–

**THE BANK MANAGEMENT SYSTEM**

**Technical report documentation for the “THE BANK MANAGEMENT SYSTEM” Desktop application This document outlines key technical details of the application. Content includes research, requirements, use cases, data requirements, analysis and design, testing, usability and class diagrams**

Chapter 1\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\*

**THE BANK MANAGEMENT SYSTEM**

1.1 Introduction

1.2 Background

1.3 Scope

1.4 Technology/resources

Chapter 2\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\*

**Requirements Specification**

2.1 Product function

2.2 PRODUCT PERSPECTIVE

2.3 USER CHARACTERISTICS

2.4 Un-register user

2.5 Register user

2.6 Data requirements

2.7 User

2.8 Create an account

2.9 Login

2.10 non functional requirement

Chapter 3\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\*

**SYSTEM ARCHITECTURE**

3.1 OPERATING ENVIRONMENT

3.2 GENERAL CONSTRAINTS

3.3 TARGET AUDIENCE

Chapter 4\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\*

**DESIGN PHASE**

4.1 Use case Index

4.2 Use case Diagram

Chapter 5\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\*

**TESTING & USABILITY**

5.1 Test scenario

5.2 Think aloud test

5.3 Think aloud tasks

**TITLE: - THE BANK MANAGEMENT SYSTEM**

PARTICIPANT

PROJECT MADE BY

**MUZAMMIL CHANNA**

**BSE-20S-026**

**Malcolm.channa@gmail.com**

**03091255471**

DOCUMENTATION MADE BY

**ZUBAIR AHMED**

**BSE-20S-042**

**BSE20S042@stu.smiu,edu.pk**

**03163820959**

**CHAPTER 1**

**THE BANK MANAGEMENT SYSTEM**

**INRODUCTION**

**Introduction**

**The bank management system is fast money transaction system this application based on desktop application with lots of feature like cash deposite, cash withdrawl, fast cash,balance inquary and also pin change features .with the help of this application user can check his bank balance,Account detail,transfer money from anywhere with the help of this Desktop Application**

**Background**

**n**

**The Bank Managmnet System is easy to use and more friendly with user.User just create aacount and access the advance feature of this Desktop application.User can transfer his/her founds/money with the fastest service,This application allows the user to check his balance and account detail any time anywhere**

**Scope**

**n**

**The bank management system provide strong scurity for his user.The user data will be secured and not sharing with others only user can check his current account status online/offline activity and balance inquary,account detail.when user transfer cash or deposite cash or performed any task on the application the application requried user pin to performed user givin task.**

**TECHNOLOGIES/RESOURCES**

**n**

|  |  |
| --- | --- |
| Resources | Description |
| Windows Host Machine | Developing an Desktop applications requires the Java Integrated Development Environment |
| Neat beans appache | Project created in appache neat bean |

**CHAPTER 2**

**REQUIREMENTS SPECIFICATION**

**PRODUCT PERSPECTIVE**

**n**

.

**The Application will be written in Java using neat beans. It will run on desktop, Open-source platform. It will provide users with the ability to check his account detail etc inside the application. Users will be able to register an account and log in using their details to take part in an community and gain access to user submitted content**

**PRODUCT FUNCTIONS**

**n**

* **Register user will be able to access thier account from anywhere throw this desktop application username and pin code required for login.**
* **Unrigester user will be apply for the rigistration with their personal details, be a part of our community**
* **User will be abble to transfer their money**
* **User will be able to get account detail any time**
* **User will be abble to change their pin.**
* **User will be abble to Withdrawl cash.**
* **User will be abble to deposite cash.**
* **User will be able to check balance inquary.**
* **User will be able check mini bank statement.**

**USER CHARACTERISTICS**

**n**

:

UNREGISTERED USER

**Unrigisteruser will not be able to gain access to the application or use any of the its features.**

**Becouse of our community provide strong privacy to their user so unrigester user can not be able to access any of the application features,main reason is that the application based on banking system so, community do not take any of the risk registerition required strongly.**

**REGISTERED USER**

**Registered users will gain full, non admin access to the application. Once a user registers their details and agrees to the terms and condition they will able to check their account detail,balance detail etc**

**DATA REQUIREMENTS**

**n**

**The data required for the registration**

**FUNCTIONAL REQUIREMENTS**

**n**

**USER**

**All users of the Application shall have the ability to create an account which is used to store user data. User registration and login shall be mandatory.**

**CREATE AN ACCOUNT**

**n**

**The system should provide the user with an easy to use application to facilitate their creating an account. The system shall ask for an user`s full name, father name, gender, marital status and Email address home address city, pin code, status• The system shall notify the user if incorrect characters are used in the email or password fields. • The system should notify the user if their email has already been used. • The system should notify the user if any required fields are left empty. The system should prevent the user from completing registration if the term of service has not been agreed.**

**Login**

**The system should provide a user-friendly graphic user interface to allow the user to login their account in giving field user name and pin code to access their account for further detail, the system should notify the user if submitted information is incorrect**

.

