**PROJECT REPORT**

**ONLINE E-HOSPITAL**

**Report submitted fulfilment of the requirements for**

**The award of the degree in**

**BACHELOR OF COMPUTER APPLICATION**

**BENGALURU NORTH UNIVERSITY**



**PRESENTED BY**

KEERTHAN K N (R1918048)

**Under the Guidance** **of:**

**MS.KOUNIN**

**Dept. of Bachelor of Computer Application**



**EAST POINT COLLEGE OF HIGHER EDUCATION**

**Jnana Prabha East Point Campus Virgo Nagar Post**

**BIDARAHALLI**, **BANGLORE-49**



**Department of Computer Science-BCA-(BNU)**

**East Point Campus, Jnana Prabha Virgo Nagar Post**

**Bidharahalli,Bangalore-560049**

Date:

**CERTIFICATE**

This is to certify that the following students have done their project work on ONLINE E-HOSPITAL submitted to the Bengaluru North University in the partial fulfillment of the requirement for the award of the Degree of work carried out, under the guidance and supervision of Ms.Kounin.

1. Mr. Keerthan K N [R1918048]

This report does not form part of any previous dissertation or reports previously Submitted to this university or any other universities for the award of degree or diploma.

…............ …............ …............

GUIDE HOD PRINCIPLE

Jnana Prabha Campus

#147 Bidarahalli M: +91 9916865999

Virgo Nagar Post T: +91 8028472999 E: [info@eastpoint.ac.in](mailto:info@eastpoint.ac.in)

Bangalore560049 F: +91 8025073903 [www.eastpoint.ac.in](http://www.eastpoint.ac.in)



# **DECLARATION**

I do hereby declare that the project work entitled **ONLINE E-HOSPITAL** Submitted to the Bengaluru North University in the partial fulfillment of the requirements for the award of degree of Bachelor of Computer Application is a record of confide and independent project work carried out by ourself under guidance and supervision of **Ms.Kounin** (Asst. Professor, Dept of BCA) and this report does not form any part of any previous dissertations or reports previously submitted to this University or any other Universities for the award of degree or diploma.

BY:

1.Keerthan K N [R1918048]



**ACKNOWLEDGEMENT**

We take this opportunity to express our deep sense of gratitude to our founder chairman **Dr VENKATAPATHI .S.M**

We express our sincere thanks to our respected Principal **Dr .VIJAYA BHASKARAN** And HOD **Dr . Periyasamy ,** Coordinator of BCADepartment for providing all necessary help during our project work.

We also express our sincere thanks to our respected **MS.Kounin,** Department of BCA for his valuable guidance keen interest and helpful during project.

We would like to thank our staff, friends and all other who have directly and indirectly helped us in the successful completion of this project.

BY:

1. Keerthan K N[R1918048]

**CONTENTS**

|  |  |  |
| --- | --- | --- |
| SL.No | INDEX | Page No |
| 1 | Introduction | 06 |
| 2 | Objective And Scope | 07 |
| 3 | System Requirements Specification | 08 |
| 4 | System Requirement | 09 |
| 5 | Module Description | 10 |
| 6 | Introduction to Front End (JAVA) | 13 |
| 7 | Introduction to Back End (SQL) | 14 |
| 8 | System Analysis | 15 |
| 9 | Feasibility Study | 17 |
| 10 | System Design | 20 |
| 11 | Software Development Process | 21 |
| 12 | E-R Diagram | 26 |
| 13 | Dataflow Diagram | 28 |
| 14 | Data Base Design | 29 |
| 15 | Screen Shots | 35 |
| 16 | Source Code | 45 |
| 17 | Testing | 131 |
| 18 | System Maintenance | 133 |
| 19 | Feature Enhancement | 135 |
| 20 | Conclusion | 135 |
| 21 | Bibliography | 136 |

**INTRODUCTION**

The “Bank Management System” project is a model Banking Application. This application enables the customers to perform the basic banking transactions by sitting at their office or at homes through PC or laptop. The system provides the access to the customer to deposit/withdraw the cash from his account, also to view reports of all accounts present. The customers can access the banks website for viewing their Account details and perform the transactions on account as per their requirements. Today's banking is no longer confined to branches. E-banking facilitates banking transactions by customers round the clock globally.

The primary aim of this “Bank Management System” is to provide an improved design methodology, which envisages the future expansion, and modification, which is necessary for a core sector like banking. This necessitates the design to be expandable and modifiable and so a modular approach is used in developing the application software. Anybody who is an Account holder in this bank can become a member of Bank Account Management System. He has to fill a form with his personal details and Account Number.

Bank is the place where customers feel the sense of safety for their property. In the bank, customers deposit and withdraw their money. Transaction of money also is a part where customer takes shelter of the bank. Now to keep the belief and trust of customers, there is the positive need for management of the bank, which can handle all this with comfort and ease. Smooth and efficient management affects the satisfaction of the customers and staff members, indirectly. And of course, it encourages management committee in taking some needed decision for future enhancement of the bank.

Now a day’s, managing a bank is tedious job up to certain limit. So software that reduces the work is essential. Also today’s world is a genuine computer world and is getting faster and faster day-by-day. Thus, considering above necessities, the software for bank management has became necessary which would be useful in managing the bank more efficiently. All transactions are carried out online by transferring from accounts in the same Bank or international bank. The software is meant to overcome the drawbacks of the manual system .The software has been developed using the most powerful and secure backend MYSQL database which is used in almost all platforms**.**

**Scope and Objectives**

1. Main Goals:

* Our motto is to develop a software program for managing the entire bank process related to Administration accounts customer accounts and to keep each every track about transaction processes efficiently.
* Hereby, our main objective is the customer’s satisfaction considering today’s faster in the world.

2. Customer Satisfaction:

* Client can do his operations comfortably without any risk or losing of his privacy.
* Our software will perform and fulfill all the tasks that any customer would desire.

3. Saving Customer Time:

* Client doesn't need to go to the bank to do small operation.

4. Protecting The Customer:

* It helps the customer to be satisfied and comfortable in his choices, this protection contains customer’s account, money and his privacy.

5. Transferring Money:

* Help client transferring money to/or another bank or country.

## SOFTWARE REQUIREMENT SPECIFICATION

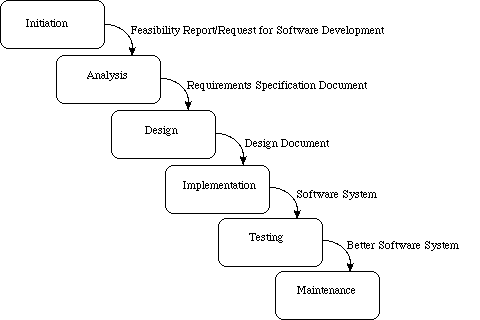
### **(SRS)**

#### **INTRODUCTION**

Software Requirement Specification (SRS) is a fundamental document, which forms the foundation of the software development process. SRS not only lists the requirements of a system but also has a description of its major features. These recommendations extend the IEEE standards. The recommendations would form the basis for providing clear visibility of the product to be developed serving as baseline for execution of a contract between client and the developer.

A system requirement is one of the main steps involved in the development process. It follows after a resource analysis phase that is the task to determine what a particular software product does. The focus in this stage is one of the users of the system and not the system solutions. The result of the requirement specification document states the intention of the software, properties and constraints of the desired system.

SRS constitutes the agreement between clients and developers regarding the contents of the software product that is going to be developed. SRS should accurately and completely represent the system requirements as it makes a huge contribution to the overall project plan. The software being developed may be a part of the overall larger system or may be a complete standalone system in its own right. If the software is a system component, the SRS should state the interfaces between the system and software portion.



**SYSTEM REQUIREMENT**

**HARDWARE AND SOFTWARE REQUIREMENTS**

Once the system analyst have identified the precise user requirements and analysed these requirements to without inconsistencies,they proceed to write the document called the Software Requirement phase.

1.Functional requirements of the system

2.Non-functional requirements of the system

3.Constraints system

**Hardware requirements:**

A Computer with the following minimum requirements:

1. RAM : Minimum size required to run the program(2GB).
2. ROM : Size of the database(20GB).
3. Windows XP and higher versions.
4. Any other peripherals if required like Mouse,Keyboard.

**Software requirements:**

1. Netbeans IDE 7.4
2. My SQL(5.1.5.4)
3. Source codes(Java code) to create,display,insert,delete and update.

**Module Description**

1.**ADMIN MODULE:** The admin module is the major module as it is responsible for carrying out the major operations regarding updates, job alerts and it maintains information regarding other all modules. The various components of admin module are update information, update profile, delete data,

2. **LOGIN MODULE**: In this page, the user needs to give the username and password, and existing user need to login by their respective username and password. Login module will help in authentication of user accounts. In this module the database contains the user’s details.

3. **REGISTER MODULE:** In this page, helps user needs to give the username and password, and existing user need to login by their respective username and password. Login module will help in authentication of user accounts. In this module the database contains the user’s details like Account Number, Name, Contact Number, Password, Address .

4.**ACCOUNT PAGE:** This page is the First page, where user visits on particular site/page .In this page , it convey information about departments and provide access to user to perform the activities like Check Balance ,withdraw, Transfer Amount, Transaction History.

5.**CHECK BALANCE:** In Banking ,the account balance is the amount of money you have available in your account. your account balance is the net amount available to you after all deposits and credits have been balanced with charges or debits. Sometimes your account balance dose not reflect the most accurate representation of your available amount, due to pending transactions or checks that not been processed.

Many other financial accounts also have an account balance. Everything from a utility bill to a mortgage account needs to show you the balance of the account. For financial accounts that have recurring bills, such as a water bill, your account balance usually shows the amount owed. An account balance can also refer to the total amount of money you owe to a third party, such as a credit card company, utility company, mortgage banker, or another type of lender or creditor.

6.**WITHDRAW:** A withdraw involves removing funds from a bank account. In some cases, conditions must be met to withdraw funds without penalty and penalty for early withdrawal usually arises when a clause in an investment contract is broken.

A withdrawal can be carried out over a period of time in fixed or variable amounts or in one lump sum and as a cash withdrawal or in-kind withdrawal. A cash withdrawal requires converting the holdings of an account, plan, pension, or trust into cash, usually through a sale, while an in-kind withdrawal simply involves taking possession of assets without converting to cash

7.**TRANSFER AMOUNT:** When an account holder moves funds from one account to another, say from a [checking account](https://www.investopedia.com/terms/c/checkingaccount.asp) to a [savings account](https://www.investopedia.com/terms/s/savingsaccount.asp) with a higher interest rate, or from savings to an [IRA account](https://www.investopedia.com/terms/i/ira.asp), a transfer has occurred. The transfer does not have to be within the same bank. It can be an interbank transfer from one account held at Bank A to another held at Bank B. Within the banking industry, funds can also be transferred cross-border through [wire transfers](https://www.investopedia.com/terms/w/wiretransfer.asp) from a domestic account to a foreign account, and vice versa. The receiving accounts could be held by the same account holder or could be owned by a different person or company. Funds are normally transferred for purposes of financial planning, to take advantage of better investment rates, to make payments for goods or services, to bring up the required balance of another account, to gift someone or an organization, or to save money, among other reasons.

