

**MULUNGUSHI UNIVERSITY**

Pursing the frontiers of knowledge

**SCHOOL OF SCIENCE, ENGINEERING & TECHNOLOGY**

**DEPARTMENT OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY**

**ICT382 – SOFTWARE ENGINEERING**

**COURSE PROJECT**

Names: Evans .C. Chongo Paul Kunda Joseph Mwanza

Student ID No: 201701615 201600507 201600171

Program: BCS BCS BIT

Title: Remote Banking System

Lecturer: Dr. M. J. Simfukwe (PhD)

June 2019

Table Contents

[CHAPTER ONE 1](#_Toc10540657)

[1.0 INTRODUCTION 1](#_Toc10540658)

[CHAPTER TWO 2](#_Toc10540659)

[2.1 Software Requirements Specification 2](#_Toc10540660)

[**2.1.1**  **Purpose** 2](#_Toc10540661)

[**2.1.2**  **Document Conventions** 2](#_Toc10540662)

[**2.1.3**  **Intended Audience and Reading Suggestions** 3](#_Toc10540663)

[**2.1.4**  **Product Scope** 3](#_Toc10540664)

[2.2. Overall Description 3](#_Toc10540665)

[**2.2.1**  **Product Perspective** 3](#_Toc10540666)

[**2.2.2**  **Product Functions** 3](#_Toc10540667)

[**2.2.3**  **User Classes and Characteristics** 4](#_Toc10540668)

[**2.2.4**  **Operating Environment** 4](#_Toc10540669)

[**2.2.5**  **Design and Implementation Constraints** 4](#_Toc10540670)

[**2.2.6**  **User Documentation** 4](#_Toc10540671)

[**2.2.7**  **Assumptions and Dependencies** 4](#_Toc10540672)

[2.3. External Interface Requirements 5](#_Toc10540673)

[**2.3.1**  **User Interfaces** 5](#_Toc10540674)

[**2.3.2**  **Hardware Interfaces** 5](#_Toc10540675)

[**2.3.3**  **Software Interfaces** 5](#_Toc10540676)

[**2.3.4**  **Communications Interfaces** 5](#_Toc10540677)

[2.4. System Features 5](#_Toc10540678)

[**2.4.1    Login/Logout** 6](#_Toc10540679)

[**2.4.2     Pay Zesco bills** 6](#_Toc10540680)

[**2.4.3     View Account** 6](#_Toc10540681)

[**2.4.4   Perform an EFT** 7](#_Toc10540682)

[2.5. Other Nonfunctional Requirements 7](#_Toc10540683)

[**2.5.1     Performance Requirements** 7](#_Toc10540684)

[**2.5.2**  **Safety Requirements** 7](#_Toc10540685)

[**2.5.3**  **Security Requirements** 7](#_Toc10540686)

[**2.5.4**  **Software Quality Attributes** 7](#_Toc10540687)

[**2.5.5**  **Business Rules** 8](#_Toc10540688)

[CHAPTER THREE 9](#_Toc10540689)

[3.1 Use Case Diagram 9](#_Toc10540690)

[3.2 Class Diagram 10](#_Toc10540691)

[3.3 Sequence Diagram 11](#_Toc10540692)

[3.4 Entity Relationship Diagram 16](#_Toc10540693)

[3.5 Data Flow Diagram 17](#_Toc10540694)

[3.6 Component Diagram 19](#_Toc10540695)

[3.7 Activity Diagram 20](#_Toc10540696)

[3.8 Deployment Diagram 21](#_Toc10540697)

[CHAPTER FOUR 22](#_Toc10540698)

[4.1 Client’s View 22](#_Toc10540699)

[4.2 Banker’s View 27](#_Toc10540700)

[References 31](#_Toc10540701)

# CHAPTER ONE

## 1.0 INTRODUCTION

# Online banking is the wave of the future, with the growth in use of computers and smartphones people are relying on stress-free solutions to do their day by day duties. Remote banking system (RBS) is a web-based application that provide enormous benefits to consumers in terms of easy transactions through the internet. The RBS allows the clients to transfer funds and pay Zesco bills from the comfort of their offices, hotel-room, home, etc.

# CHAPTER TWO

## 2.1 Software Requirements Specification

### **2.1.1 Purpose**

The purpose of this document is to illustrate the requirements of the Remote Bank System. The document gives the detail of both functional and non-functional requirements described by the client.

### **2.1.2 Document Conventions**

RBS – Remote Bank System.

DB – This refers to the database of the system.

Client - These are intended users of the system.

Banker (Accountant) – Register a new user (client) to the system and assign them with password and username.

SQL- Structured Query Language which is used to retrieve information from the Database

Apache- The web server

HTTP = Hypertext transfer (or transport) protocol.

MYSQL-is a relational database management system (RDBMS) that runs as a server providing multi user access to a number of databases.

PHP – Hypertext Preprocessor

HTML – Hypertext make-up language.

CSS - Cascading Style Sheets.

JavaScript - an object-oriented computer programming language commonly used to create interactive effects within web browsers.

SSL – Secure Socket Layer.

URD-User Requirements Document.

SMTP – Secure Message transfer Protocol.

EFT – Electronic Fund Transfer.

### **2.1.3 Intended Audience and Reading Suggestions**

The intended users of the system are the clients (account holders), bank clerks and the admin (bank employee in charge of the system).

### **2.1.4 Product Scope**

RBS is a web-based system that will help clients to perform the following actions:

* Perform an EFT, transfer and receive funds to/from other clients who have accounts with the same bank
* Make bill payments to Zesco.

The system also has a second user called a banker who will be able to perform the following actions:

* Generate a report based on one client of the account based on given period of time.
* Generate a report of all clients of the bank based on given duration.

# 2.2. Overall Description

### **2.2.1 Product Perspective**

The RBS will be an assistant to the traditional banking system, the following are the features that will be included in the RBS:

* Transfer and Receive funds from other clients on the same bank.
* View client a bank statement
* Banker generate reports of transactions made by clients at a given period.
* Banker generate reports of transactions made by a specific client at a given period.

### **2.2.2 Product Functions**

* The RBS provides online real time account information to the clients. The software is capable of transferring funds between accounts in the same bank, allow the customer to pay Zesco bills and generate bank statements.
* The system will also generate various reports requested by bankers (accountants).
* The bankers (accountants) with a certain level of authorization can add customers to the system.

### **2.2.3 User Classes and Characteristics**

The characteristics that are available for the clients are:

* Login/Logout of the system.
* Perform an EFT
* Transfer funds to other clients

### **2.2.4 Operating Environment**

The product will be operating on any operating system. The RBS is web-based and will operate in a web browser such as Internet explorer, safari, google chrome etc.

### **2.2.5 Design and Implementation Constraints**

Any transactions and changes regarding the clients account should be notified immediately. Any update regarding the client’s account is to be recorded and updated with correct value.

### **2.2.6 User Documentation**

The users will be able to download the user manual from the RBS.

### **2.2.7 Assumptions and Dependencies**

* The system will only allow a client to sign up if they have an account with the bank.
* A client cannot transfer more money than what he/she currently has in the account.

# 2.3. External Interface Requirements

### **2.3.1 User Interfaces**

User interface will provide the following features:

* Responsive to fit a screen size including mobile screen devices.
* Efficient and user friendly.
* Making the interface consistent.

### **2.3.2 Hardware Interfaces**

·         256 color display.

·         Operating system: Windows Operating system.

·         Hard Disk: 250 GB

·         RAM: 4 GB

·         Processor: Pentium Dual core.

### **2.3.3 Software Interfaces**

·         MySQL 5

·         PHP 7

·         Apache server

·         Web kit

### **2.3.4 Communications Interfaces**

·         HTTP

·         SSL

·         SMTP

# 2.4. System Features

The functional requirements of the system include:

1. Log-in and Log-out.
2. Pay ZESCO bills.
3. View Account
4. Perform an EFT.

### **2.4.1    Login/Logout**

#### **2.4.1.1 Description and Priority**

The purpose of this module is to enable the client and the banker to login and log-out of the system.

#### **2.4.1.2 Functional Requirements**

For a customer to be able to use the system, one has to login. This function is for both the admin (Banker) and the client.

1. Login: The input in this function must be a valid username and valid password and the output if the user is a valid user then he/she will get into a page which make him/her be able to perform transaction, but if the user made wrong inputs be it password or username then he/she will be an invalid user and will be prompted with a message “Alert invalid username and password” and asked to login with correct details.
2. Logout: This function is utilized when a logged in client completes his/her activity and needs to be logged out with the goal that nobody can abuse his/her username and password. The system will notify the client when logging out was/is successfully.

### **2.4.2     Pay Zesco bills**

#### **2.4.2.1 Description and Priority**

The purpose of this module is to enable the clients to pay bills (transfer) funds from their personal accounts to the Zesco Account (We assume the Zesco Account is on the same bank as the clients).

#### **2.4.2.2 Functional Requirements**

Firstly, the customer must be logged into the RBS for one to be able to make payments to corporations (in this case ZESCO)

### **2.4.3     View Account**

2.4.3.1 Description and Priority

The purpose of this module is to enable the client to view account balance and generate bank statements.

### **2.4.4   Perform an EFT**

#### **2.4.4.1 Description and Priority**

The module enables the client to transfer funds from their account to another user account within the same bank.

#### **2.4.4.2 Stimulus/Response Sequences**

The client transferring funds clicks the submit button to enable funds transfer. If the account has insufficient funds the system will alert the client.

#### **2.4.4.3 Functional Requirements**

 REQ-1: The client needs to be logged in to perform EFT.

## 2.5. Other Nonfunctional Requirements

### **2.5.1     Performance Requirements**

The system shall accommodate a large number of user and multiple transactions without any faults.

The system should be fast and accurate. The system should be available 24/7 to the users.  The system should have good error handling to prevent long down time.

### **2.5.2 Safety Requirements**

The database may get corrupted due to virus or operating system failure. Therefore, it is required to take the database backup so that the database is not lost. Proper ups or inverter facility should be there in case of power supply failure.