**User**

User name xyz

Pin code \*\*\*\*

**next**

**Create account**

Enter user name

Enter user password

**Tap next for proccede next step**

**If user don`t have any account create an account for login**

Login sample page

Login page sample

**Cash deposite Transaction**

**Fast cash Pin change**

**Withdrawl cash Balance inquary**

**bank mini statement account detail**

**Transaction**

**NEXT**

**ENTER PIN CODE**

**\*\*\*\***

**Process …..**

**NON-FUNCTIONAL REQUIREMENTS**

**PORTABILITY**

**The application will be written in JAVA using the JDK and JRE in appache neat beans , the application will run natively on any desktop pc laptop running windows 7,8,10.**

**RELIABILITY**

**The system should be extremely reliable and have an approximate up time of 99.999%. In the event of a crash or any other error, the System should inform the user of any problems and gracefully terminate.**

**The application should be user friendly and intuitive to use. GUIs should make their functions clear and navigation around the application should be straight forward.**

**• Users should be comfortable using the application after 20 minutes of use.**

**Ease of Use**

**SPEED**

**The application should open and be ready to use within 10 seconds of being selected. The system should be quick and smooth with no delays**

**All database reads/writes should take no longer than 5 seconds. If the database encounters any errors, a user-friendly warning should be displayed to the user.**

**PRIVACY All data retained by the system will be stored in accordance with the Data Protection Act 1988 and the Data Protection (Amendment) Act 2003.**

**CHAPTER 3**

**SYSTEM ARCHITECTURE**

**OPERATING ENVIRONMENT**

**The application is designed to work on all kind of pc or laptop that are running windows operating system 7,8,10. User can easily install this application in their pc.it will required storage for installtion.minimum 1 gb ram required for using this application**

**GENERAL CONSTRAINTS**

* **This application required desktop pc or laptop with portable keyboard and mouse for desktop, for laptop attached keyboard and mouse is ok for this application**
* **The ability of the software to interface with database must be implemented**
* **This application required internet connection**

**TARGET AUDIENCE**

**The bank management system aim is to provide better security and fast and safe transaction and friendly –user enviroment. User data will be safe and keep private even our member will not be able to seen user`s personal data without user permission.**

**User account setting(transation,pin-code, withdrawl,balance history,etc) will stored in server for provideing better experince and service to the user.**

**If user wants balance history or account detail. Application will be abble to provide it to the owner of the account at the time.**

**The only restriction on the applications target audience is that users will need to be 18 years of age or older. users will need to agree to the Terms of Service when registering to use the application.**

**CHAPTER 4**

**DESING PHASE**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Use case Id | Use case Name | Primary Actor | Scope | Complexity | Priority |
| 1 | Create Account | Unregister user | In | Medium | High |
| 2 | Login | Register user | In | Medium | High |
| 3 | Submit command | Register user | In | Medium | High |
| 4 | Set value | Register user | In | Medium | High |
| 5 | Pin code | Register user | In | Medium | High |
| 6 | Edit pin code | Register user | In | Medium | High |
| 7 | balance history | Register user | In | Medium | High |
|  |  |  |  |  |  |

**UML class diagram**

accountdetail();

Pincode();

Mainframe();

Withdrawalcash();

Amount();

Pincode();

Mainframe();

HOME

activeUser();

|  |
| --- |
| Withdrawalcash();  Transaction();  Deposit();  Codechange();  Fastcash()  Balanceinquairy();  Ministatment();  Pincode(); |

HOME UI CONTROLL

**USE CASE INDEX**

ministatmnt();

Pincode();

Mainframe();

Codechange();

pincode();

newpincode();

Mainframe();

balanceinquiry ();

Pincode();

Mainframe();

transaction ();

Amount();

Pincode();

Mainframe();

deposit();

Amount();

Pincode();

Mainframe();

fastlcash();

Amount();

Pincode();

Mainframe();

|  |
| --- |
| Login/Register |
| Username:string  Pin code :string |
| +RegisterUser();  +loginUser();  +mainFrame(); |

Withdrawalcash();

Amount();

Pincode();

Withdrawalcash();

Amount();

Pincode();

Withdrawalcash();

Amount();

Pincode();

Withdrawalcash();

Amount();

Pincode();