8.**TRANSACTION HISTORY:** Parts of a bank statement include information about the bank—such as bank name and address—as well as your information. The bank statement will also contain account information and the statement date, as well as the beginning and ending balance of the account. Details of each transaction—notably the amount, date, and payee—that took place in the bank account during the period will also be included, such as deposits, withdrawals, checks paid, and any service charges.

Bank statements are a great tool to help account holders keep track of their money. They can help account holders track their finances, identify errors, and recognize spending habits. An account holder should verify their bank account on a regular basis—either daily, weekly, or monthly—to ensure their records match the bank’s records. This helps reduce overdraft fees, errors, and fraud.

**Introduction to JAVA**

**Front End Application (User Interface):**

Swing API is a set of extensible GUI Components to ease the developer's life to create JAVA based Front End/GUI Applications. It is build on top of AWT API and acts as a replacement of AWT API, since it has almost every control corresponding to AWT controls. Swing component follows a Model-View-Controller architecture to fulfill the following criterias.

* A single API is to be sufficient to support multiple look and feel.
* API is to be model driven so that the highest level API is not required to have data.
* API is to use the Java Bean model so that Builder Tools and IDE can provide better services to the developers for use.

**MVC Architecture**

Swing API architecture follows loosely based MVC architecture in the following manner.

* Model represents component's data.
* View represents visual representation of the component's data.
* Controller takes the input from the user on the view and reflects the changes in Component's data.
* Swing component has Model as a seperate element, while the View and Controller part are clubbed in the User Interface elements. Because of which, Swing has a pluggable look-and-feel architecture.

**Swing Features**

* **Light Weight** − Swing components are independent of native Operating System's API as Swing API controls are rendered mostly using pure JAVA code instead of underlying operating system calls.
* **Rich Controls** − Swing provides a rich set of advanced controls like Tree, TabbedPane, slider, colorpicker, and table controls.
* **Highly Customizable** − Swing controls can be customized in a very easy way as visual apperance is independent of internal representation.
* **Pluggable look-and-feel** − SWING based GUI Application look and feel can be changed at run-time, based on available values.

**Introduction MY SQL**

**Back End Application (Data Store and Management):**

A Back-End Database is a database that is accessed by users indirectly through an external application rather than by application programming stored within the database itself or by low level manipulation of the data (e.g. through SQL commands).

A back-end database stores data but does not include end-user application elements such as stored queries, forms, macros or reports.

MySQL is an open-source relational database management system (RDBMS). Its name is a combination of "My", the name of co-founder Michael Widenius's daughter, and "SQL", the abbreviation for Structured Query Language.

The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements.

MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation. For proprietary use, several paid editions are available, and offer additional functionality.

SQL Server is DBMS tool which has been used by me as back-end due to following reason:

• In today’s competitive environment, an organization wants a comprehensive, secure, reliable and productive data platform for its business applications. SQL Server provides all these facilities.

• SQL Server 2005 combines data analysis, reporting, integration, and notification services.

• The SQL Server database Engine provides a platform that allows managing data application very easily.

• Independesntly accepted standard

• High transaction processing

• Rational architecture: Independent of physical data storage

• Large database and space management

• Client/server (Distributed processing) environment

• Portability and connectivity

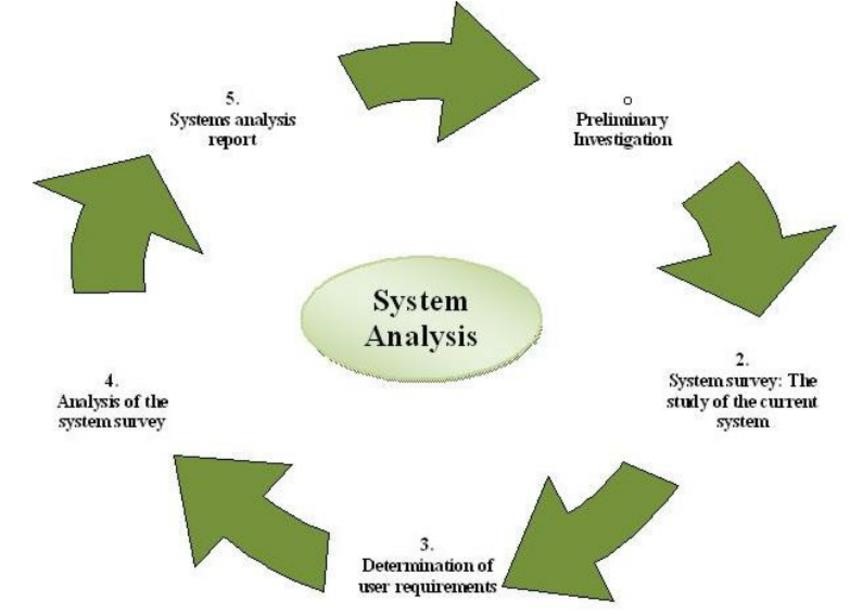
• Back and recovery facilities

• Full proof security management

**SYSTEM ANAYLSIS**

System analysis is a process of gathering and interpreting facts, diagnosing problems and the information is recommend improvements on the system. It is a problem-solving activity that requires intensive communication between the system users and system developers. System analysis or study is an important phrase of any system development process. The system is studied to the minutes details and Analyse it. The system analyst plays the role of the interrogator and wells deep into working of the present system. The system is viewed as a whole and the input to the systems are identified. The outputs from the organizations are trace to the various processes. System analysis concerned with becoming aware of the problem, identifying the element and decisional variables, analysis and synthesizing the various factors and determining an optimal or at least a satisfactory solution or program of action.

A detailed study of the process must by various techniques like interviews, questionnaires etc. The data collected by these sources must be scrutinized to a conclusion. The conclusion is an understanding of how the system functions. This system is called the existing system. Now the existing system is subjected to close study and problem areas are identified. The designer now functions as a problem solver and tries to sort out the difficulties that the enterprise faces. Solution is given as proposals. The proposal is then weighed with the existing system analytically and the best one is selected. The proposal is presented to the user for an endorsement by the user. The proposal is reviewed on user request and suitable changes are made. This is the loop that and as soon as that is satisfied with the proposal.



**EXISTING SYSYEM:**

In existing system, the person who are visiting a particular city need to gather information from the person who is staying in the city or take the help of the city guide. Gathering of all these information you need to visit the city. This possesses a lot of time and preplanning. In order to get each piece of information we need to go for help desk.

**LIMITATIONS OF EXISTING SYSTEM:**

* The existing system is a manual system. Here the city info needs to save the info in the form of excel sheet or disk drivers.
* There is no sharing is possible if data is in the form of paper or disk drivers.
* The manual system gives us very less security for saving data Some data may be lost due to mismanagement.
* It is a limited system and with fewer user friendly.

**PROPOSED SYSTEM:**

The Proposed System provides an online information about the particular city going to visit. It also provides additional services to the registered user. The development of this new system contains the following activities, which try to automate the entire process keeping in the view of database system integration approach.

**ADVANTAGES OF THE PROJECT:**

* User friendliness is provided in the application with various controls provided by rich user interface.
* The system makes the overall project management much easier and flexible.
* The city information files can be stored in centralized database which can be maintained by the system.
* Easy, user-friendly GUI.

**FEASIBILITY STUDY**

The feasibility study proposes one or more feasible conceptual solutions to the problem set of the project. The conceptual solutions give an idea of what the new system will look like. They indicate what inputs are needed by the system and what outputs will be produced. Three things to be done to established feasibility. First, it must be checked that the project is technically feasible. Second, operational feasibility must be established. For this, it is necessary to consult the system users to see if the proposed solution satisfies user objectives and can be fitted in to current system operation. Third, economic feasibility must be checked. The study must determine whether the project ‘s goal can be achieved within the resource limits allocated to it. It must also determine whether it is worthwhile to proceed with the project at all or whether the benefits obtained from the new system are not worth the cost, in which case the project will be terminated.

Feasibility study is necessary to determine whether the proposed system is feasible considering the technical, operational and economic factors. By having detailed feasibility study one can have a clear view of the proposed system with respect to its benefits and draw backs. For a successful feasibility study of system feasibility, the existing systems and proposed system are studied carefully.

System Feasibility The feasibility study is carried out to determine whether the proposed system can be developed with the available resources.

* Operational Feasibility
* Technical Feasibility
* Economic Feasibility
* Motivational Feasibility
* Schedule Feasibility

**TECHNICAL FEASIBILITY**

Technical feasibility is the study of resource availability that may affect the ability to achieve an acceptable system. Technical feasibility is the most difficult area to ensure at initial stages. Since the objectives functions and performance cannot be predicted to its fullest, everything seems possible provided proper assumptions are made. It is essential that the process of technical feasibility. The consideration that is normally associated with technical feasibility included resource availability at the organization where the project is to be developed and implemented.

**ECONOMICAL FEASIBILITY**

An evaluation of development cast weighted against the ultimate income or benefit derived from the developed system. Economical economic justification includes a broad range of concerns that include cost-benefit analysis. Cost benefit delineates costs for project development and weighs them against tangible and intangible benefits of a system. Regarding the cost and benefits, the project, which is to man-hours with compared to man that are required to record data about activity task report manually and also in terms of money benefits by the selling of this system as a product. Thus, this project work is economically feasible for the development in any company.

**MOTIVATIONAL FEASIBILITY**

An evaluation of the probability that the company is significantly motivated to support the development and implementation of the application with necessary user participation, resources, training etc. the participation and support by the organization during system study was encouraging thus eliminating any resistance in this regard. So, from behavioral aspect the new system is supposed to have efficient from the company.

**SCHEDULE FEASIBILITY**

The time schedule required for the development of this project is very important since over- runs result in escalated projects costs and also hinders in the development of the other systems.

**OPERATIONAL FEASIBILITY**

The project is going to be used by the organization under different circumstances. Anyone can work with this application as it supports user-friendly approach. It provides graphical user interfaces to the user, so that user can easily interact with the system. Users no need to have the knowledge about ASP. Net, MSSQL concepts to use the application. The application is designed in such a way that it can be easily implemented in any android version device or cell.

**SYSTEM DESIGN**

**INTRODUCTION TO SYSTEM DESIGN**

System design is the process of defining elements of a system like modules, architecture, components and their interfaces and data for a system based on the specified requirements. And it is the process of defining, developing systems which satisfies the specific needs and requirements of a business or organization.

