaces for the web and mobile applications must be intuitive and easy to use.

## 3.3 System Architecture

The Gocashless system is designed using a microservices architecture. This architectural style structures the application as a collection of loosely coupled services, each responsible for a specific business capability. This approach offers several advantages, including improved scalability, fault isolation, and technology diversity.

The system consists of the following components:

- \*\*API Gateway:\*\* A single entry point for all client requests, which routes them to the appropriate microservice.

- \*\*Eureka Server:\*\* A service discovery mechanism that allows microservices to find and communicate with each other.

- \*\*Backend Microservices:\*\* A set of independent services that implement the core business logic.

- \*\*Frontend Applications:\*\* Web and mobile applications that provide the user interface.

## 3.4 Technology Stack

- \*\*Backend:\*\* Java, Spring Boot, Spring Cloud

- \*\*Frontend (Web):\*\* Next.js, React, Tailwind CSS

- \*\*Frontend (Mobile):\*\* React Native

- \*\*Database:\*\* PostgreSQL

- \*\*Service Discovery:\*\* Netflix Eureka

- \*\*API Gateway:\*\* Spring Cloud Gateway

## 3.5 Database Design

The system uses a PostgreSQL database. Each microservice has its own database to ensure loose coupling. The main entities in the database include:

- \*\*Users:\*\* Stores information about all users of the system.

- \*\*Routes:\*\* Stores information about bus routes.

- \*\*BusStops:\*\* Stores information about bus stops along a route.

- \*\*Fares:\*\* Stores information about the fares for different routes.

- \*\*Transactions:\*\* Stores information about all payment transactions.

# 4. Implementation

## 4.1 Introduction

This chapter describes the implementation of the Gocashless system. It details the development of the backend microservices, the frontend web application, and the mobile application for conductors. The implementation follows the design outlined in the previous chapter, using the specified technology stack.

## 4.2 Backend Development

The backend is developed as a set of microservices using Java and the Spring Boot framework. This modular approach allows for independent development, deployment, and scaling of each service.

### 4.2.1 User Management Service

This service is responsible for all user-related operations, including registration, authentication, and profile management. It exposes a set of REST APIs for other services and frontend applications to consume. The service uses Spring Security for authentication and authorization.

### 4.2.2 Route and Fare Management Service

This service manages the creation, retrieval, updating, and deletion of bus routes, stops, and fares. It provides REST APIs for the admin dashboard to manage this data.

### 4.2.3 QR Code Generation Service

This service generates QR codes based on the route and fare information provided by the conductor's mobile app. It uses the ZXing library to generate the QR code images.

### 4.2.4 Payment Processing Service

This service is responsible for integrating with the Airtel Money API to process payments. It handles the payment initiation, confirmation, and transaction recording. Due to the complexities of integrating with a live payment gateway, this service is currently in a prototype stage, with a mock implementation for testing purposes.

## 4.3 Frontend Development

### 4.3.1 Admin Dashboard

The admin dashboard is a web application built with Next.js and Tailwind CSS. It provides a user interface for administrators to manage users, routes, fares, and view transaction reports. The dashboard communicates with the backend microservices via the API Gateway.

### 4.3.2 Conductor Mobile App

The conductor's mobile app is developed using React Native. It allows conductors to log in, select their route, and generate QR codes for passengers. The app is designed to be simple and intuitive, with a focus on ease of use in a fast-paced environment.

## 4.4 Challenges and Solutions

One of the main challenges during the implementation was the integration of the various microservices. To address this, we used Spring Cloud Gateway as an API gateway to provide a single entry point for all frontend requests, and Netflix Eureka for service discovery. This simplified the communication between the services and the frontend applications.

Another challenge was the real-time notification of payment confirmation to the conductor's app. This is planned to be solved by implementing a notification service using WebSockets or a similar technology in a future iteration.

# 5. Testing and Evaluation

## 5.1 Introduction

This chapter describes the testing and evaluation process for the Gocashless system. The goal of this process is to ensure that the system meets the specified requirements and is of high quality. A comprehensive testing strategy was adopted, including unit testing, integration testing, and user acceptance testing.

## 5.2 Unit Testing

Unit tests were written for the individual components of the backend microservices. The JUnit framework was used to write and execute the tests. The tests focused on verifying the correctness of the business logic in the service layer and the data access layer. Mockito was used to mock dependencies and isolate the components under test.

## 5.3 Integration Testing

Integration tests were conducted to verify the interaction between the different microservices. These tests focused on the communication between the services through the API Gateway and the service discovery mechanism. Postman was used to send requests to the API Gateway and verify the responses from the microservices.

## 5.4 User Acceptance Testing

User acceptance testing (UAT) was conducted with a small group of target users, including a bus company administrator and a conductor. The users were asked to perform a set of tasks using the admin dashboard and the conductor's mobile app. The feedback from the UAT was used to identify and fix usability issues.

## 5.5 Test Results

The unit tests and integration tests achieved a high level of code coverage and helped to identify and fix several bugs before deployment. The UAT revealed some minor usability issues, which were addressed in subsequent iterations of the development. Overall, the testing and evaluation process demonstrated that the Gocashless system is reliable, functional, and meets the core requirements of the project.

# 6. Conclusion and Recommendations

## 6.1 Conclusion

This project has successfully designed, developed, and tested a prototype of the Gocashless system, a cashless payment solution for public transportation in Zambia. The system, built on a microservices architecture, provides a solid foundation for a full-fledged implementation. The project has achieved its primary objectives of creating a user management system, a route and fare management system, a QR code generation system, and a web-based admin dashboard. The conductor's mobile app provides a simple and effective way for conductors to collect fares.

## 6.2 Limitations

The current implementation has some limitations. The passenger mobile application has not been developed, and the integration with the Airtel Money API is a mock implementation. The system has not been tested in a live environment with a large number of users. The security aspects of the system, while considered in the design, require further hardening and a thorough security audit.

## 6.3 Recommendations for Future Work

Future work on the Gocashless project should focus on the following areas:

- \*\*Passenger Mobile App:\*\* Develop the passenger mobile app to allow users to scan QR codes and make payments.

- \*\*Payment Gateway Integration:\*\* Complete the integration with the Airtel Money API to enable real-time payment processing.

- \*\*Notification Service:\*\* Implement a real-time notification service to inform conductors of successful payments.

- \*\*Offline Functionality:\*\* Add offline capabilities to the conductor's app to handle situations with poor or no internet connectivity.

- \*\*Data Analytics:\*\* Enhance the admin dashboard with data analytics features to provide insights into travel patterns and revenue.

- \*\*Security Audit:\*\* Conduct a comprehensive security audit of the entire system.

- \*\*Pilot Program:\*\* Launch a pilot program with a bus company to test the system in a real-world environment.

# 7. References

[1] nl4worldbank.org

[2] themorning.lk

[3] worldbank.org

[4] worldbank.org

[5] mobiliseyourcity.net