

**Software Requirements**

**Specification**

**for**

**Informatic System for Banking Transactions**

**Version 1.0**

**Prepared by Biraianu Alex-Valentin**

**Ungureanu Adrian-Ion**

**Gingeoveanu Stefan**

**Banica Mihai-Vlad**

**Becherescu Corina**

**Prepared for university project**

**12.03.2017**

# Table of Contents

[Table of Contents 1](#_1fob9te)

[Revision History 1](#_3znysh7)

[1. Introduction 2](#_tyjcwt)

[1.1 Purpose 2](#_3dy6vkm)

[1.2 Product Scope 2](#_1t3h5sf)

[1.3 Intended Audience and Reading Suggestions 2](#_4d34og8)

[1.4 Definitions, Acronyms, and Abbreviations 2](#_2s8eyo1)

[1.5 References 2](#_17dp8vu)

[2. Overall Description 3](#_26in1rg)

[2.1 Product Perspective 3](#_lnxbz9)

[2.2 Product Functions 3](#_35nkun2)

[2.3 User Classes and Characteristics 3](#_1ksv4uv)

[2.4 Operating Environment 3](#_44sinio)

[2.5 Design and Implementation Constraints 3](#_2jxsxqh)

[2.6 User Documentation 3](#_z337ya)

[2.7 Assumptions and Dependencies 4](#_1y810tw)

[3. External Interface Requirements 4](#_4i7ojhp)

[3.1 User Interfaces 4](#_2xcytpi)

[3.2 Hardware Interfaces 4](#_1ci93xb)

[3.3 Software Interfaces 4](#_3whwml4)

[3.4 Communications Interfaces 4](#_2bn6wsx)

[4. System Features 5](#_3as4poj)

[4.1 System Feature 1 5](#_1pxezwc)

[4.2 System Feature 2 (and so on) 5](#_49x2ik5)

[5. Other Nonfunctional Requirements 5](#_2p2csry)

[5.1 Performance Requirements 5](#_147n2zr)

[5.2 Safety Requirements 6](#_23ckvvd)

[5.3 Security Requirements 6](#_ihv636)

[5.4 Software Quality Attributes 6](#_32hioqz)

[5.5 Business Rules 6](#_1hmsyys)

[6. Other Requirements 6](#_41mghml)

[Appendix A: Glossary 6](#_2grqrue)

[Appendix B: Analysis Models 6](#_vx1227)

[Appendix C: To Be Determined List 7](#_1v1yuxt)

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Comments** | **Version** |
|  |  |  |  |
|  |  |  |  |
| Biraianu Alex-Valentin  Ungureanu Adrian  Gingoveanu Stefan  Banica Mihai-Vlad  Becherescu Corina | 12.03.2017 | First Revision | 1 |
|  |  |  |  |

# 

# 1. Introduction

## 1.1 Purpose

The purpose of this document is to present a detailed description of the Informatic System for a Banking application. It will explain the features of the system, the interfaces of the system and what the system will do.

## 1.2 Product Scope

The system will be built on a client/server architecture for assisting the bank on achieving the following tasks:

* Allowing users to see information about their own accounts
* Changing their personal information and password
* Making transactions and pay for different services
* Viewing their transaction history

## 1.3 Intended Audience and Reading Suggestions

This document is intended for both the bank and the developers of the system.

## 1.4 Definitions, Acronyms, and Abbreviations

## 1.5 References

Link to the app*:* [*https://github.com/UAdrian/banking-app*](https://github.com/UAdrian/banking-app)

Link to the license: https://github.com/UAdrian/banking-app/blob/master/information/LICENSE

# 2. Overall Description

## 2.1 Product Perspective

This product is a self-contained system intended for managing a bank and the accounts of its clients.

## 2.2 Product Functions

The product will have the following main functions:

* Viewing and modifying Accounts of users
* Transactions between different accounts and services
* A transcribed log containing all performed actions of the user

## 2.3 User Classes and Characteristics

The application users will be bank employees which will have administrator privileges and clients which will have regular accounts.

## 

## 2.4 Operating Environment

The application supports the following OS Windows.

## 2.5 Design and Implementation Constraints

The software will have a GNU General Public License.

## 2.6 User Documentation

The user will be given a user manual containing all the applications functions.