A systematic approach is required for a coherent and well-running system. Bottom-Up and Top-Down approach is required to take into account all related variables of the system. A designer uses the modelling languages to express to information and knowledge in a structure of system that is defined by a consistent set of rules and definitions. The design can be defined as graphical or textual modelling languages.

Output design generally refers to the results and information that are generated by the system for many end-users; output is the main reason for developing the system and the basis on which they evaluate the usefulness of the application.

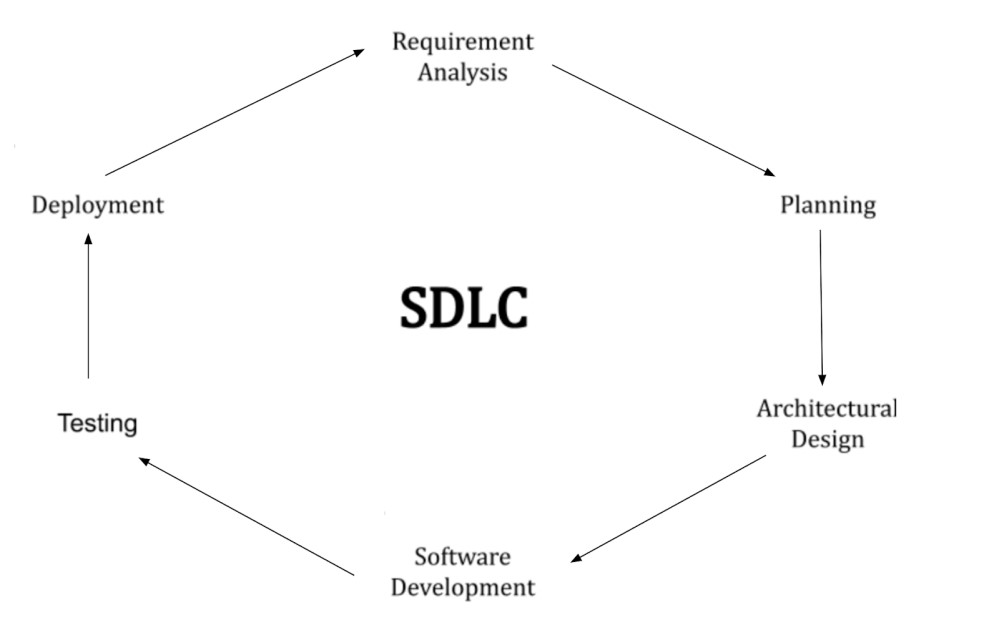
The code design should be such that with less amount of coding we can achieve more results. The speed of the system will be more if the coding is less. Whether the data in the system is usable and readable by the system is depending on the coding. In this project, the coding is being done such that proper validations are made to get the perfect input. No error inputs are accepted. In addition, care is taken such that the data integrity and referential integrity is not violated in the database. In addition, coding is designed such that concurrency avoidance of accessing the database, limited user access to the table is made perfect.

**SOFTWARE DEVELOPMENT PROCESS**

A software development process is the process of dividing software development work into smaller, parallel or sequential steps or subprocesses to improve design and product management. It is also known as a **software development life cycle** (**SDLC**).

SDLC is a process followed for a software project, within a software organization. It consists of a detailed plan describing how to develop, maintain, replace and alter or enhance specific software. The life cycle defines a methodology for improving the quality of software and the overall development process.

The following figure is a graphical representation of the various stages of a typical SDLC.



A typical Software Development Life Cycle consists of the following stages –

**Stage 1: Planning and Requirement Analysis**

Requirement analysis is the most important and fundamental stage in SDLC. It is performed by the senior members of the team with inputs from the customer, the sales department, market surveys and domain experts in the industry. This information is then used to plan the basic project approach and to conduct product feasibility study in the economical, operational and technical areas.

Planning for the quality assurance requirements and identification of the risks associated with the project is also done in the planning stage. The outcome of the technical feasibility study is to define the various technical approaches that can be followed to implement the project successfully with minimum risks.

**Stage 2: Defining Requirements**

Once the requirement analysis is done the next step is to clearly define and document the product requirements and get them approved from the customer or the market analysts. This is done through an **SRS (Software Requirement Specification)** document which consists of all the product requirements to be designed and developed during the project life cycle.

**Stage 3: Designing the Product Architecture**

SRS is the reference for product architects to come out with the best architecture for the product to be developed. Based on the requirements specified in SRS, usually more than one design approach for the product architecture is proposed and documented in a DDS - Design Document Specification.

This DDS is reviewed by all the important stakeholders and based on various parameters as risk assessment, product robustness, design modularity, budget and time constraints, the best design approach is selected for the product.

A design approach clearly defines all the architectural modules of the product along with its communication and data flow representation with the external and third-party modules (if any). The internal design of all the modules of the proposed architecture should be clearly defined with the minutest of the details in DDS.

**Stage 4: Building or Developing the Product**

In this stage of SDLC the actual development starts and the product is built. The programming code is generated as per DDS during this stage. If the design is performed in a detailed and organized manner, code generation can be accomplished without much hassle.

Developers must follow the coding guidelines defined by their organization and programming tools like compilers, interpreters, debuggers, etc. are used to generate the code. Different high level programming languages such as C, C++, Pascal, Java and PHP are used for coding. The programming language is chosen with respect to the type of software being developed.

**Stage 5: Testing the Product**

This stage is usually a subset of all the stages as in the modern SDLC models, the testing activities are mostly involved in all the stages of SDLC. However, this stage refers to the testing only stage of the product where product defects are reported, tracked, fixed and retested, until the product reaches the quality standards defined in the SRS.

**Stage 6: Deployment in the Market and Maintenance**

Once the product is tested and ready to be deployed it is released formally in the appropriate market. Sometimes product deployment happens in stages as per the business strategy of that organization. The product may first be released in a limited segment and tested in the real business environment (UAT- User acceptance testing).

**(About NETBEANS IDE 7.4 )**

**NetBeans** is an [integrated development environment](https://en.wikipedia.org/wiki/Integrated_development_environment) (IDE) for [Java](https://en.wikipedia.org/wiki/Java_(programming_language)). NetBeans allows applications to be developed from a set of modular [software components](https://en.wikipedia.org/wiki/Software_component) called *modules*. NetBeans runs on [Microsoft Windows](https://en.wikipedia.org/wiki/Microsoft_Windows), [macOS](https://en.wikipedia.org/wiki/MacOS), [Linux](https://en.wikipedia.org/wiki/Linux) and [Solaris](https://en.wikipedia.org/wiki/Solaris_(operating_system)). In addition to Java development, it has extensions for other languages like [PHP](https://en.wikipedia.org/wiki/PHP), [C](https://en.wikipedia.org/wiki/C_(programming_language)), [C++](https://en.wikipedia.org/wiki/C%2B%2B), [HTML5](https://en.wikipedia.org/wiki/HTML5),[[3]](https://en.wikipedia.org/wiki/NetBeans#cite_note-3), and [Javascript](https://en.wikipedia.org/wiki/JavaScript). Applications based on NetBeans, including the NetBeans IDE, can be extended by [third party developers](https://en.wikipedia.org/wiki/Third_party_developer).

The NetBeans Platform is a [framework](https://en.wikipedia.org/wiki/Software_framework) for simplifying the development of [Java Swing](https://en.wikipedia.org/wiki/Java_Swing)desktop applications. The NetBeans IDE bundle for Java SE contains what is needed to start developing NetBeans plugins and NetBeans Platform based applications; no additional SDK is required.

Applications can install modules dynamically. Any application can include the Update Center module to allow users of the application to download [digitally signed](https://en.wikipedia.org/wiki/Digital_signature) upgrades and new features directly into the running application. Reinstalling an upgrade or a new release does not force users to download the entire application again.

The platform offers reusable services common to desktop applications, allowing developers to focus on the logic specific to their application. Among the features of the platform are:

* User interface management (e.g. menus and toolbars)
* User settings management
* Storage management (saving and loading any kind of data)
* Window management
* Wizard framework (supports step-by-step dialogs)
* NetBeans Visual Library
* Integrated development tools

NetBeans IDE

**NetBeans IDE** is an [open-source](https://en.wikipedia.org/wiki/Open_source) integrated development environment. NetBeans IDE supports development of all Java application types ([Java SE](https://en.wikipedia.org/wiki/Java_Platform,_Standard_Edition) (including [JavaFX](https://en.wikipedia.org/wiki/JavaFX)), [Java ME](https://en.wikipedia.org/wiki/Java_Platform,_Micro_Edition), [web](https://en.wikipedia.org/wiki/Web_application), [EJB](https://en.wikipedia.org/wiki/EJB) and [mobile](https://en.wikipedia.org/wiki/MIDlet) applications) out of the box. Among other features are an [Ant](https://en.wikipedia.org/wiki/Apache_Ant)-based project system, [Maven](https://en.wikipedia.org/wiki/Apache_Maven) support, [refactorings](https://en.wikipedia.org/wiki/Refactoring), [version control](https://en.wikipedia.org/wiki/Version_control_system) (supporting [CVS](https://en.wikipedia.org/wiki/Concurrent_Versions_System), [Subversion](https://en.wikipedia.org/wiki/Subversion_(software)), [Git](https://en.wikipedia.org/wiki/Git_(software)), [Mercurial](https://en.wikipedia.org/wiki/Mercurial_(software)) and [Clearcase](https://en.wikipedia.org/wiki/Clearcase)).

**Modularity**: All the functions of the IDE are provided by modules. Each module provides a well-defined function, such as support for the [Java language](https://en.wikipedia.org/wiki/Java_(programming_language)), editing, or support for the [CVS](https://en.wikipedia.org/wiki/Concurrent_Versions_System) versioning system, and SVN. NetBeans contains all the modules needed for Java development in a single download, allowing the user to start working immediately. Modules also allow NetBeans to be extended. New features, such as support for other programming languages, can be added by installing additional modules. For instance, [Sun Studio](https://en.wikipedia.org/wiki/Sun_Studio_Compiler_Suite), Sun Java Studio Enterprise, and [Sun Java Studio Creator](https://en.wikipedia.org/wiki/Sun_Java_Studio_Creator) from [Sun Microsystems](https://en.wikipedia.org/wiki/Sun_Microsystems) are all based on the NetBeans IDE.