### **2.5.3 Security Requirements**

* The system will use a secured database.
* Proper user authentication should be provided
* Password encryption using MD5 algorithm.
* Session timeout that will keep the user data secure.

### **2.5.4 Software Quality Attributes**

* The system should be available 24 hours a day and specifically during peak hours.
* The quality of the DB should be maintained in such a way that it can be very user friendly to all the users of the database.
* Banker generate a report on the transitions made by all the clients at a given period (say a month of January).
* Banker generate a report on the transitions made by a specific client at a given period.

### **2.5.5 Business Rules**

The users should avoid misconduct and protocols. The users should respect the rules and regulations of the system.

# CHAPTER THREE

## 3.1 Use Case Diagram

Use case diagrams depict a set of use cases and actors and their relationships. Use case diagrams address the static use case view of a system, these diagrams are important in organizing and modeling the behaviors of a system.

According to the requirements document that was given, the only actors in RBS is the client and the banker the use case diagram showing the roles that the two actors play when interacting with the system is shown in Figure 1:

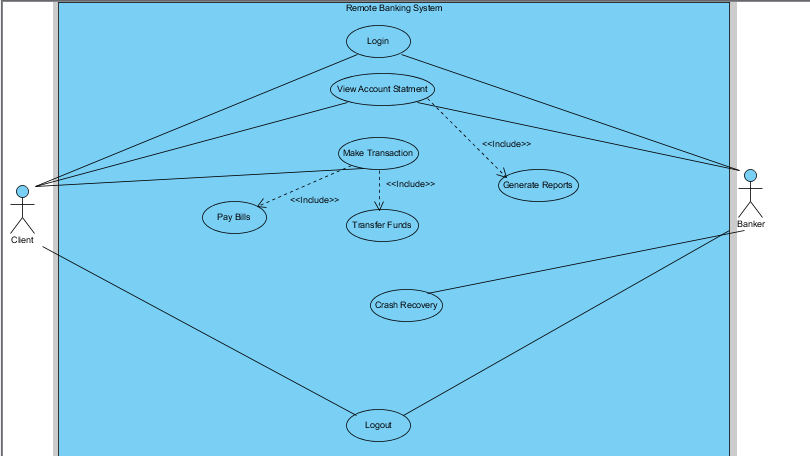


Figure 1. Use Case Diagram for RBS. 1

## 3.2 Class Diagram

Class diagram, one of the most commonly used diagrams in object-oriented system, models the static design view for a system. The static view mainly supports the functional requirements of a system – the services the system should provide to the end users the class diagram of the RBS is shown in Figure 2:

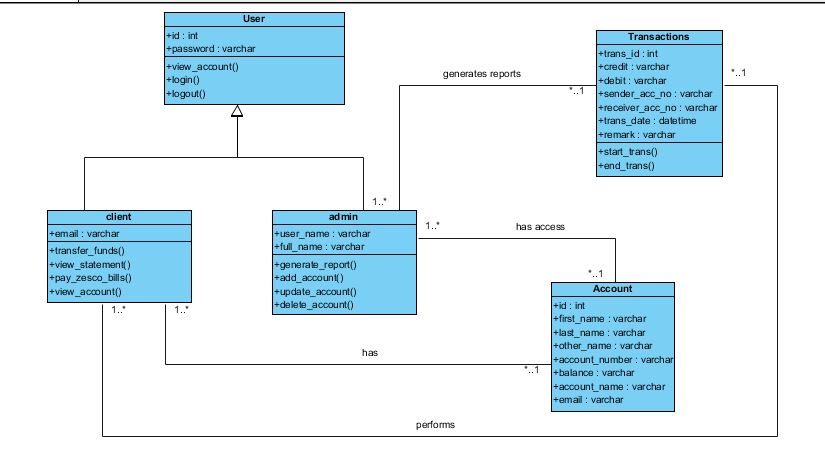


Figure 2. Class Diagram for RBS. 1

## 3.3 Sequence Diagram

Sequence diagram is one kind of interaction diagrams, which shows an interaction among a set of  
objects and their relationships. The purpose of the Sequence diagram is to document the sequence of messages among objects in a time-based view.

* The sequence diagram (**transferring of funds by client**) of the RBS is shown in Figure 3:

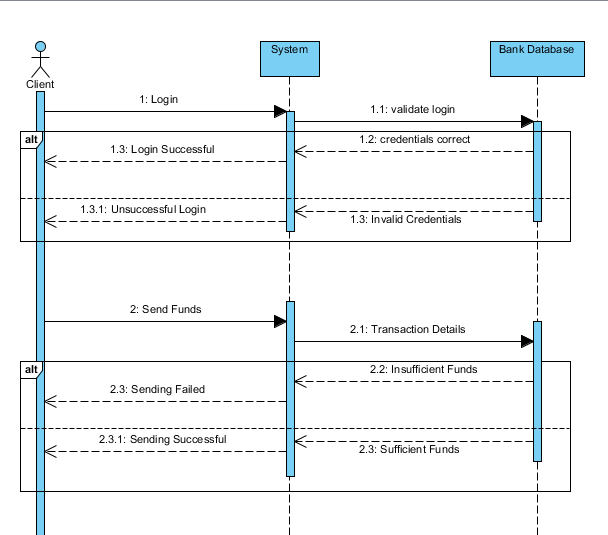


Figure 3. Client funds transfer sequence diagram for RBS.1

* The sequence diagram (**making a bill payment by client**) of the RBS is shown in Figure 4:

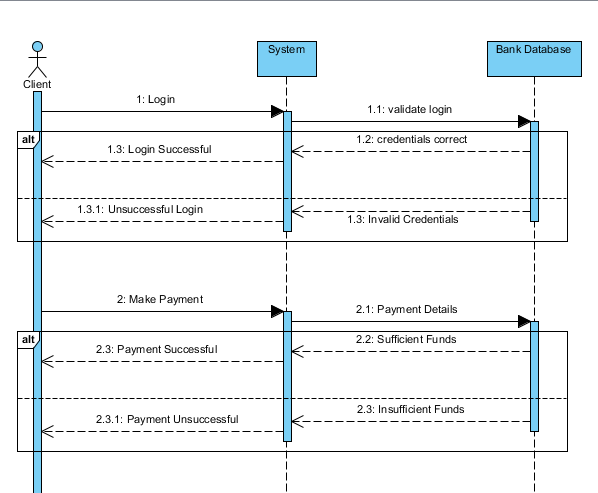


Figure 4. Client bill payment sequence diagram for RBS.

* The sequence diagram (**viewing client’s account statement by banker**) of the RBS is shown in Figure 5:

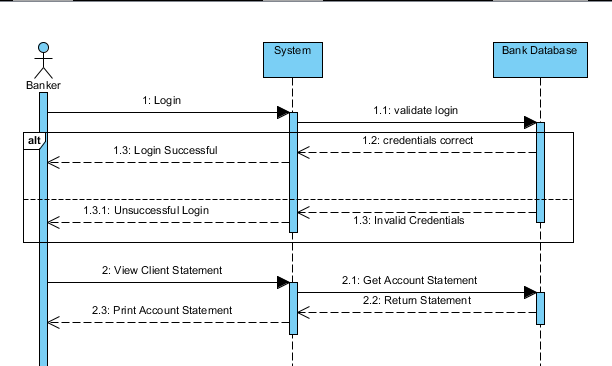


Figure 5. Account statement sequence diagram for RBS.

* The sequence diagram (**crash recovery by banker**) of the RBS is shown in Figure 6:

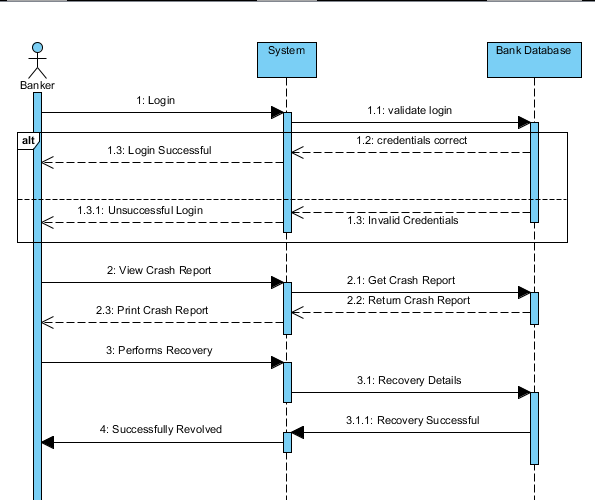


Figure 6. Crash recovery sequence diagram for RBS.

* The sequence diagram (**viewing of account statement by client**) of the RBS is shown in Figure 7:

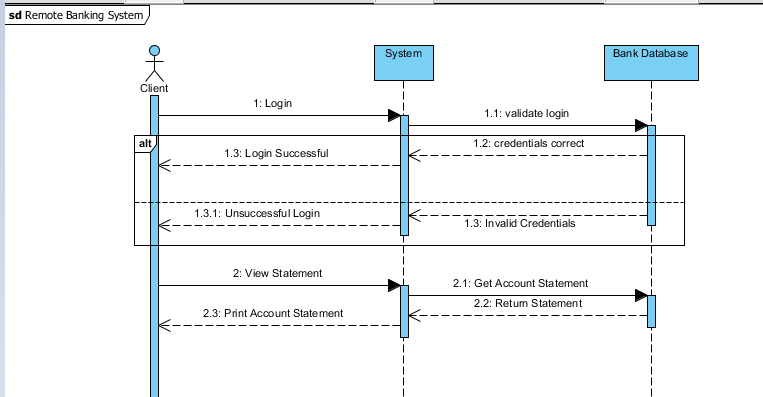


Figure 7. Account statement sequence diagram for RBS.

## 3.4 Entity Relationship Diagram

The ER is a high-level conceptual data model diagram. The Entity Relationship diagram is based on the notion of real-world entities and the relationship between then. According to the system requirements, the Entity Relationship Diagram of the RBS is shown in Figure 8:



Figure 8. Entity relationship diagram for RBS.