## 2.7 Assumptions and Dependencies

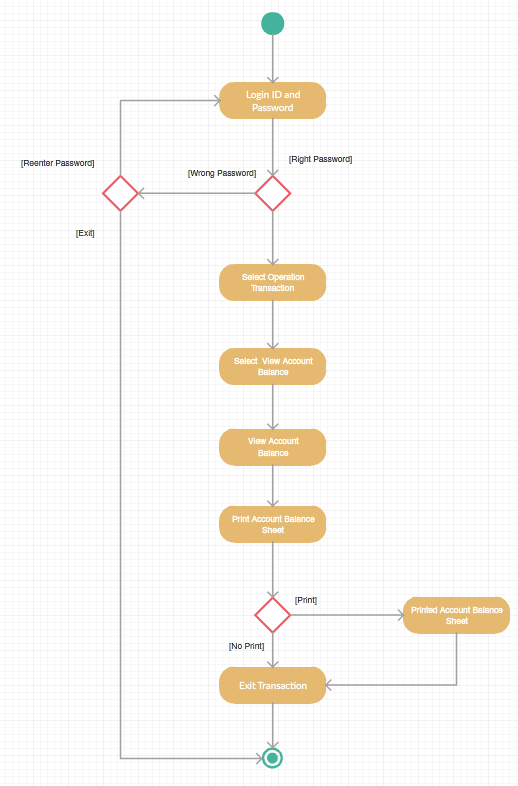
# 3. External Interface Requirements

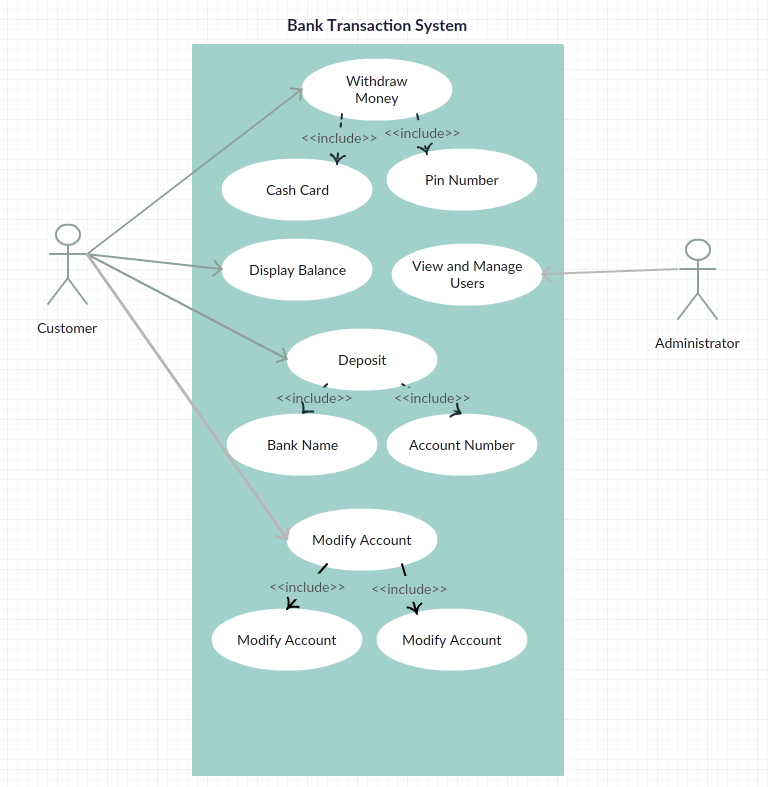
## User Interfaces

The interface will have the standard help, minimize and close buttons.

The log in interface will open in the middle of the screen.

After the user logs in this interface will close and the main app will open, this facilitates the operations between the user and the app.





## 3.2 Hardware Interfaces

Not applicable for this type of software.

## 3.3 Software Interfaces

Not applicable for this type of software.

## 3.4 Communications Interfaces

The application will have a server that will communicate between the app and it’s database and it will encrypt and decrypt the data from the database.

# 4. System Features

**4.1 General:**

* User Interface should be secure and accessed with user and password.
* The user and password of the client (pre-generated) are kept in database and are

accessed by the app.

* The data presented in the interface should be easy to understand and accessible to

the user.

* The user will be able to see account details, change personal information and

password, perform transaction to other accounts and special services and view his

transaction history, all accessible from the main body of the application.

* The accounts should be of different types depending on the currency.
* The user can download a copy of the transaction history as PDF.
* The system should contain at least the following modules : a module that extracts

the data from the database, a module that writes the data to the database and a

module that facilitates the transactions.

## 4.2 Database:

* Management of data for clients, accounts and the bank.
* The database should allow:
  + Data recovery at a specified time.
  + Support of offline backup.
  + Administration of data.
  + Users to access information concurrently.
  + Definitions of diferite types of users with different access rights.
  + Monitoring all types of queries and implementing complex data structures.
* Should have a ‘log’ for the activities performed.

**4.3** **Stand-alone application:**

* It should facilitate the login of the users, access information about the account and

manage transactions.

* The Login screen is a graphical interface containing two box fields for the user and

password, ’login’ button and a ’forgot password’ button.

* The password box field is case sensitive and conceals the password.
* Pressing the login button queries the database.
* If they are correct it leads to the account page else an error message appears declaring

either one or both of the fields maybe incorrect .

* The Login screen is in the middle of the window.
* The Account Page is the main body of the application:
  + It greats the user with a welcome message and notifications about the account.
  + It contains a menu on the left alowing access to the different sections.
  + In the middle there are some basic informations about the bank accounts the user

has access to.

* On the top there are tabs that switch from the overview of all acounts to specific

accounts (saving accounts, loans etc.)

**4.4 Web server:**

* Facilitates the contact between the stand-alone application and the database.

# 5. Other Nonfunctional Requirements

## 5.1 Performance Requirements

It requires a pc with at least a microprocessor.

## 5.2 Safety Requirements

Not applicable for this software.

## 5.3 Security Requirements

## 5.4 Software Quality Attributes

The software respects the rules of usability, testability and maintainability.

## 5.5 Business Rules

Disclaimer this product is made for the intent use as a banking application for administrators of a bank and clients of said bank not for any other purpose or intent. Using this application for any other purpose invalidates the agreement between the software makers and users .

# 6. Other Requirements

The Applications language will be English.

# Appendix A: Glossary

# Appendix B: Analysis Models

The Analysis model shall be waterfall model.

# Appendix C: To Be Determined List