**License**: From July 2006 through 2007, NetBeans IDE was licensed under Sun's [Common Development and Distribution License](https://en.wikipedia.org/wiki/Common_Development_and_Distribution_License) (CDDL), a license based on the [Mozilla Public License](https://en.wikipedia.org/wiki/Mozilla_Public_License) (MPL). In October 2007, Sun announced that NetBeans would henceforth be offered under a [dual license](https://en.wikipedia.org/wiki/Dual_license) of the CDDL and the [GPL](https://en.wikipedia.org/wiki/GPL) version 2 licenses, with the [GPL linking exception](https://en.wikipedia.org/wiki/GPL_linking_exception) for [GNU Classpath](https://en.wikipedia.org/wiki/GNU_Classpath)[[15]](https://en.wikipedia.org/wiki/NetBeans#cite_note-15) The NetBeans Community blog has announced that Oracle is proposing to entrust the development of the NetBeans platform and IDE to the Apache Foundation to “open up the government model,” reaffirming its commitment to the project. NetBeans is currently submitted as a Proposal to Apache, and it will enter incubation if accepted.

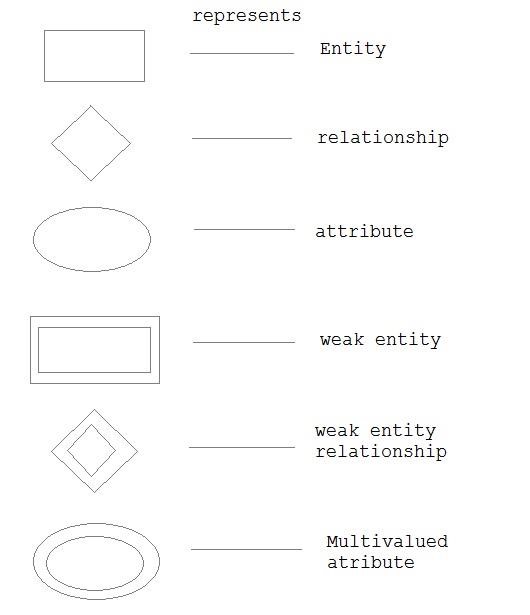
**E-R Diagram**

An entity-relationship model (or ER model) describes interrelated things of interest in a specific domain of knowledge. A basic ER model is composed of entity types (which classify the things of interest) and specifies relationships that can exist between entities (instances of those entity type).

In software engineering, an ER model is commonly formed to represent things a business needs to remember in order to perform business processes. Consequently, the ER model becomes an abstract model, that defines a data or information structure which can be implemented in a database, typically a relational database.

**SYMBOLS USED IN ER DIAGRAM:**

There are three basic elements in an ER diagram: entity, attribute, relationship. They are weak entity, multi valued attribute, derived attribute, weak relationship, and recursive relationship. Cardinality and ordinary are two other notations used in ER diagrams to further define relationships.





**Data Flow Diagram**

A data-flow diagram is a way of responding a flow of data through a process or a system (usually an information system). The DFD also provides information about the outputs and inputs of each entity and the process itself. A data-flow diagram has no control flow, there are no decision rules and no loops. Specific operation based on the data can be represented by flowchart.

There are several notations for displaying data-flow diagrams. For each data-flow at least one of the endpoints (source and /or destination) must exit in a process. The refined representation of a process can be done in another data flow diagram. Which subdivides this process into sub processes.

The data-flow diagram is a part of the structured-analysis modelling tools. When using UML., the activity diagram typically takes over the role of the data-flow diagram. A special form of data-flow plan is a site-oriented data-flow plan.



**DATA BASE DESIGN**

Data base design is defined as a collection of steps that help with designing, creating, implementing and maintaining a business’s data management systems. The main purpose of designing a database is to produce physical and logical models of designs for the proposed database system.

A good database design process is governed by specific rules. The first rule dictates that redundant data must be avoided as its wastes space and increases the probability of faults and discrepancies within the database. The next rule is that the accuracy and comprehensiveness of information is extremely imperative. If the database contains erroneous information any documents that fetch data from such a database will also include inaccurate information.

**CODE DESIGN**

A code design is a document that sets rules for the design. It is a tool that can be used in the design and planning process, but goes further and is more regulatory than other forms of guidance. It can be thought of as a process and document and therefore a mechanism which operationalize design guidelines or standards which have been established through a master plan process

**INPUT DESIGN**

Input design is the process of converting a user-oriented description of the inputs to a computer-based business system into a programmer-oriented specification.

In an information system, input is the raw data is processed to produce output. During the input design, the developers must consider the input devices such as PC, MICR, OMR, etc.

Therefore, the quality of system input determines the quality of system output. Welldesigned input forms and have following properties:

* It should serve specific purpose effectively such as storing, recording and retrieving the information.
* It ensures proper completion with accuracy. bank
* It should be easy to fill and straightforward.
* It should focus on user’s attention, consistency and simplicity.

**OUTPUT DESIGN**

The design of output is the most important task of any system. During output design, developers identify the type of outputs needed and consider the necessary output controls and prototype report layouts.

Objective of output design:

* To develop output design that serves the intended purpose and eliminates the production of unwanted output.
* To develop the output design that meets the end user's requirement.
* To deliver the appropriate quantity of output.
* To form the output in appropriate format and direct it to the right person.
* To make the output available on time for making good decisions.

**TABLE DETAILS**

**ACCOUNTCONFIG TABLE**

CREATE TABLE ‘

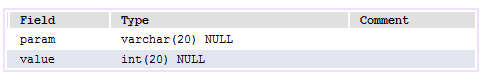
‘accountconfig’(

‘param’ varchar(20) DEFAULT NULL,

‘value’ int(20) DEFAULT NULL,

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

Table-5.1 Accountconfig Table

****

**ACCOUNTS TABLE**

CREATE TABLE `accounts` (

`username` varchar(30) DEFAULT NULL,

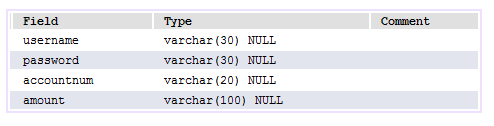
`password` varchar(30) DEFAULT NULL,

`accountnum` varchar(20) DEFAULT NULL,

`amount` varchar(100) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

Table-5.2 Accounts Table

****

**ADMIN TABLE**

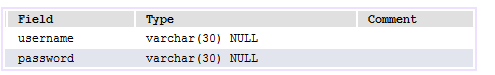
CREATE TABLE `admin` (

`username` varchar(30) DEFAULT NULL,

`password` varchar(30) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

Table-5.3 Admin Table

****

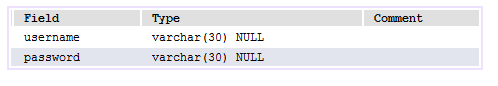
**LOGIN TABLE**

CREATE TABLE `login` (

`username` varchar(30) DEFAULT NULL,

`password` varchar(30) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1; Table-5.4 Login Table