## 3.5 Data Flow Diagram

The data flow diagram represents a flow of a process in a system. The data flow diagrams also provide information about the outputs and inputs of each entity and process. Below are a number of dataflow diagrams for various processes that are carried out by the RBS.

* The Data Flow Diagram (**Banker Login**) of the RBS is shown in Figure 9:

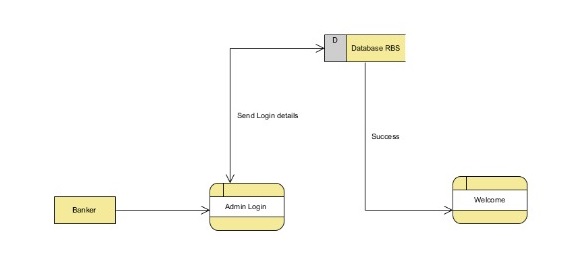
****

Figure 9. Banker login data flow diagram for RBS.

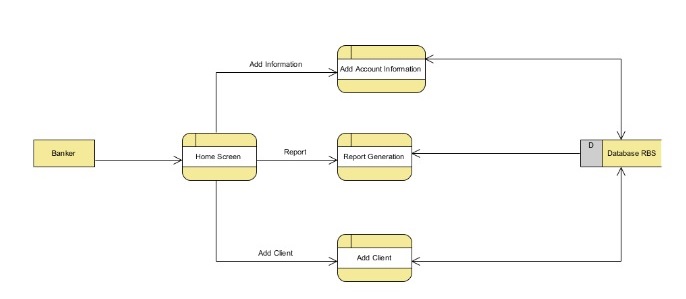
* The Data Flow Diagram (**Banker Operations**) of the RBS is shown in Figure 10:

Figure 10. Banker operations data flow diagram for RBS.

* The Data Flow Diagram (**Client Login**) of the RBS is shown in Figure 11:

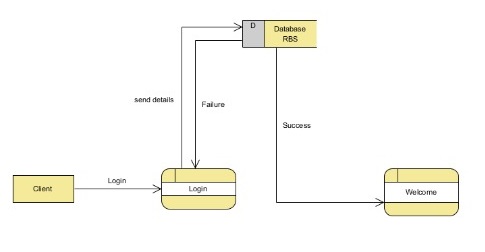


Figure 11. Client login data flow diagram.

* The Data Flow Diagram (**Client Operations**) of the RBS is shown in Figure 12:

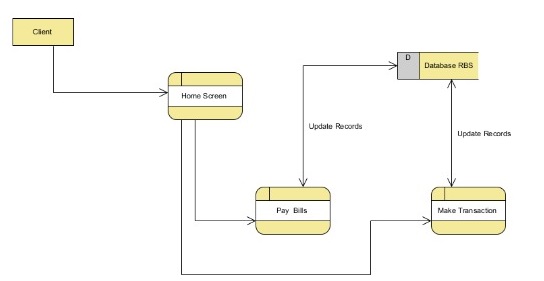


Figure 12. Client operations data flow diagram.

## 3.6 Component Diagram

The purpose of the component diagram is to depict the relationship between different components in the system, the Component Diagram of the RBS is shown in Figure 13:

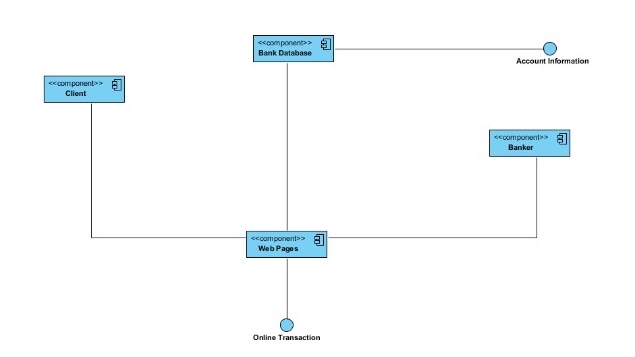


Figure 13. Component diagram of RBS.

## 3.7 Activity Diagram

An activity diagrams is basically a flowchart to represent the flow from one activity to the another.

* The Activity Diagram (**Banker**) of the RBS is shown in Figure 14:

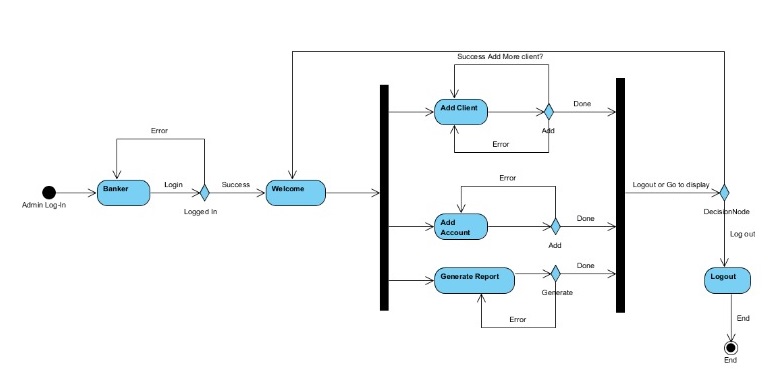


Figure 14. Banker activity diagram for RBS.

* The Activity Diagram (**Client**) of the RBS is shown in Figure 15:

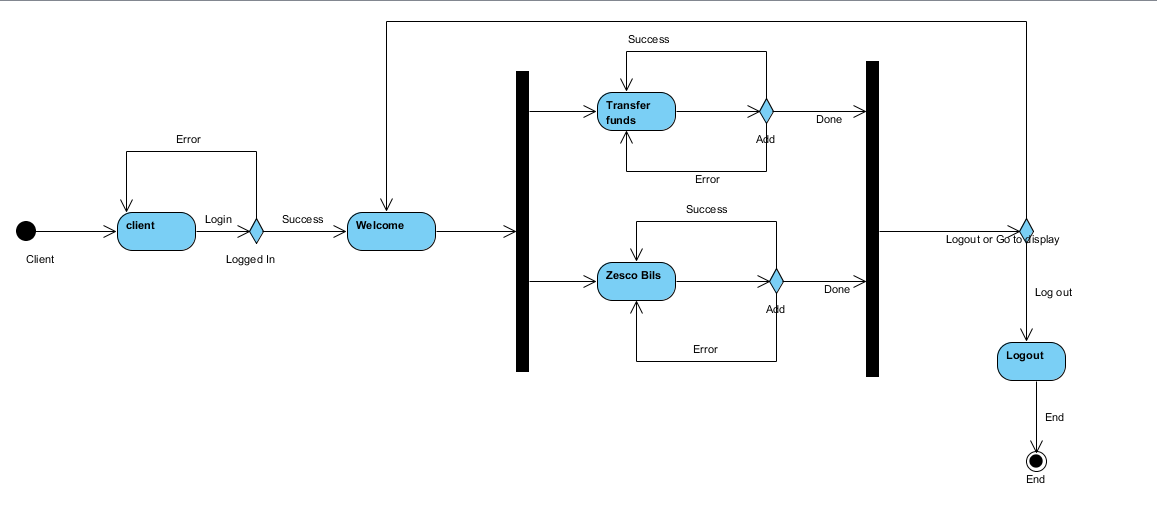


Figure 15. Client activity diagram for RBS.

## 3.8 Deployment Diagram

The deployment diagram simply shows the execution architecture of the system, including nodes such as hardware or software execution environment and the middleware connecting them.

* The deployment diagram of the RBS is shown in Figure 16:

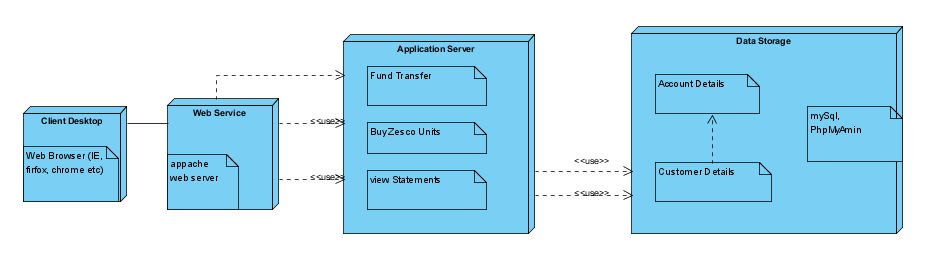


Figure 16. Deployment diagram for RBS.

# CHAPTER FOUR

## 4.1 Client’s View

The first page that the client sees upon loading the bank system URL is the login page that is shown in Figure 17:

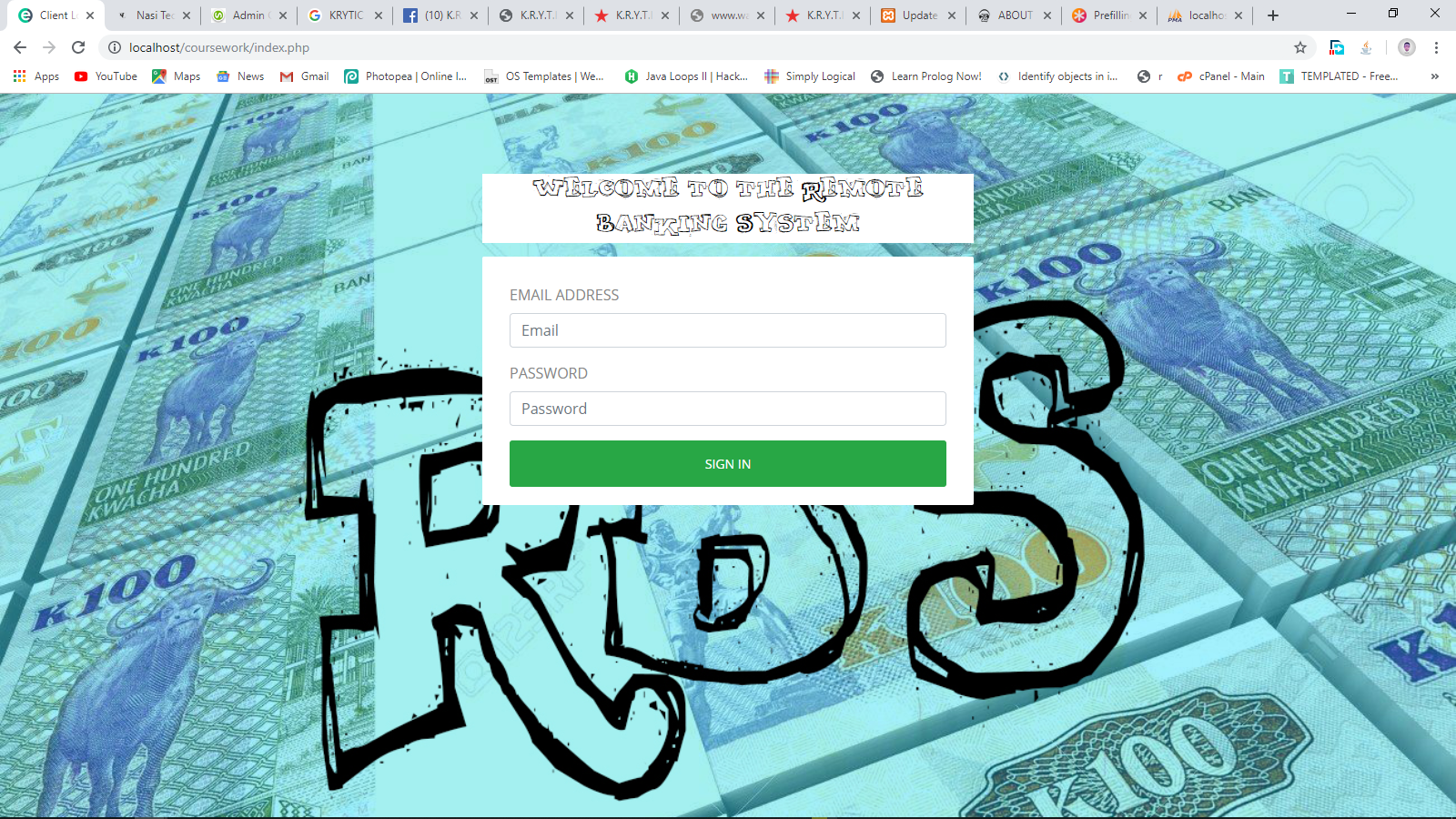


Figure 17. Client login page for RBS.

After the client has successfully logged in, Figure 18 shows the page that opens. For new clients, they will be notified that they have to change the default generated password that was given to them.

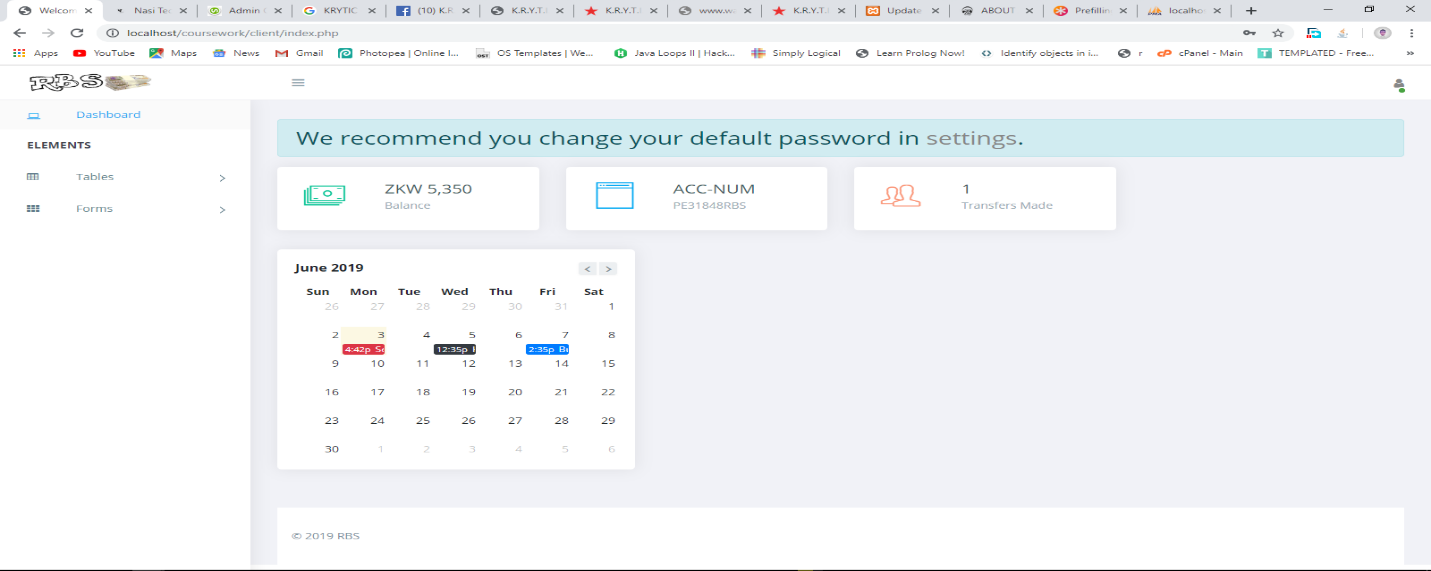


Figure 18. Client dashboard for RBS.

Once the client clicks on settings so that they can change their default password, Figure 19 shows the screen that appears:

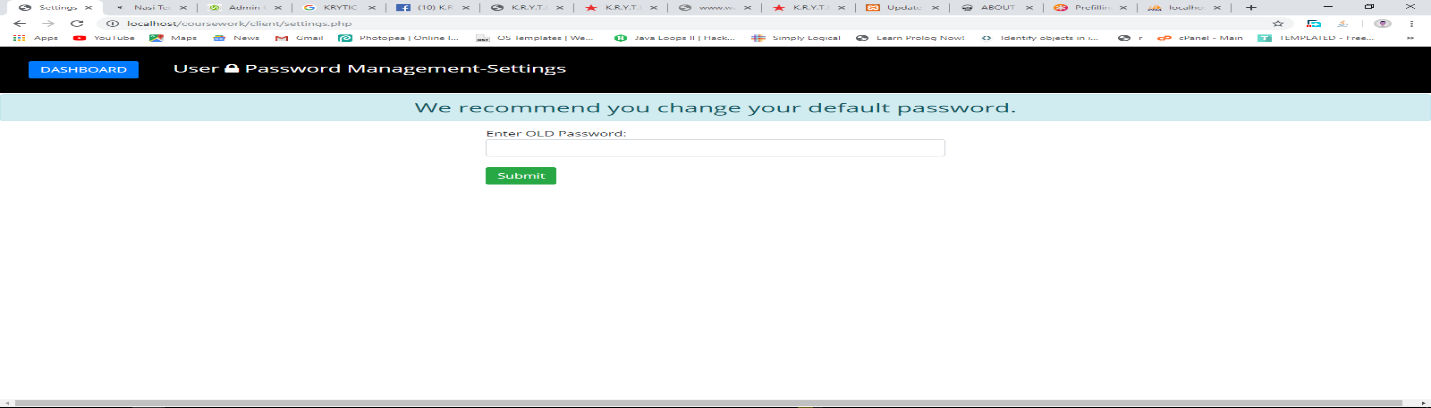


Figure 19. Client password management settings for RBS.

Fund and Zesco Bill Payment screen shown in Figure 20:

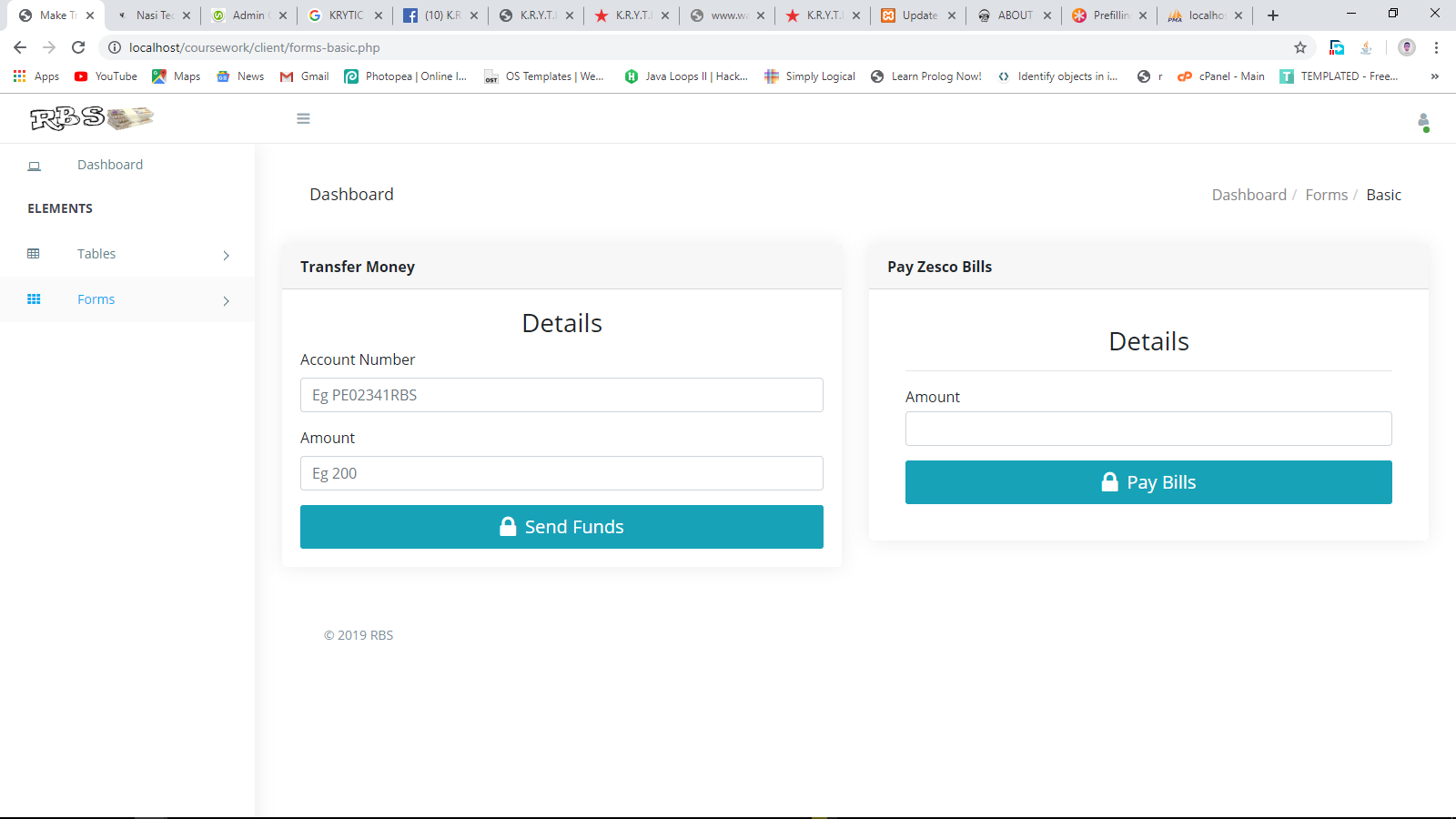


Figure 20. Fund and Zesco Bill Payment for RBS.