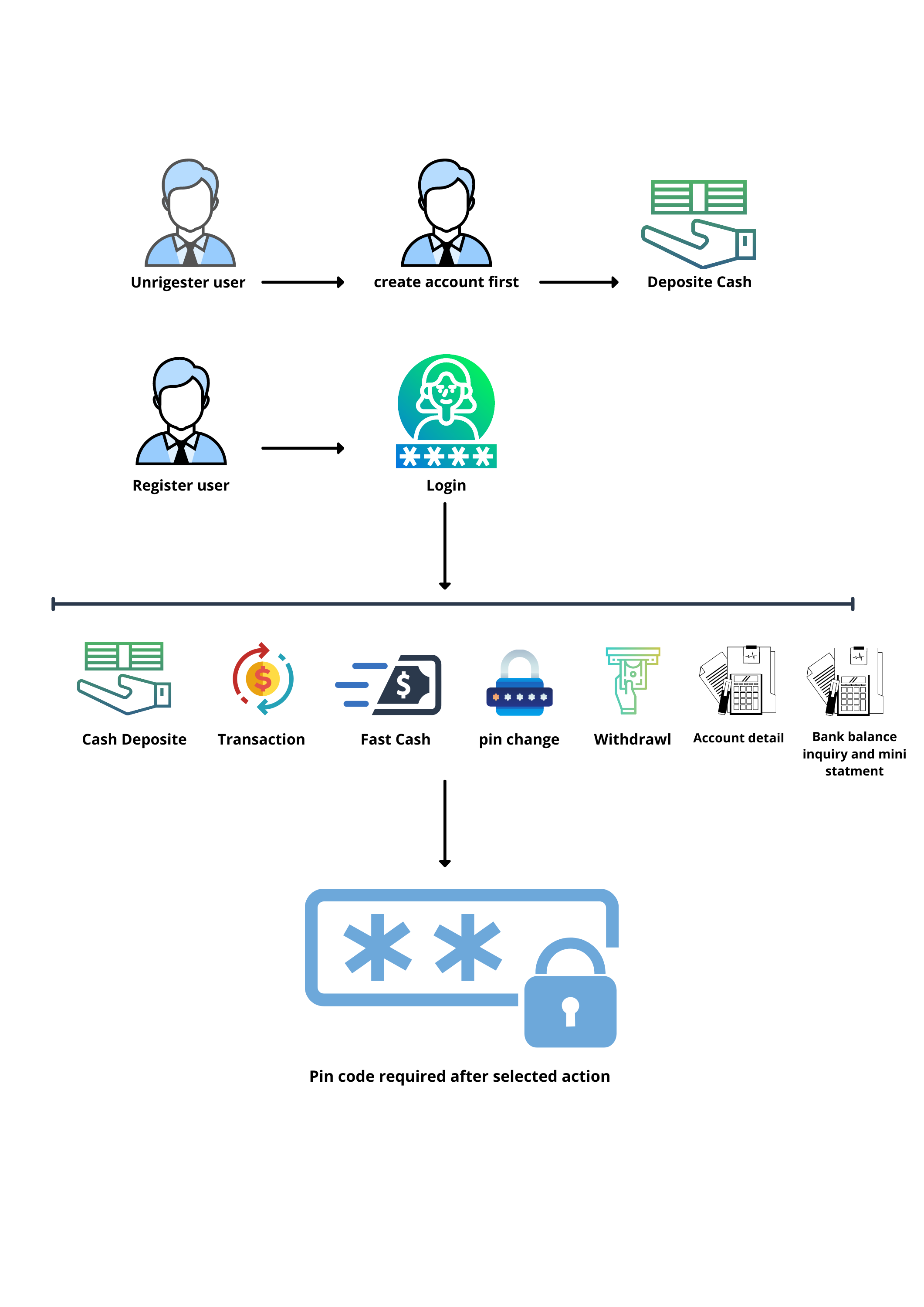
Withdrawalcash();

Amount();

Pincode();

Login/register ui controll

**USE CASE Diagram**



**CHAPTER 4**

**TESTING & USABILITY**

**Test scenario**

**The goal of this test was to establish if the application’s design and key visual properties convey to the user the purpose of the application. From the 5 second test the user should be able to answer three crucial questions:**

**Who are you?**

**What product or service do you provide?**

**Why should I care (What is in it for me)?**

**To establish whether the application makes this information clear or not, several users took part in a 5 second test and were asked three questions once the time limit expired. What do you think this application does?**

**What do you like most about this application? What do you like least about this application?**

|  |  |  |
| --- | --- | --- |
| What do you think this application does? | What do you like most about this application? | Have you feel secured with this application |
| “Fast and safe transaction” | “simple and easy to use” | “yes, the developer provide the best and secured environment its very good ” |
| “pin code change facility” | “no need high end performance pc or laptop simply we can use in low end level pc” | “not sure ” |
| “balance history” | “Application section is very clear to understand ” | “yes, as compare to privacy policy this application totally secured also developer said the personal user data keep safe from others and himself ” |
| “cash deposit for the safety” | “no worry about to remember the balance because the management provide us balance history at the time ” | “have no idea” |

**THINK ALOUD TEST**

**The think aloud tests provides a wealth of valuable information as it demonstrates a user’s thought process when interacting with an application. The think aloud test makes it possible to identify issues before a product goes live. A list of the most common functionality in the application was compiled and added to the think aloud test. Using this list of functionalities, questions were drawn up to evaluate the design and layout of the application.**

**2.1. Think Aloud Tasks**

**• Create an account**

**• Login**

**• Deposit cash**

**• Withdrawal cash**

**• account detail**

**• Browse balance history**

|  |  |
| --- | --- |
| Task | Result |
| Create an Account | Without hesitation the testers successfully created an account. Required steps were taken quickly and efficiently |
| Login | Without hesitation testers successfully logged in quickly with no issues. |
| Deposit cash | taster deposit cash without any hesitation |
| Withdrawal cash | tester withdrawal his cash without any problem and hesitation |
| Account detail | tester check their account detail |
| Browse balance history | Tester check their account balance history what they deposit, withdrawal when and where with the time and date proof without facing any problem |