**LOGINDETAILS TABLE**

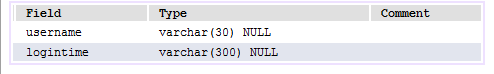
CREATE TABLE `logindetails` (

`username` varchar(30) DEFAULT NULL,

`logintime` varchar(300) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

Table-5.5 Logindetails



**REGESTRATION TABLE**

CREATE TABLE `regestration` (

`Accountnum` varchar(30) DEFAULT NULL,

`Name` varchar(30) DEFAULT NULL,

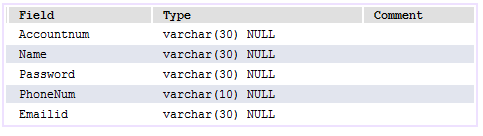
`Password` varchar(30) DEFAULT NULL,

`PhoneNum` varchar(10) DEFAULT NULL,

`Emailid` varchar(30) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

Table-5.6 Regestration Table

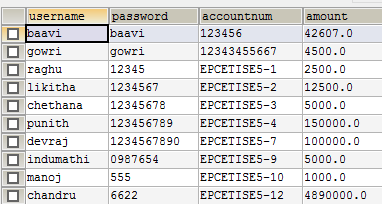


**TABLE WITH VALUES**

**ACCOUNT CONFIG TABLE**



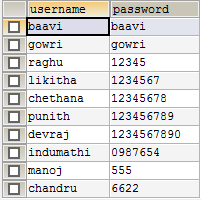
**ACCOUNT TABLE**



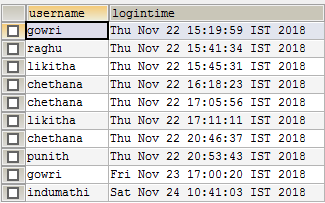
**ADMIN TABLE**



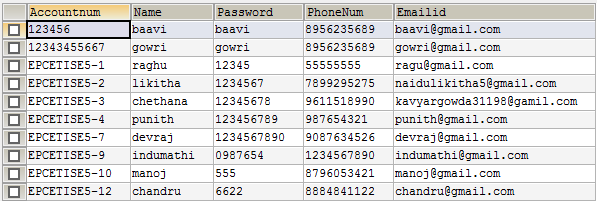
**LOGIN TABLE**



**LOGIN DETAILS TABLE**



**REGESTRATION TABLE**



**SCREEN SHOTS**

**LOGIN FORM**

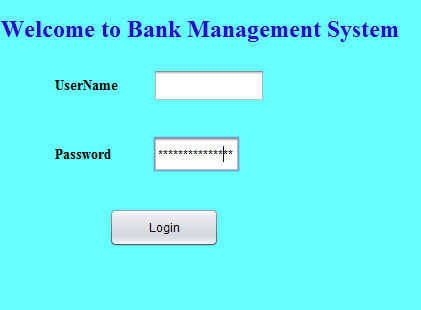


Fig:8(a) LOGIN FORM

**FORM TO SELECT THE REGISTER USER OR THE USER LOGIN**

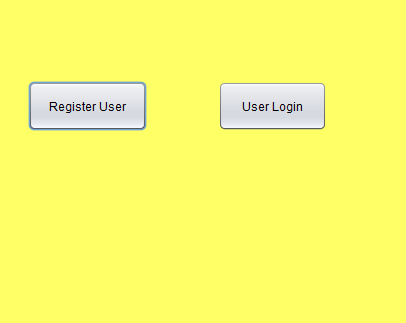


Fig:8(b) REGISTER USER OR USER LOGIN FORM

**FORM TO REGISTER NEW USER**

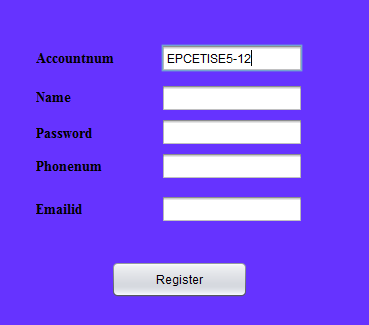


Fig:8(c) Create new user

**LOGIN FORM FOR NEW USER**

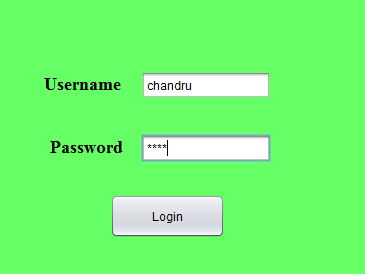


Fig:8(d) New user login form

**FORM TO SELECT THE PARTICULAR PROCESS TO BE DONE IN THE BANK**

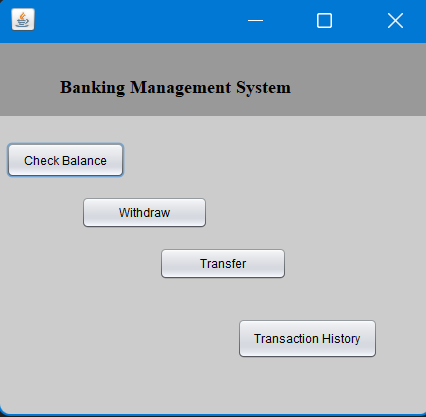


Fig:8(e) Select particular process form

**CHECK BALANCE FORM**

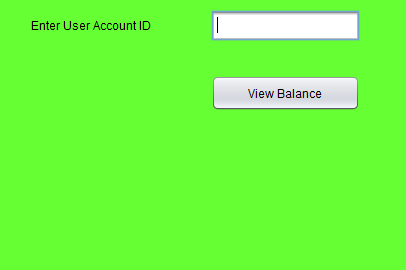


Fig:8(f) Check balance form

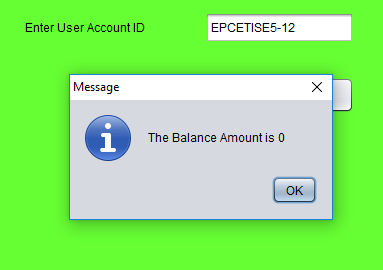
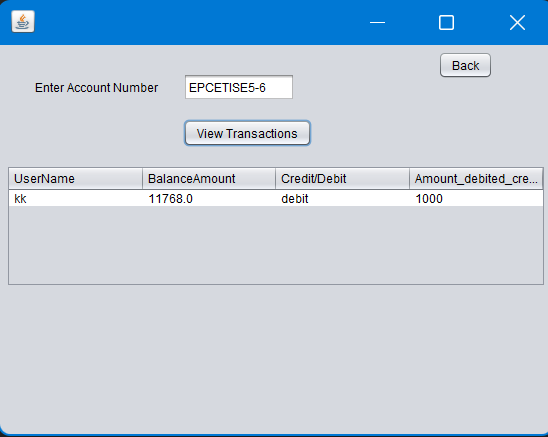


Fig:8(g) Balance amount viewed

**TRANSACTION FORM**

Fig:8(h) Transaction form



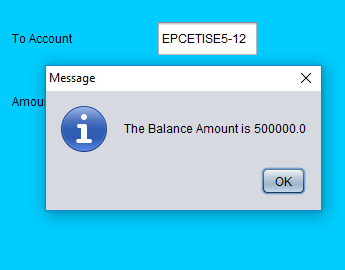


Fig:8(i) Amount after credited to the account

**WITHDRAWAL FORM**

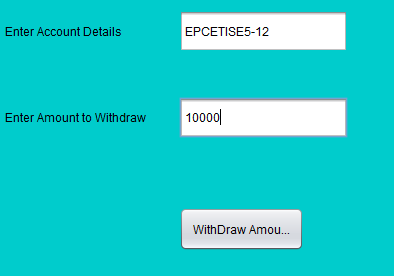


Fig: 8(j) Withdrawal form

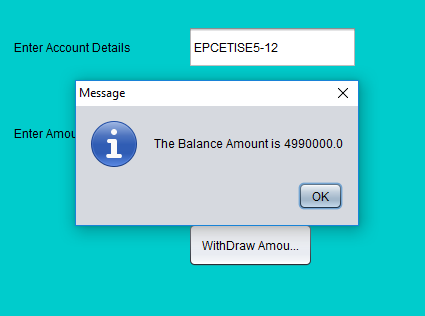


Fig :8(k) Form after withdrawal

**TRANSFER FORM**

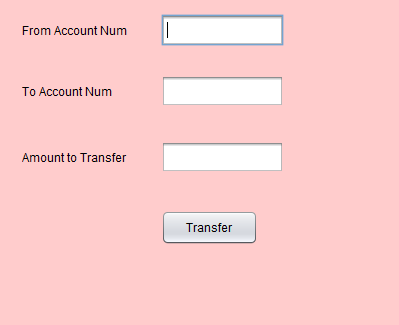


Fig:8(l) Amount transfer form

**FORM AFTER AMOUNT TRANSFER**

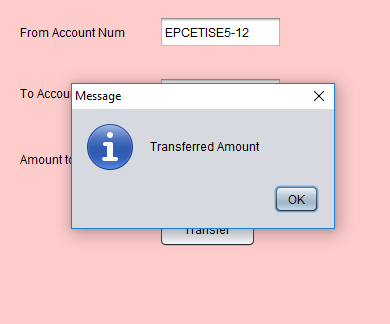


Fig: 8(m) After amont transferred

**USER LOGIN**

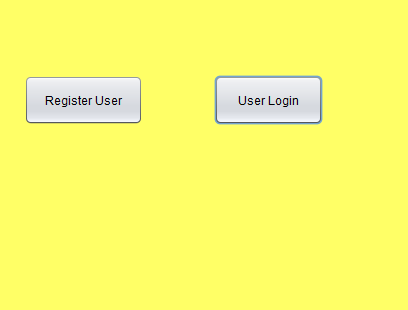


Fig:8(n) User to login

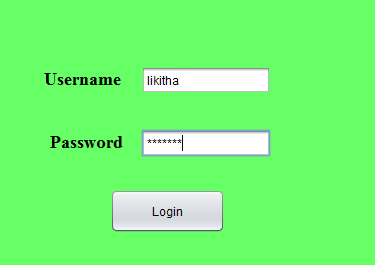


Fig: 8(0) Form to login for already existing customer

**SOURCE CODE**

**Admin page**

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.JOptionPane;

/\*

\* To change this template, choose Tools | Templates

\* and open the template in the editor.

\*/

/\*\*

\*

\* @author keerthan k n

\*/

public class adminpage extends javax.swing.JFrame {

/\*\*

\* Creates new form adminpage

\*/

public adminpage() {

initComponents();

}

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

jLabel1 = new javax.swing.JLabel();

jLabel2 = new javax.swing.JLabel();

jTextField1 = new javax.swing.JTextField();

jLabel3 = new javax.swing.JLabel();

jPasswordField1 = new javax.swing.JPasswordField();

jButton1 = new javax.swing.JButton();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

jLabel1.setText("WELCOME TO BANK MANAGEMENT SYSTEM");

jLabel2.setText("Username");

jLabel3.setText("Password");

jButton1.setText("Login");

jButton1.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton1ActionPerformed(evt);

}

});

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addGap(19, 19, 19)

.addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED\_SIZE, 363, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGroup(layout.createSequentialGroup()

.addGap(34, 34, 34)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

.addGroup(layout.createSequentialGroup()

.addComponent(jLabel3, javax.swing.GroupLayout.PREFERRED\_SIZE, 111, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(18, 18, 18)

.addComponent(jPasswordField1))

.addGroup(layout.createSequentialGroup()

.addComponent(jLabel2, javax.swing.GroupLayout.PREFERRED\_SIZE, 87, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(38, 38, 38)

.addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED\_SIZE, 147, javax.swing.GroupLayout.PREFERRED\_SIZE))))

.addGroup(layout.createSequentialGroup()

.addGap(132, 132, 132)

.addComponent(jButton1)))

.addContainerGap(18, Short.MAX\_VALUE))

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addContainerGap()

.addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED\_SIZE, 32, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(34, 34, 34)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

.addComponent(jLabel2, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jTextField1, javax.swing.GroupLayout.DEFAULT\_SIZE, 32, Short.MAX\_VALUE))

.addGap(18, 18, 18)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel3, javax.swing.GroupLayout.PREFERRED\_SIZE, 33, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jPasswordField1, javax.swing.GroupLayout.PREFERRED\_SIZE, 37, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(53, 53, 53)

.addComponent(jButton1)

.addContainerGap(60, Short.MAX\_VALUE))

);

pack();

}// </editor-fold>

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

String uname=jTextField1.getText();

String pword=jPasswordField1.getText();

try {

Database db = new Database();

String q = "Select \* from admin where username = '"+uname+"' and password = '"+ pword +"'";

ResultSet rs = db.executeQuery(q);

if(rs.next())

{

AdminConfig ac=new AdminConfig();

ac.setVisible(true);

}

else

{

JOptionPane.showMessageDialog(null,"INVALID LOGIN DETAILS","LOGIN ERROR",JOptionPane.ERROR\_MESSAGE);

jTextField1.setText(null);

jPasswordField1.setText(null);

}

} catch (SQLException ex) {

Logger.getLogger(adminpage.class.getName()).log(Level.SEVERE, null, ex);

} catch (ClassNotFoundException ex) {

Logger.getLogger(adminpage.class.getName()).log(Level.SEVERE, null, ex);

}

}

/\*\*

\* @param args the command line arguments

\*/

public static void main(String args[]) {

/\* Set the Nimbus look and feel \*/

//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

/\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

\* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

\*/

try {

for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(adminpage.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(adminpage.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(adminpage.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(adminpage.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

/\* Create and display the form \*/

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new adminpage().setVisible(true);

}

});

}

// Variables declaration - do not modify

private javax.swing.JButton jButton1;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel2;

private javax.swing.JLabel jLabel3;

private javax.swing.JPasswordField jPasswordField1;

private javax.swing.JTextField jTextField1;

// End of variables declaration

}

**Admin config**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

/\*\*

\*

\* @author keerthan k n

\*/

public class AdminConfig extends javax.swing.JFrame {

/\*\*

\* Creates new form AdminConfig

\*/

public AdminConfig() {

initComponents();

}

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

jPanel1 = new javax.swing.JPanel();

jButton1 = new javax.swing.JButton();

jButton2 = new javax.swing.JButton();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

jPanel1.setBackground(new java.awt.Color(255, 255, 102));

jButton1.setText("Register User");

jButton1.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton1ActionPerformed(evt);

}

});

jButton2.setText("User Login");

jButton2.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton2ActionPerformed(evt);

}

});

javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);

jPanel1.setLayout(jPanel1Layout);

jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addGap(34, 34, 34)

.addComponent(jButton1, javax.swing.GroupLayout.PREFERRED\_SIZE, 119, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(71, 71, 71)

.addComponent(jButton2, javax.swing.GroupLayout.PREFERRED\_SIZE, 109, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap(94, Short.MAX\_VALUE))

);

jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addGap(84, 84, 84)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