Upon a successfully completed transaction the user is notified as shown in Figure 21:

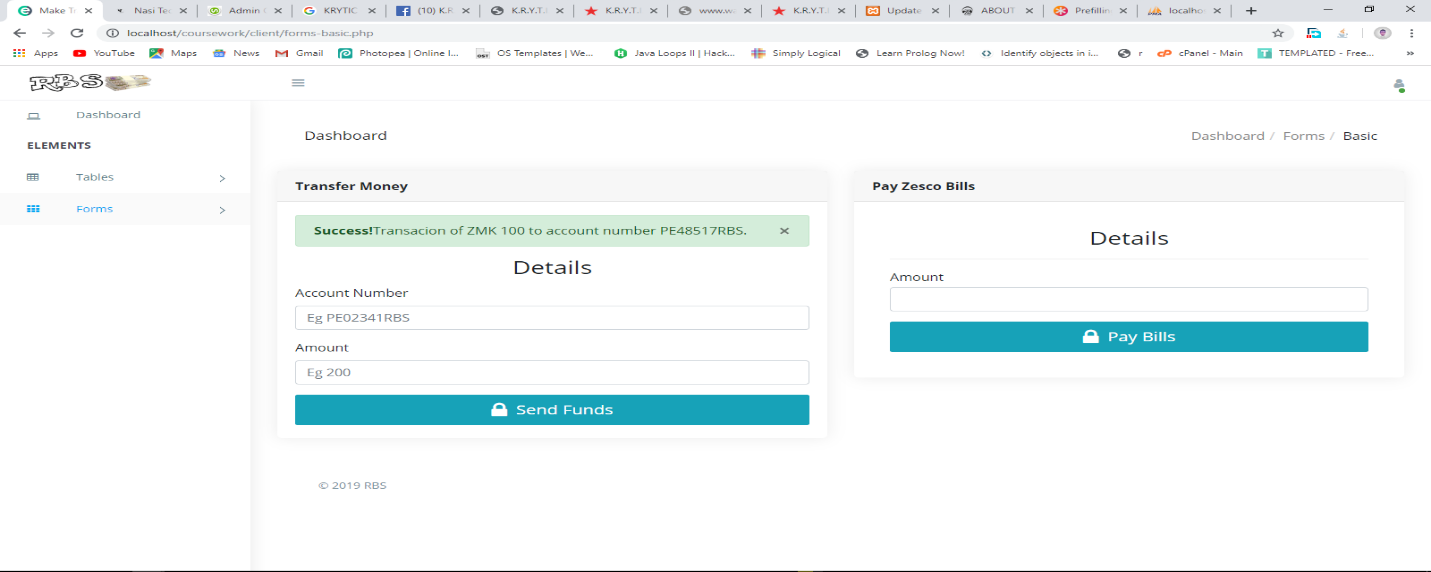


Figure 21. Successfully completed transaction for RBS.

In cases where the transaction fails when the user inputs an account number that doesn’t exist, the user will be notified as shown in Figure 22:

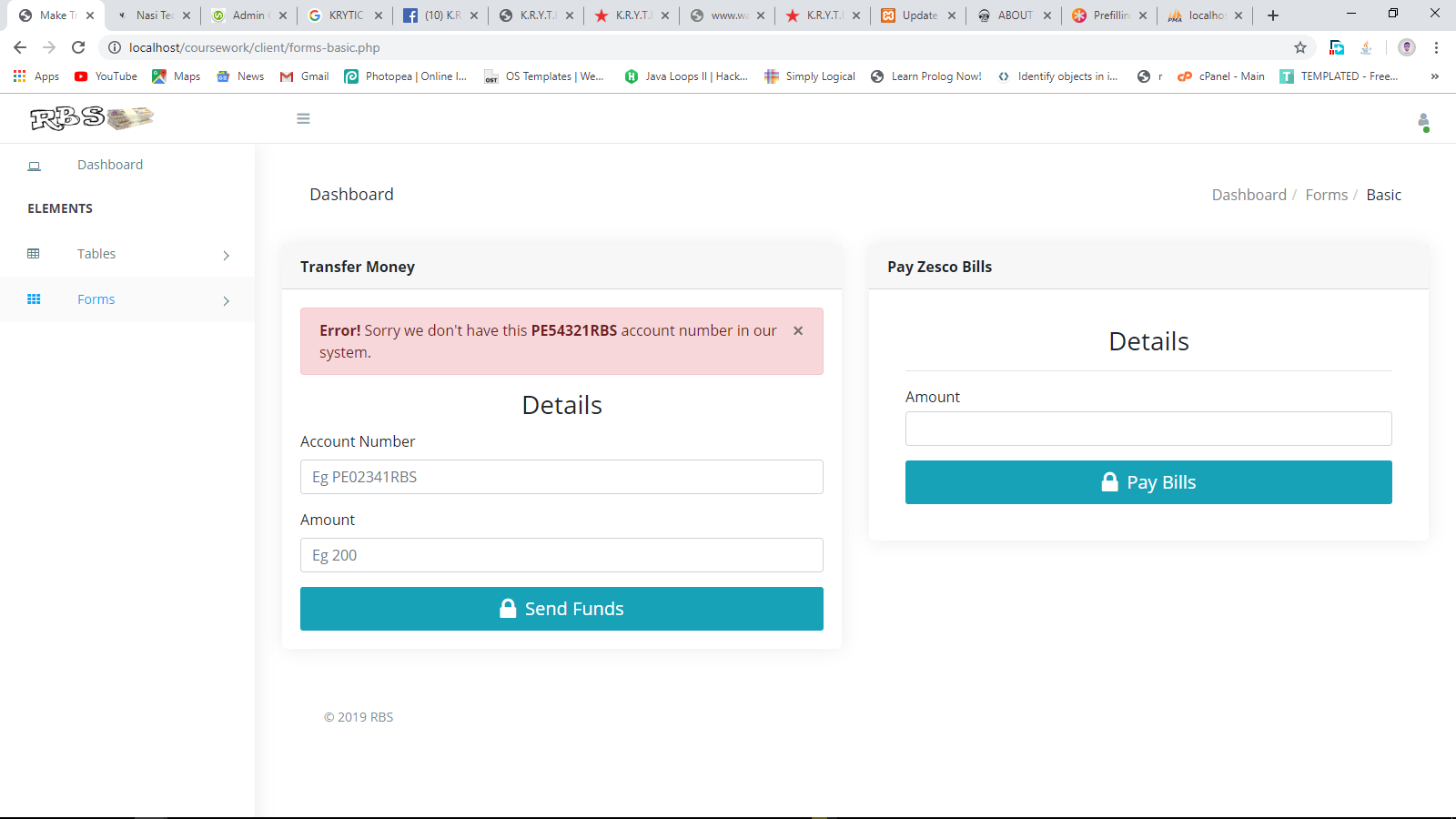


Figure 22. Un-successful transaction for RBS.

In case where the user tries to transfer more money than what one has in the account the system will decline the transaction as seen in Figure 23:

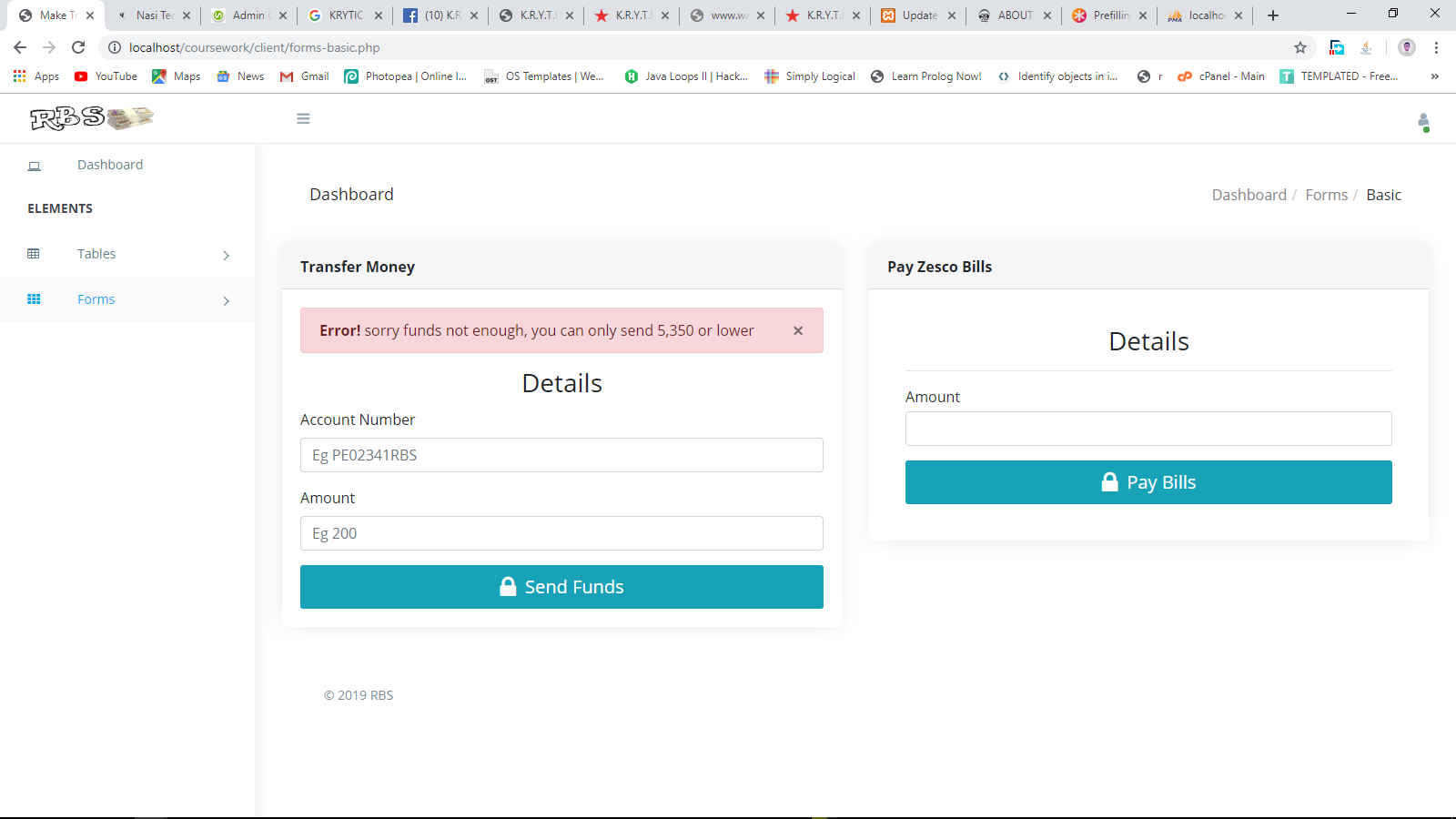


Figure 23. Un-successful transaction for RBS.

Client is able to view the transactions one has made. Figure 24 show client’s transaction table:

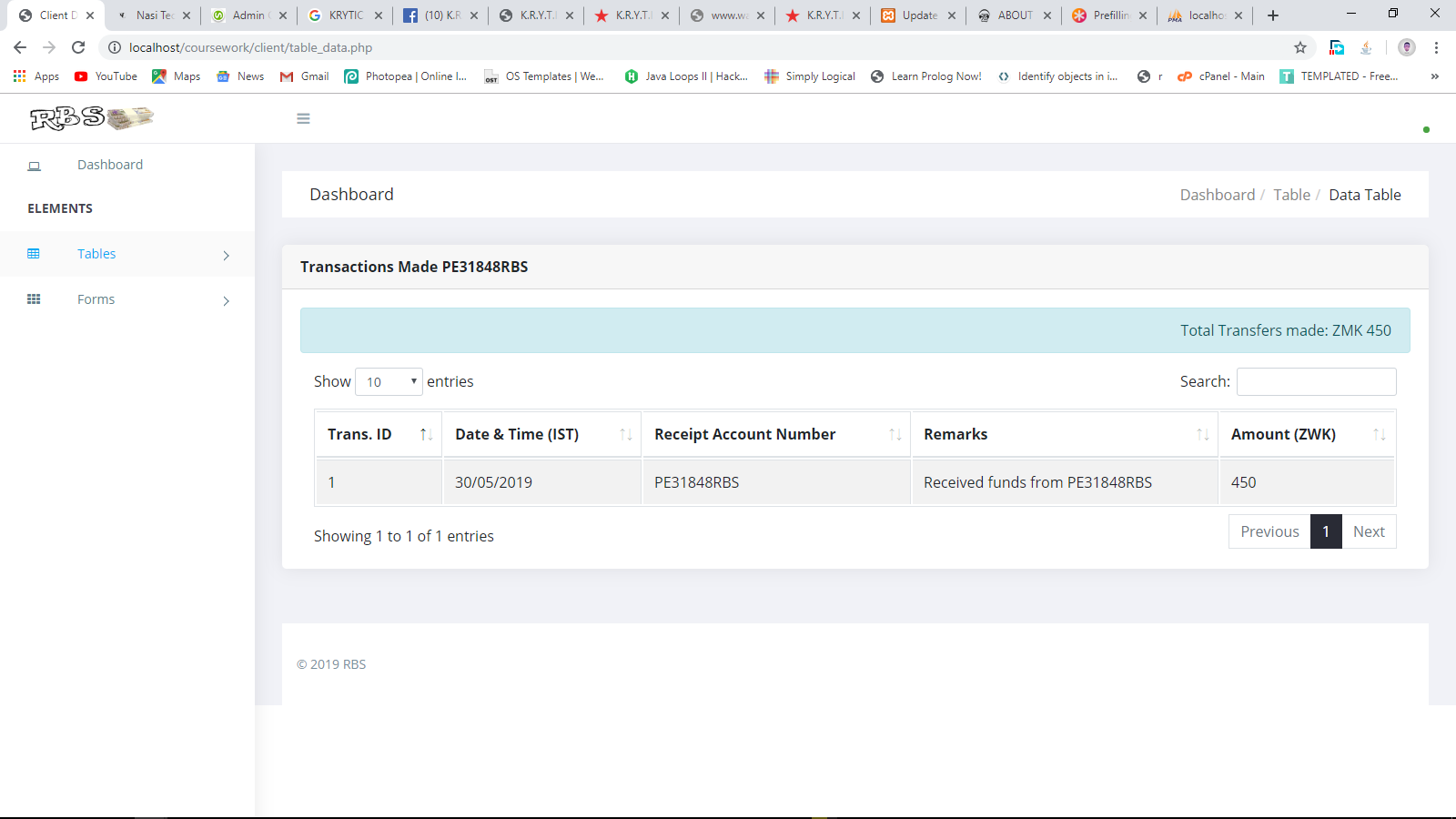


Figure 24. Client’s transaction table for RBS.

## 4.2 Banker’s View

The first page that the Banker sees upon loading the bank system admin URL is the login page that is shown in Figure 25:

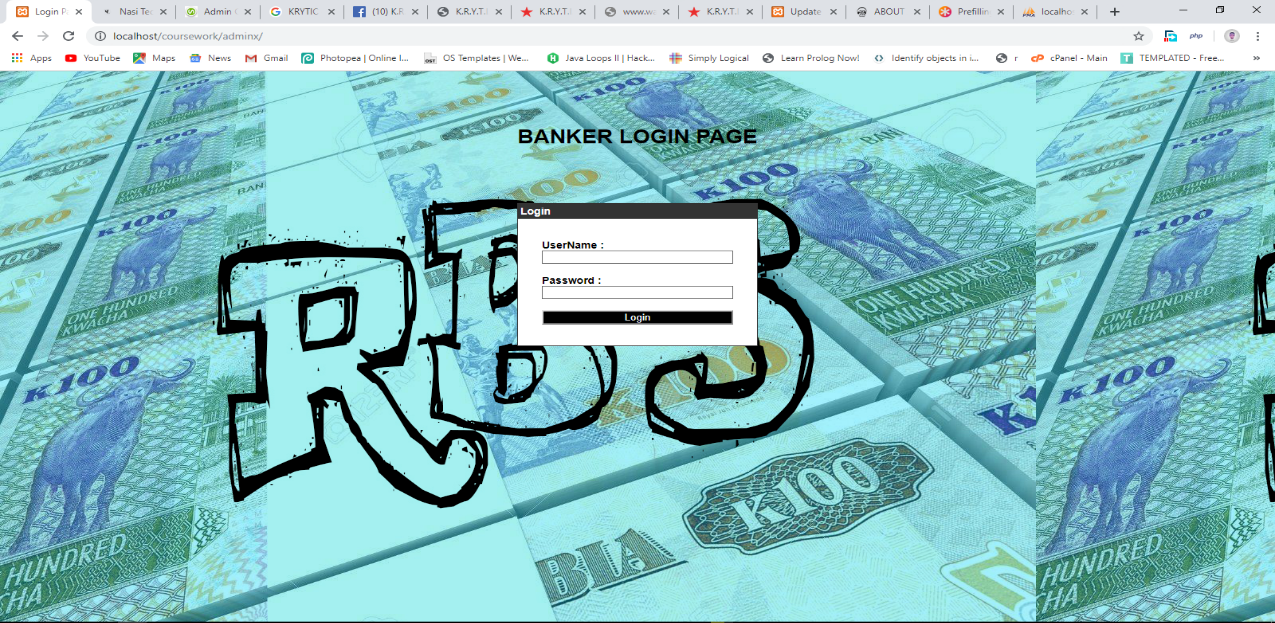


Figure 25. Banker login page for RBS.