.addComponent(jButton1, javax.swing.GroupLayout.DEFAULT\_SIZE, 50, Short.MAX\_VALUE)

.addComponent(jButton2, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

.addContainerGap(200, Short.MAX\_VALUE))

);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, layout.createSequentialGroup()

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap())

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, layout.createSequentialGroup()

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap())

);

pack();

}// </editor-fold>

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

userregistration ur = new userregistration();

ur.setVisible(true);

dispose();

}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

userlogin ul = new userlogin();

ul.setVisible(true);

dispose();

}

/\*\*

\* @param args the command line arguments

\*/

public static void main(String args[]) {

/\* Set the Nimbus look and feel \*/

//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

/\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

\* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

\*/

try {

for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(AdminConfig.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(AdminConfig.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(AdminConfig.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(AdminConfig.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

/\* Create and display the form \*/

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new AdminConfig().setVisible(true);

}

});

}

// Variables declaration - do not modify

private javax.swing.JButton jButton1;

private javax.swing.JButton jButton2;

private javax.swing.JPanel jPanel1;

// End of variables declaration

}

**Login page**

import java.sql.Date;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.Calendar;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.JOptionPane;

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

/\*\*

\*

\* @author keerthan kn

\*/

public class userlogin extends javax.swing.JFrame {

/\*\*

\* Creates new form Accounts

\*/

public userlogin() {

initComponents();

}

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

jPanel1 = new javax.swing.JPanel();

jLabel1 = new javax.swing.JLabel();

jLabel2 = new javax.swing.JLabel();

jPasswordField2 = new javax.swing.JPasswordField();

jTextField1 = new javax.swing.JTextField();

jButton1 = new javax.swing.JButton();

jLabel4 = new javax.swing.JLabel();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

jPanel1.setBackground(new java.awt.Color(102, 255, 102));

jPanel1.setForeground(new java.awt.Color(204, 255, 51));

jPanel1.setLayout(null);

jLabel1.setFont(new java.awt.Font("Times New Roman", 1, 18)); // NOI18N

jLabel1.setText("Username");

jPanel1.add(jLabel1);

jLabel1.setBounds(57, 78, 85, 30);

jLabel2.setFont(new java.awt.Font("Times New Roman", 1, 18)); // NOI18N

jLabel2.setText("Password");

jPanel1.add(jLabel2);

jLabel2.setBounds(63, 139, 79, 34);

jPanel1.add(jPasswordField2);

jPasswordField2.setBounds(152, 148, 130, 20);

jTextField1.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jTextField1ActionPerformed(evt);

}

});

jPanel1.add(jTextField1);

jTextField1.setBounds(152, 85, 130, 20);

jButton1.setText("Login");

jButton1.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton1ActionPerformed(evt);

}

});

jPanel1.add(jButton1);

jButton1.setBounds(123, 203, 115, 44);

jLabel4.setIcon(new javax.swing.ImageIcon("D:\\banking\\src\\image.jpg")); // NOI18N

jLabel4.setText("jLabel4");

jPanel1.add(jLabel4);

jLabel4.setBounds(-10, 0, 410, 300);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT\_SIZE, 402, Short.MAX\_VALUE)

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT\_SIZE, 299, Short.MAX\_VALUE)

);

pack();

}// </editor-fold>

private void jTextField1ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

String uname=jTextField1.getText();

String pword=jPasswordField2.getText();

Calendar cal = Calendar.getInstance();

//String date = Calendar

String date = cal.getTime().toString();

System.out.println(date);

try {

Database db = new Database();

String q = "Select \* from login where username = '"+uname+"' and password = '"+ pword +"'";

String q1 = "insert into logindetails values('"+uname+"','"+date+"')";

ResultSet rs = db.executeQuery(q);

if(rs.next())

{

db.executeUpdate(q1);

account acc=new account();

acc.setVisible(true);

}

else

{

JOptionPane.showMessageDialog(null,"INVALID LOGIN DETAILS","LOGIN ERROR",JOptionPane.ERROR\_MESSAGE);

jTextField1.setText(null);

jPasswordField2.setText(null);

}

} catch (SQLException ex) {

} catch (ClassNotFoundException ex) {

Logger.getLogger(userlogin.class.getName()).log(Level.SEVERE, null, ex);

}

}

/\*\*

\* @param args the command line arguments

\*/

public static void main(String args[]) {

/\* Set the Nimbus look and feel \*/

//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

/\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

\* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

\*/

try {

for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(userlogin.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(userlogin.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(userlogin.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(userlogin.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

/\* Create and display the form \*/

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new userlogin().setVisible(true);

}

});

}

// Variables declaration - do not modify

private javax.swing.JButton jButton1;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel2;

private javax.swing.JLabel jLabel4;

private javax.swing.JPanel jPanel1;

private javax.swing.JPasswordField jPasswordField2;

private javax.swing.JTextField jTextField1;

// End of variables declaration

}

Register page

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.logging.Level;

import java.util.logging.Logger;

import java.util.regex.Matcher;

import java.util.regex.Pattern;

import javax.swing.JOptionPane;

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

/\*\*

\*

\* @author keerthan k n

\*/

public class userregistration extends javax.swing.JFrame {

/\*\*

\* Creates new form userregistration

\*/

public userregistration() {

initComponents();

try

{

Database db = new Database();

String q = "select value from accountconfig where param='numseq'";

ResultSet rs = db.executeQuery(q);

rs.next();

int n=rs.getInt("value");

jTextField1.setText("EPCETISE5-"+n);

q = "update accountconfig set value=value+1 where param='numseq'";

db.executeUpdate(q);

db.close();

}

catch(Exception e)

{

}

}

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

jPanel1 = new javax.swing.JPanel();

jLabel1 = new javax.swing.JLabel();

jLabel2 = new javax.swing.JLabel();

jLabel4 = new javax.swing.JLabel();

jLabel5 = new javax.swing.JLabel();

jTextField1 = new javax.swing.JTextField();

jTextField2 = new javax.swing.JTextField();

jTextField4 = new javax.swing.JTextField();

jTextField5 = new javax.swing.JTextField();

jLabel6 = new javax.swing.JLabel();

jButton1 = new javax.swing.JButton();

jPasswordField1 = new javax.swing.JPasswordField();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

jPanel1.setBackground(new java.awt.Color(102, 51, 255));

jPanel1.setToolTipText("Password");

jLabel1.setFont(new java.awt.Font("Times New Roman", 1, 14)); // NOI18N

jLabel1.setText("Accountnum");

jLabel2.setFont(new java.awt.Font("Times New Roman", 1, 14)); // NOI18N

jLabel2.setText("Name");

jLabel4.setFont(new java.awt.Font("Times New Roman", 1, 14)); // NOI18N

jLabel4.setText("Phonenum");

jLabel5.setFont(new java.awt.Font("Times New Roman", 1, 14)); // NOI18N

jLabel5.setText("Emailid");

jTextField1.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jTextField1ActionPerformed(evt);

}

});

jTextField2.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jTextField2ActionPerformed(evt);

}

});

jLabel6.setFont(new java.awt.Font("Times New Roman", 1, 14)); // NOI18N

jLabel6.setText("Password");

jButton1.setText("Register");

jButton1.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton1ActionPerformed(evt);

}

});

jPasswordField1.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jPasswordField1ActionPerformed(evt);

}

});

javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);

jPanel1.setLayout(jPanel1Layout);

jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addGap(55, 55, 55)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

.addComponent(jLabel2, javax.swing.GroupLayout.PREFERRED\_SIZE, 68, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel5, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jLabel4, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jLabel1, javax.swing.GroupLayout.DEFAULT\_SIZE, 94, Short.MAX\_VALUE)

.addComponent(jLabel6, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

.addGap(31, 31, 31)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

.addComponent(jTextField1)

.addComponent(jTextField2)

.addComponent(jTextField4)

.addComponent(jTextField5, javax.swing.GroupLayout.DEFAULT\_SIZE, 142, Short.MAX\_VALUE)

.addComponent(jPasswordField1)))

.addGroup(jPanel1Layout.createSequentialGroup()

.addGap(130, 130, 130)

.addComponent(jButton1, javax.swing.GroupLayout.PREFERRED\_SIZE, 137, javax.swing.GroupLayout.PREFERRED\_SIZE)))

.addContainerGap(137, Short.MAX\_VALUE))

);

jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addGap(62, 62, 62)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED\_SIZE, 25, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel2, javax.swing.GroupLayout.PREFERRED\_SIZE, 26, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jTextField2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addGap(9, 9, 9)

.addComponent(jLabel6, javax.swing.GroupLayout.PREFERRED\_SIZE, 22, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGroup(jPanel1Layout.createSequentialGroup()

.addGap(18, 18, 18)

.addComponent(jPasswordField1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel4, javax.swing.GroupLayout.PREFERRED\_SIZE, 26, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jTextField4, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel5, javax.swing.GroupLayout.PREFERRED\_SIZE, 34, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jTextField5, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(35, 35, 35)

.addComponent(jButton1, javax.swing.GroupLayout.PREFERRED\_SIZE, 37, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap(54, Short.MAX\_VALUE))

);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(0, 0, Short.MAX\_VALUE))

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addContainerGap()

.addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

);

pack();

}// </editor-fold>

private void jTextField1ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

String Accountnum = jTextField1.getText();

String Name = jTextField2.getText();

String Password = jPasswordField1.getText();

String PhoneNum = jTextField4.getText();

String Emailid = jTextField5.getText();

if(validateFields() && isValid(PhoneNum) && isValidEmailID(Emailid))

{

try {

Database db = new Database();

String q = "insert into regestration values('"+Accountnum+"','"+ Name +"','"+Password+"','"+ PhoneNum +"','"+Emailid+"')";

String q1 = "insert into login values('"+ Name +"','"+Password+"')";

String q2 = "insert into accounts values('"+ Name +"','"+ Password+"','"+ Accountnum +"',00000,null,null)";

db.executeUpdate(q);

db.executeUpdate(q1);

db.executeUpdate(q2);

JOptionPane.showMessageDialog(this, "Data Inserted");

userlogin ul = new userlogin();

ul.setVisible(true);

} catch (SQLException ex) {

Logger.getLogger(userregistration.class.getName()).log(Level.SEVERE, null, ex);

} catch (ClassNotFoundException ex) {

Logger.getLogger(userregistration.class.getName()).log(Level.SEVERE, null, ex);

}

}

else

{

JOptionPane.showMessageDialog(this, "Please enter PhoneNumber or Email in Format");

}

}

private void jPasswordField1ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

}

private void jTextField2ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

}

public boolean isValid(String s)

{

Pattern p = Pattern.compile("(0/91)?[6-9][0-9]{9}");

Matcher m = p.matcher(s);

return (m.find() && m.group().equals(s));

}

public boolean isValidPhone(String s)

{

if (isValid(s))

{

return true;

}

else

return false;

}

public boolean isValidmail(String s)

{

Pattern p = Pattern.compile("[a-zA-Z0-9\\.]+@[a-zA-Z0-9\\-\\\_\\.]+\\.[a-zA-Z0-9]{3}");

// Pattern p = Pattern.compile("^\d{10}$");

Matcher m = p.matcher(s);

return (m.find() && m.group().equals(s));

}

public boolean isValidEmailID(String s)

{

if (isValidmail(s))

{

return true;

}

else

return false;

}

public boolean validateFields()

{

if (! validateField( jTextField1.getText(), "Please enter Acc Num"))

return false;

else

if (! validateField( jTextField2.getText(), "Please enter User name"))

return false;

else

if (! validateField( jPasswordField1.getText(), "Please enter Password"))

return false;

else

if (! validateField( jTextField4.getText(), "Please enter Phone Number"))

return false;

else

if (! validateField( jTextField5.getText(), "Please enter Mail ID"))

return false;

else

/\* if(! validateInteger(jTextField2.getText(),"Please enter only Integers"))

return false;

else\*/

// if (! validateInteger( jTextField2.getText(), "Please enter Phone Number"))

// return false;

//else

return true;

}

public boolean validateInteger( String f, String errormsg )

{

try

{ // try to convert input to integer

int i = Integer.parseInt(f);

// input must be greater then 0

// if it is, success

if ( i > 0 )

return true; // success, validation succeeded

}

catch(Exception e)

{

// if conversion failed, or input was <= 0,

// fall-through and do final return below

}

return failedMessage( f, errormsg );

}

public boolean validateField( String f, String errormsg )

{

if ( f.equals("") )

return failedMessage( f, errormsg );

else

return true; // validation successful

}

public boolean failedMessage(String f, String errormsg)

{

JOptionPane.showMessageDialog(null, errormsg); // give user feedback

// f.requestFocus(); // set focus on field, so user can change

return false; // return false, as validation has failed

}

/\*\*

\* @param args the command line arguments

\*/

public static void main(String args[]) {

/\* Set the Nimbus look and feel \*/

//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

/\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

\* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

\*/

try {

for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(userregistration.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(userregistration.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(userregistration.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(userregistration.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

/\* Create and display the form \*/

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new userregistration().setVisible(true);

}

});

}

// Variables declaration - do not modify

private javax.swing.JButton jButton1;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel2;

private javax.swing.JLabel jLabel4;

private javax.swing.JLabel jLabel5;

private javax.swing.JLabel jLabel6;

private javax.swing.JPanel jPanel1;

private javax.swing.JPasswordField jPasswordField1;

private javax.swing.JTextField jTextField1;

private javax.swing.JTextField jTextField2;

private javax.swing.JTextField jTextField4;

private javax.swing.JTextField jTextField5;

// End of variables declaration

}

**Account page**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

/\*\*

\*

\* @author keerthan k n

\*/

public class account extends javax.swing.JFrame {

/\*\*

\* Creates new form UserLogin

\*/

public account() {

initComponents();

}

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

jPanel1 = new javax.swing.JPanel();

jButton1 = new javax.swing.JButton();

jButton2 = new javax.swing.JButton();

jButton3 = new javax.swing.JButton();

jPanel2 = new javax.swing.JPanel();

jLabel1 = new javax.swing.JLabel();

jButton4 = new javax.swing.JButton();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

jPanel1.setBackground(new java.awt.Color(204, 204, 204));

jButton1.setText("Check Balance");

jButton1.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton1ActionPerformed(evt);

}

});

jButton2.setText("Withdraw");

jButton2.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton2ActionPerformed(evt);

}

});

jButton3.setText("Transfer");

jButton3.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton3ActionPerformed(evt);

}

});

jPanel2.setBackground(new java.awt.Color(153, 153, 153));

jLabel1.setBackground(new java.awt.Color(204, 204, 0));

jLabel1.setFont(new java.awt.Font("Times New Roman", 1, 18)); // NOI18N

jLabel1.setText("Banking Management System");

javax.swing.GroupLayout jPanel2Layout = new javax.swing.GroupLayout(jPanel2);

jPanel2.setLayout(jPanel2Layout);

jPanel2Layout.setHorizontalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addGap(60, 60, 60)

.addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED\_SIZE, 278, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap(92, Short.MAX\_VALUE))

);

jPanel2Layout.setVerticalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addGap(22, 22, 22)

.addComponent(jLabel1, javax.swing.GroupLayout.DEFAULT\_SIZE, 45, Short.MAX\_VALUE)

.addContainerGap())

);

jButton4.setText("Transaction History");

jButton4.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton4ActionPerformed(evt);

}

});

javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);

jPanel1.setLayout(jPanel1Layout);

jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel2, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addGroup(jPanel1Layout.createSequentialGroup()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addContainerGap()

.addComponent(jButton1, javax.swing.GroupLayout.PREFERRED\_SIZE, 119, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGroup(jPanel1Layout.createSequentialGroup()

.addGap(81, 81, 81)

.addComponent(jButton2, javax.swing.GroupLayout.PREFERRED\_SIZE, 127, javax.swing.GroupLayout.PREFERRED\_SIZE)))

.addContainerGap())

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel1Layout.createSequentialGroup()

.addGap(0, 0, Short.MAX\_VALUE)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel1Layout.createSequentialGroup()

.addComponent(jButton3, javax.swing.GroupLayout.PREFERRED\_SIZE, 128, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(143, 143, 143))

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel1Layout.createSequentialGroup()

.addComponent(jButton4, javax.swing.GroupLayout.PREFERRED\_SIZE, 141, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(52, 52, 52))))

);

jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addComponent(jPanel2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(26, 26, 26)

.addComponent(jButton1, javax.swing.GroupLayout.PREFERRED\_SIZE, 36, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(18, 18, 18)

.addComponent(jButton2, javax.swing.GroupLayout.PREFERRED\_SIZE, 33, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(18, 18, 18)

.addComponent(jButton3, javax.swing.GroupLayout.PREFERRED\_SIZE, 33, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(38, 38, 38)

.addComponent(jButton4, javax.swing.GroupLayout.PREFERRED\_SIZE, 41, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap(55, Short.MAX\_VALUE))

);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(0, 0, Short.MAX\_VALUE))

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(0, 0, Short.MAX\_VALUE))

);

pack();

}// </editor-fold>

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

WithdrawCash wc = new WithdrawCash();

wc.setVisible(true);

dispose();

}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

Balancecheck blc = new Balancecheck();

blc.setVisible(true);

dispose();

}

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

TransferAmount ta = new TransferAmount();

ta.setVisible(true);

dispose();

}

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

TransactionHistory th = new TransactionHistory();

th.setVisible(true);

dispose();

}

/\*\*

\* @param args the command line arguments

\*/

public static void main(String args[]) {

/\* Set the Nimbus look and feel \*/

//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

/\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

\* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

\*/

try {

for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(account.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(account.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(account.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(account.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

/\* Create and display the form \*/

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new account().setVisible(true);

}

});

}

// Variables declaration - do not modify

private javax.swing.JButton jButton1;

private javax.swing.JButton jButton2;

private javax.swing.JButton jButton3;

private javax.swing.JButton jButton4;

private javax.swing.JLabel jLabel1;

private javax.swing.JPanel jPanel1;

private javax.swing.JPanel jPanel2;

// End of variables declaration

}

**Check balance**

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.JOptionPane;

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

/\*\*

\*

\* @author keerthan k n

\*/

public class Balancecheck extends javax.swing.JFrame {

/\*\*

\* Creates new form Balancecheck

\*/

public Balancecheck() {

initComponents();

}

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

jPanel1 = new javax.swing.JPanel();

jLabel1 = new javax.swing.JLabel();

jTextField1 = new javax.swing.JTextField();

jButton1 = new javax.swing.JButton();

jButton2 = new javax.swing.JButton();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

jPanel1.setBackground(new java.awt.Color(102, 255, 51));

jLabel1.setText("Enter User Account ID");

jButton1.setText("View Balance");

jButton1.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton1ActionPerformed(evt);

}

});

jButton2.setText("back");

jButton2.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton2ActionPerformed(evt);

}

});

javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);

jPanel1.setLayout(jPanel1Layout);

jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addGap(49, 49, 49)

.addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED\_SIZE, 129, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(51, 51, 51)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jButton2)

.addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED\_SIZE, 149, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jButton1, javax.swing.GroupLayout.PREFERRED\_SIZE, 149, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(0, 66, Short.MAX\_VALUE))

);

jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addGap(24, 24, 24)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED\_SIZE, 31, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED\_SIZE, 31, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(34, 34, 34)

.addComponent(jButton1, javax.swing.GroupLayout.PREFERRED\_SIZE, 36, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 122, Short.MAX\_VALUE)

.addComponent(jButton2)

.addGap(68, 68, 68))

);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addContainerGap())

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addContainerGap())

);

pack();

}// </editor-fold>

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

String acc = jTextField1.getText();

try {

Database db = new Database();

String query = " select \* from accounts where accountnum = '"+acc+"' ";

ResultSet rs = db.executeQuery(query);

if(rs.next())

{

String amount = rs.getString("balamount");

JOptionPane.showMessageDialog(this, "The Balance Amount is " +amount);

}

} catch (SQLException ex) {

Logger.getLogger(Balancecheck.class.getName()).log(Level.SEVERE, null, ex);

} catch (ClassNotFoundException ex) {

Logger.getLogger(Balancecheck.class.getName()).log(Level.SEVERE, null, ex);

}

}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {

account acc = new account();

acc.setVisible(true);

dispose();

// TODO add your handling code here:

}

/\*\*

\* @param args the command line arguments

\*/

public static void main(String args[]) {

/\* Set the Nimbus look and feel \*/

//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

/\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

\* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

\*/

try {

for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(Balancecheck.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(Balancecheck.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(Balancecheck.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(Balancecheck.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

/\* Create and display the form \*/

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new Balancecheck().setVisible(true);

}

});

}

// Variables declaration - do not modify

private javax.swing.JButton jButton1;

private javax.swing.JButton jButton2;

private javax.swing.JLabel jLabel1;

private javax.swing.JPanel jPanel1;

private javax.swing.JTextField jTextField1;

// End of variables declaration

}

**Withdrawal**

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.JOptionPane;

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

/\*\*

\*

\* @author keerthan k n

\*/

public class WithdrawCash extends javax.swing.JFrame {

/\*\*

\* Creates new form WithdrawCash

\*/

public WithdrawCash() {

initComponents();