After a successful login, Figure 26 shows the admin dashboard:

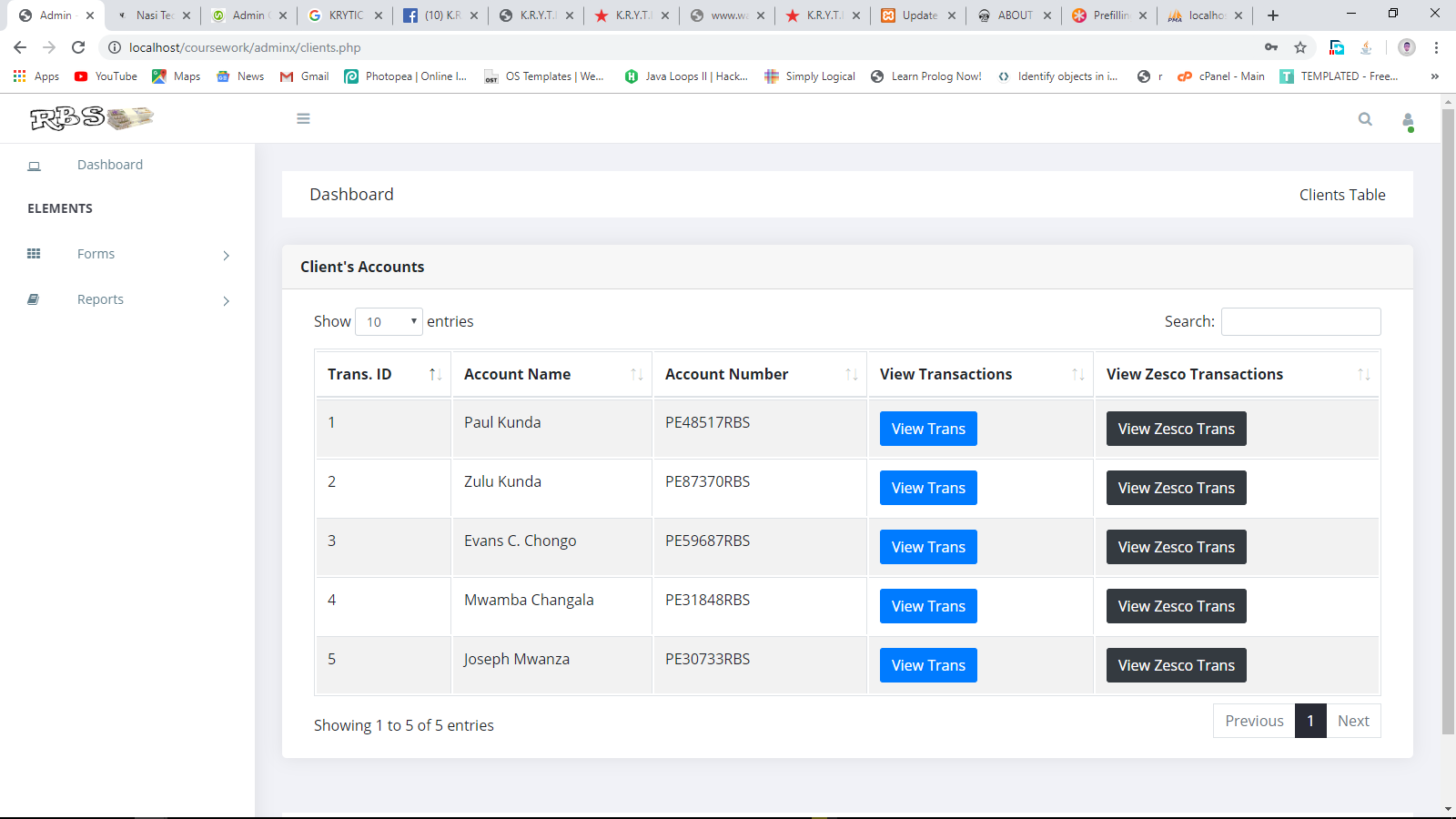


Figure 26. Banker dashboard for RBS.

The Banker will be able to add a bank account of a client and also a client bank system account for them to be able to do remote banking, as shown in Figure 27:

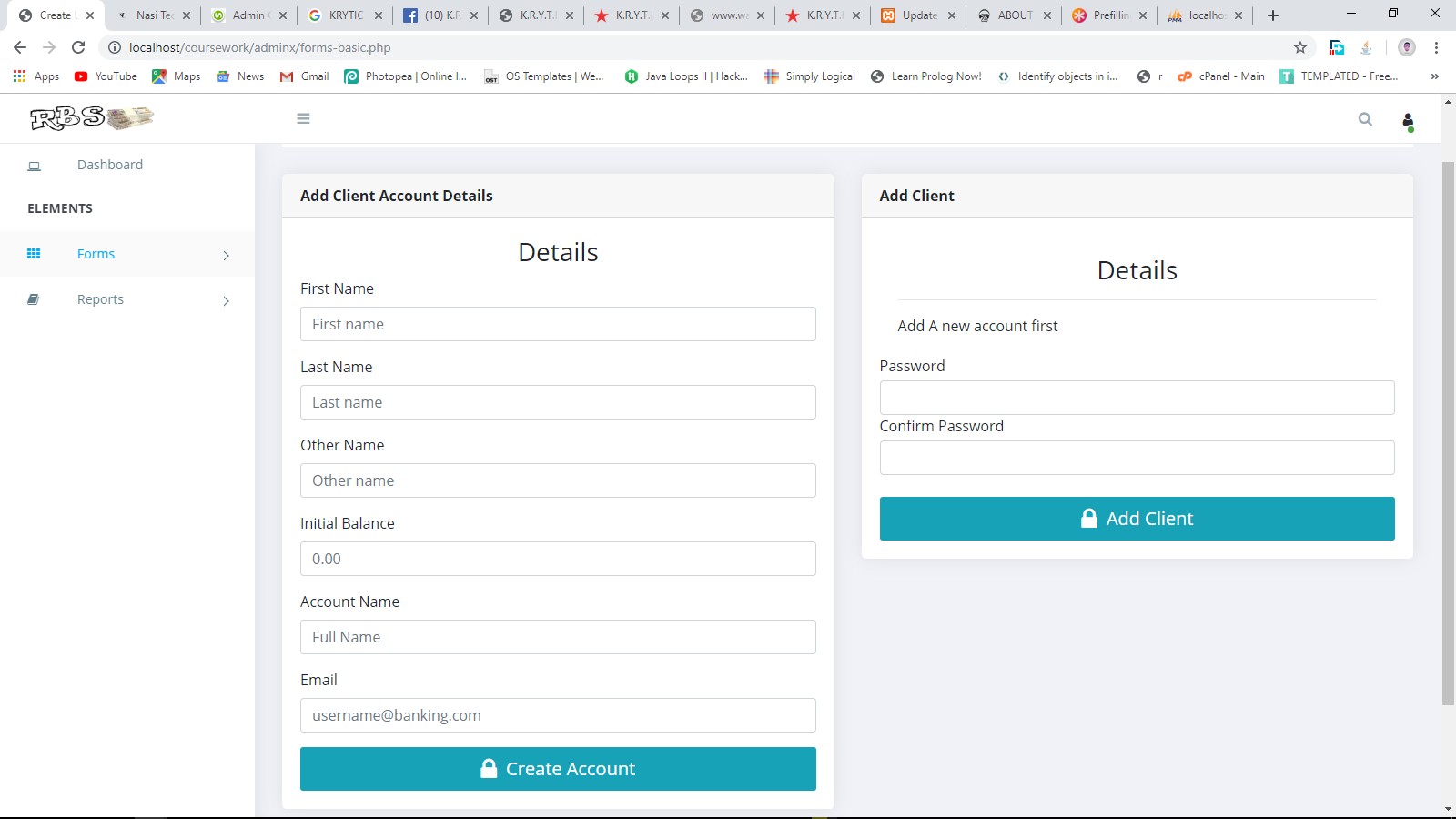


Figure 27. Add client details for RBS.

The banker is also able to generate reports as shown in Figure 28:

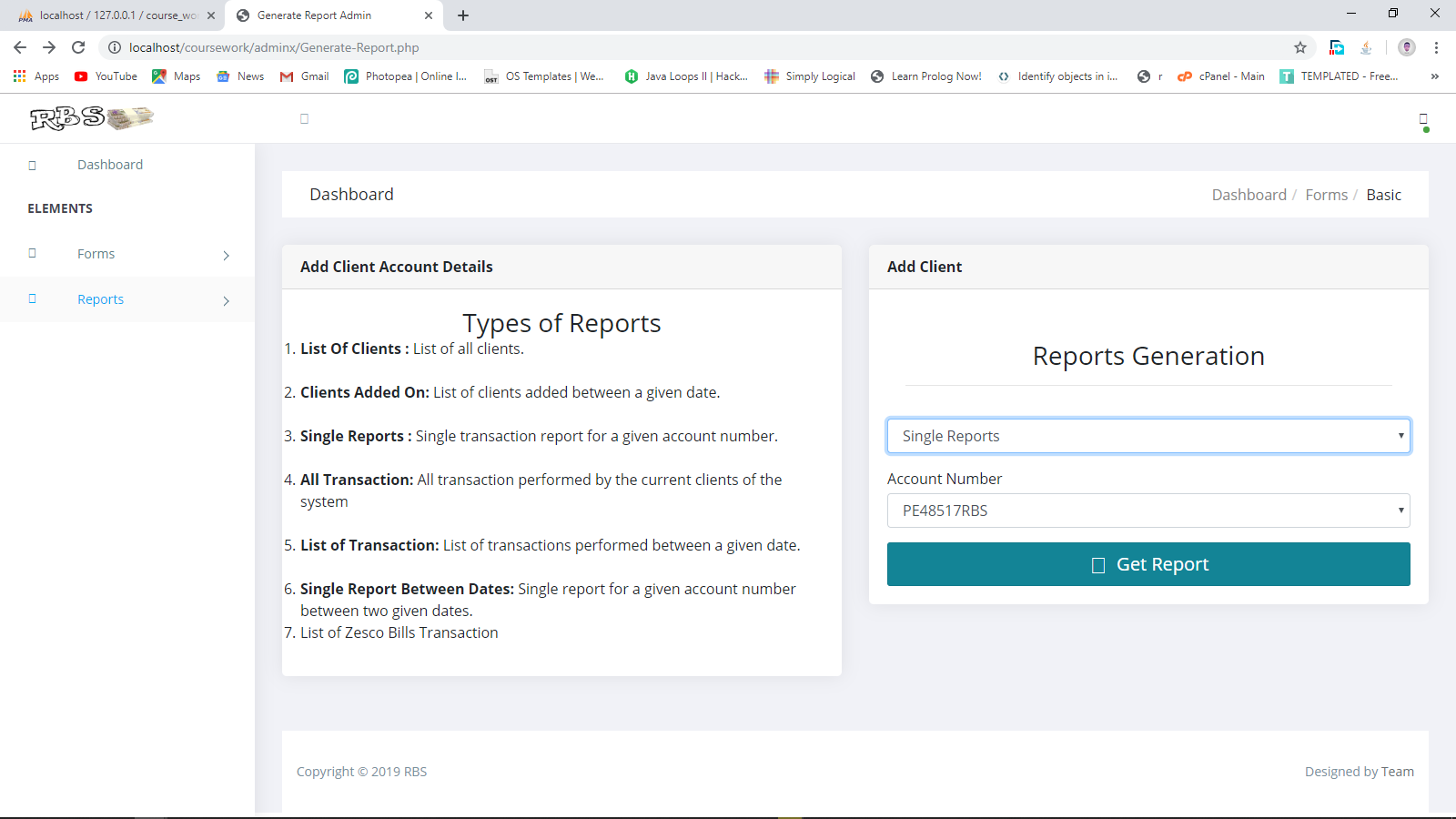


Figure 28. Banker report forms

Below are examples of how the reports will appear:

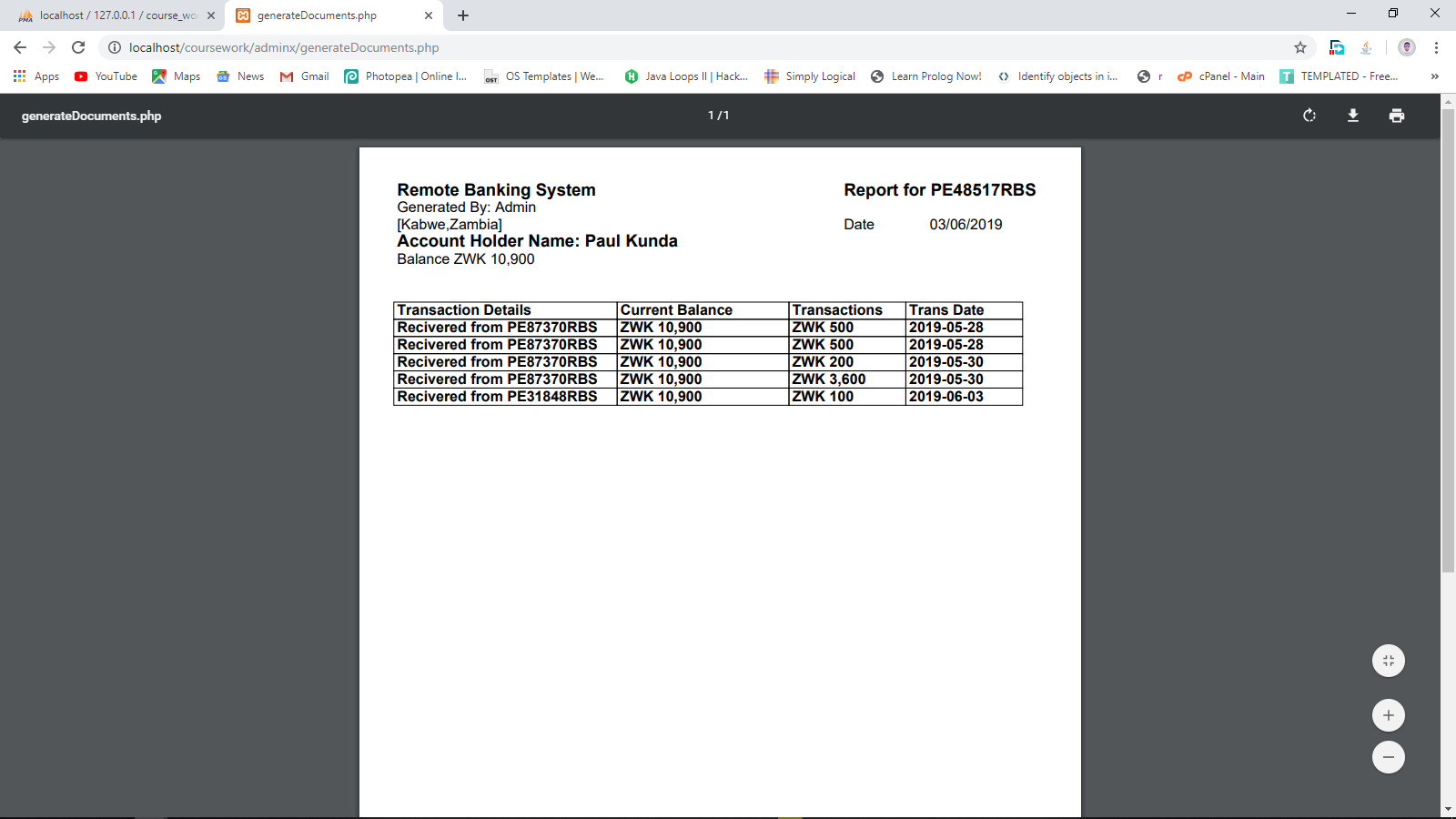


Figure 29. Report example 1

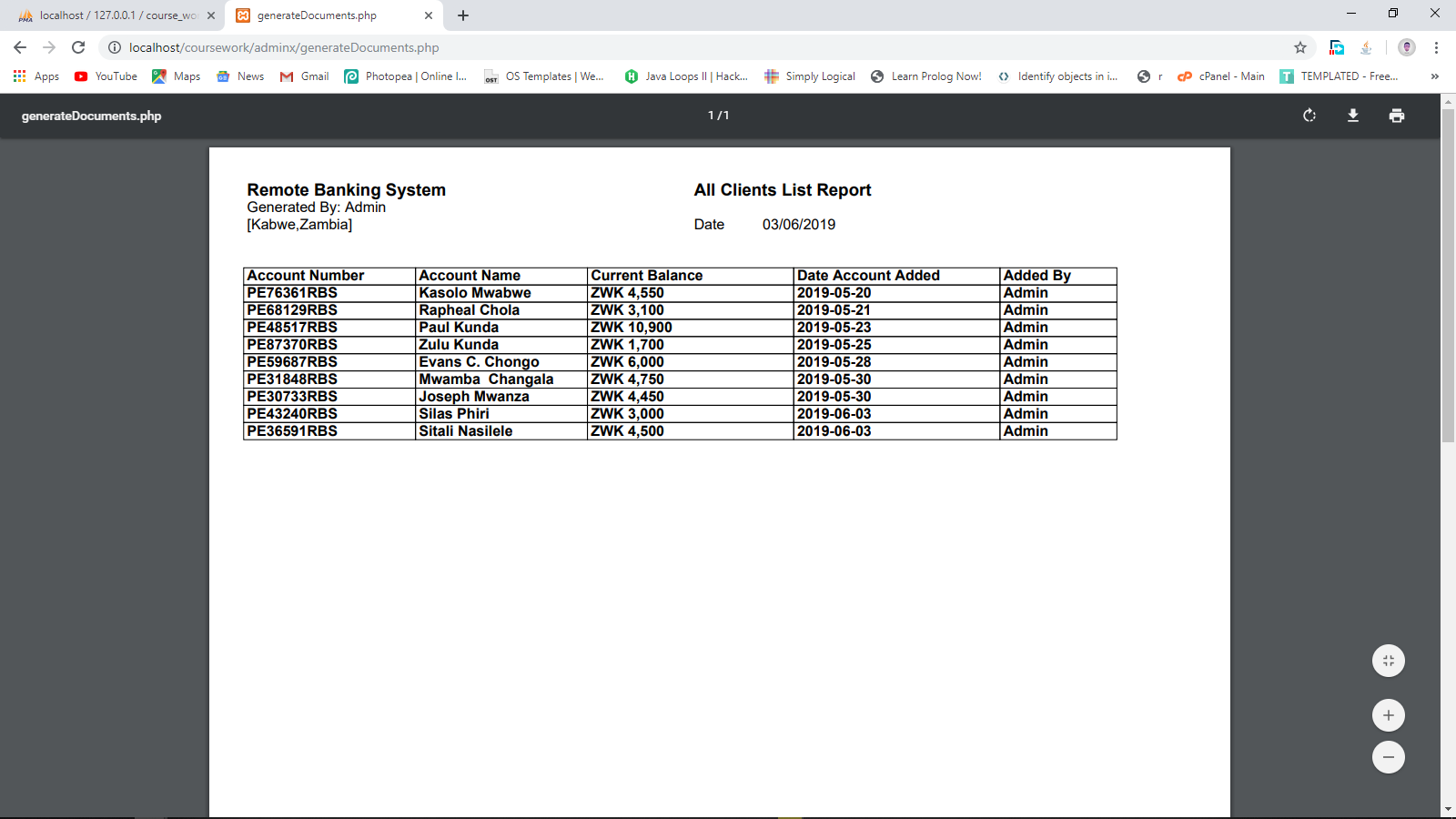


Figure 30. Report example 2

# References

* Bailey, J., & Pearson, S. (1983). Development of a Tool for Measuring and Analyzing Computer User Satisfaction. Management Science, 29(5), 530-545.
* Beckett, A., Hewer, P., & Howcroft, B. (2000). An exposition of consumer behaviour in the financial services industry. The International Journal of Bank Marketing, 18(1).
* Chin, J., Diehl, V., & Norman, L. (1988). Development of an instrument measuring user satisfaction of the human-computer interface. Paper presented at the Proceedings of the SIGCHI conference on Human factors in computing systems, New York.
* Chin, W., & Lee, M. (2000). A proposed model and measurement instrument for the formation of IS satisfaction: the case of end-user computing satisfaction. Paper presented at the Proceedings of the twenty first international conference on Information systems, Brisbane, Australia.
* Daniel, E. (1999). Provision of electronic banking in the UK and the Republic of Ireland. International Journal of Bank Marketing, 17(2), 72-82.
* DeVellis, R. (2003). Scale Development: theory and applications (2ed. Vol. 26). California: Sage Publications
* <https://www.slideshare.net/Jaydev_D_Kishnani/srs-for-banking-system> Slide share 2019, accessed 1st April 2019.
* <https://stackoverflow.com/questions/11826161/php-session-variables> Stack Overflow 2019, accessed on 20th April 2019
* <https://github.com/tncyrdmc/ONLINE-BANKING-SYSTEM-2> Github 2019, accessed 23rd April 2019
* <https://github.com/PialKanti/Online-Banking-System> Github 2019, accessed 24th April 2019