}

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

jPanel1 = new javax.swing.JPanel();

jLabel1 = new javax.swing.JLabel();

jLabel3 = new javax.swing.JLabel();

jTextField1 = new javax.swing.JTextField();

jTextField3 = new javax.swing.JTextField();

jButton1 = new javax.swing.JButton();

jButton2 = new javax.swing.JButton();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

jPanel1.setBackground(new java.awt.Color(0, 204, 204));

jLabel1.setText("Enter Account Details");

jLabel3.setText("Enter Amount to Withdraw");

jButton1.setText("WithDraw Amount");

jButton1.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton1ActionPerformed(evt);

}

});

jButton2.setText("back");

jButton2.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton2ActionPerformed(evt);

}

});

javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);

jPanel1.setLayout(jPanel1Layout);

jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addGap(22, 22, 22)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

.addComponent(jLabel3, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED\_SIZE, 122, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(33, 33, 33)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

.addComponent(jButton1, javax.swing.GroupLayout.PREFERRED\_SIZE, 125, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jTextField1, javax.swing.GroupLayout.DEFAULT\_SIZE, 169, Short.MAX\_VALUE)

.addComponent(jTextField3))

.addContainerGap(85, Short.MAX\_VALUE))

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel1Layout.createSequentialGroup()

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jButton2)

.addGap(25, 25, 25))

);

jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addGap(31, 31, 31)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

.addComponent(jTextField1, javax.swing.GroupLayout.DEFAULT\_SIZE, 42, Short.MAX\_VALUE)

.addComponent(jLabel1, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

.addGap(45, 45, 45)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jTextField3, javax.swing.GroupLayout.PREFERRED\_SIZE, 41, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel3, javax.swing.GroupLayout.PREFERRED\_SIZE, 41, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(39, 39, 39)

.addComponent(jButton1, javax.swing.GroupLayout.PREFERRED\_SIZE, 44, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(48, 48, 48)

.addComponent(jButton2)

.addContainerGap(28, Short.MAX\_VALUE))

);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(0, 0, Short.MAX\_VALUE))

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(0, 0, Short.MAX\_VALUE))

);

pack();

}// </editor-fold>

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

String acc = jTextField1.getText();

String amttowd = jTextField3.getText();

float amtw = Float.parseFloat(amttowd);

float newam = 0;

try {

Database db = new Database();

String query = " select \* from accounts where accountnum = '"+acc+"' ";

ResultSet rs = db.executeQuery(query);

if(rs.next())

{

String amount = rs.getString("balamount");

float am = Float.parseFloat(amount);

newam = am-amtw;

System.out.println(newam);

String q2 = "update accounts set balamount = '"+newam+"' where accountnum = '"+acc+"'";

db.executeUpdate(q2);

JOptionPane.showMessageDialog(this, "The Balance Amount is " +newam);

}

} catch (SQLException ex) {

Logger.getLogger(WithdrawCash.class.getName()).log(Level.SEVERE, null, ex);

} catch (ClassNotFoundException ex) {

Logger.getLogger(WithdrawCash.class.getName()).log(Level.SEVERE, null, ex);

}

}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {

account acc = new account();

acc.setVisible(true);

dispose();

// TODO add your handling code here:

}

/\*\*

\* @param args the command line arguments

\*/

public static void main(String args[]) {

/\* Set the Nimbus look and feel \*/

//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

/\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

\* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

\*/

try {

for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(WithdrawCash.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(WithdrawCash.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(WithdrawCash.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(WithdrawCash.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

/\* Create and display the form \*/

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new WithdrawCash().setVisible(true);

}

});

}

// Variables declaration - do not modify

private javax.swing.JButton jButton1;

private javax.swing.JButton jButton2;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel3;

private javax.swing.JPanel jPanel1;

private javax.swing.JTextField jTextField1;

private javax.swing.JTextField jTextField3;

// End of variables declaration

}

**Transfer Amount**

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.JOptionPane;

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

/\*\*

\*

\* @author keerthan k n

\*/

public class TransferAmount extends javax.swing.JFrame {

/\*\*

\* Creates new form TransferAmount

\*/

public TransferAmount() {

initComponents();

}

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

jPanel1 = new javax.swing.JPanel();

jLabel1 = new javax.swing.JLabel();

jLabel2 = new javax.swing.JLabel();

jTextField1 = new javax.swing.JTextField();

jTextField2 = new javax.swing.JTextField();

jTextField3 = new javax.swing.JTextField();

jLabel3 = new javax.swing.JLabel();

jButton1 = new javax.swing.JButton();

jButton2 = new javax.swing.JButton();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

jPanel1.setBackground(new java.awt.Color(255, 204, 204));

jLabel1.setText("From Account Num");

jLabel2.setText("To Account Num");

jLabel3.setText("Amount to Transfer");

jButton1.setText("Transfer");

jButton1.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton1ActionPerformed(evt);

}

});

jButton2.setText("back");

jButton2.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton2ActionPerformed(evt);

}

});

javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);

jPanel1.setLayout(jPanel1Layout);

jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addGap(28, 28, 28)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

.addComponent(jLabel2, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jLabel1, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jLabel3, javax.swing.GroupLayout.DEFAULT\_SIZE, 105, Short.MAX\_VALUE))

.addGap(34, 34, 34)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

.addComponent(jTextField3, javax.swing.GroupLayout.DEFAULT\_SIZE, 123, Short.MAX\_VALUE)

.addComponent(jTextField1)

.addComponent(jTextField2))

.addComponent(jButton1, javax.swing.GroupLayout.PREFERRED\_SIZE, 97, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addContainerGap(132, Short.MAX\_VALUE))

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel1Layout.createSequentialGroup()

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jButton2)

.addGap(30, 30, 30))

);

jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addGap(31, 31, 31)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

.addComponent(jTextField1, javax.swing.GroupLayout.DEFAULT\_SIZE, 32, Short.MAX\_VALUE)

.addComponent(jLabel1, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

.addGap(29, 29, 29)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel2, javax.swing.GroupLayout.PREFERRED\_SIZE, 24, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jTextField2, javax.swing.GroupLayout.PREFERRED\_SIZE, 32, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(34, 34, 34)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

.addComponent(jTextField3, javax.swing.GroupLayout.DEFAULT\_SIZE, 32, Short.MAX\_VALUE)

.addComponent(jLabel3, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

.addGap(37, 37, 37)

.addComponent(jButton1, javax.swing.GroupLayout.PREFERRED\_SIZE, 35, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 40, Short.MAX\_VALUE)

.addComponent(jButton2)

.addGap(34, 34, 34))

);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(0, 0, Short.MAX\_VALUE))

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(0, 0, Short.MAX\_VALUE))

);

pack();

}// </editor-fold>

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

String facc = jTextField1.getText();

String tacc = jTextField2.getText();

String amttt = jTextField3.getText();

float amtw = Float.parseFloat(amttt);

float newam = 0;

try {

Database db = new Database();

String query = " select \* from accounts where accountnum = '"+facc+"' ";

String query1 = " select \* from accounts where accountnum = '"+tacc+"' ";

ResultSet rs = db.executeQuery(query);

ResultSet rs1 = db.executeQuery(query1);

if(rs.next())

{

String amount = rs.getString("balamount");

float am = Float.parseFloat(amount);

newam = am-amtw;

System.out.println(newam);

String transmode = "debit";

String q2 = "update accounts set balamount = '"+newam+"' where accountnum = '"+facc+"'";

String qq = "update accounts set amount\_debt\_credit = '"+amttt+"' where accountnum = '"+facc+"'";

String qq1 = "update accounts set mode = '"+transmode+"' where accountnum = '"+facc+"'";

db.executeUpdate(q2);

db.executeUpdate(qq);

db.executeUpdate(qq1);

// JOptionPane.showMessageDialog(this, "The Balance Amount is " +newam);

}

if(rs1.next())

{

String amount = rs1.getString("balamount");

float am = Float.parseFloat(amount);

newam = am+amtw;

System.out.println(newam);

String transmode = "credit";

String q3 = "update accounts set balamount = '"+newam+"' where accountnum = '"+tacc+"'";

String qq1 = "update accounts set amount\_debt\_credit = '"+amttt+"' where accountnum = '"+tacc+"'";

String qq2 = "update accounts set mode = '"+transmode+"' where accountnum = '"+tacc+"'";

db.executeUpdate(q3);

db.executeUpdate(qq1);

db.executeUpdate(qq2);

// JOptionPane.showMessageDialog(this, "The Balance Amount is " +newam);

}

JOptionPane.showMessageDialog(this, "Transferred Amount");

} catch (SQLException ex) {

Logger.getLogger(TransferAmount.class.getName()).log(Level.SEVERE, null, ex);

} catch (ClassNotFoundException ex) {

Logger.getLogger(TransferAmount.class.getName()).log(Level.SEVERE, null, ex);

}

}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {

account acc = new account();

acc.setVisible(true);

dispose();

// TODO add your handling code here:

}

/\*\*

\* @param args the command line arguments

\*/

public static void main(String args[]) {

/\* Set the Nimbus look and feel \*/

//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

/\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

\* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

\*/

try {

for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(TransferAmount.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(TransferAmount.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(TransferAmount.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(TransferAmount.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

/\* Create and display the form \*/

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new TransferAmount().setVisible(true);

}

});

}

// Variables declaration - do not modify

private javax.swing.JButton jButton1;

private javax.swing.JButton jButton2;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel2;

private javax.swing.JLabel jLabel3;

private javax.swing.JPanel jPanel1;

private javax.swing.JTextField jTextField1;

private javax.swing.JTextField jTextField2;

private javax.swing.JTextField jTextField3;

// End of variables declaration

}

**Transferred History**

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.table.DefaultTableModel;

/\*

\* To change this template, choose Tools | Templates

\* and open the template in the editor.

\*/

/\*\*

\*

\* @author keerthan k n

\*/

public class TransactionHistory extends javax.swing.JFrame {

/\*\*

\* Creates new form TransactionHistory

\*/

public TransactionHistory() {

initComponents();

}

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

jLabel1 = new javax.swing.JLabel();

jTextField1 = new javax.swing.JTextField();

jButton1 = new javax.swing.JButton();

jScrollPane1 = new javax.swing.JScrollPane();

jTable1 = new javax.swing.JTable();

jButton2 = new javax.swing.JButton();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

jLabel1.setText("Enter Account Number");

jButton1.setText("View Transactions");

jButton1.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton1ActionPerformed(evt);

}

});

jTable1.setModel(new javax.swing.table.DefaultTableModel(

new Object [][] {

},

new String [] {

"UserName", "BalanceAmount", "Credit/Debit", "Amount\_debited\_credited"

}

));

jScrollPane1.setViewportView(jTable1);

jButton2.setText("Back");

jButton2.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton2ActionPerformed(evt);

}

});

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addGap(35, 35, 35)

.addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED\_SIZE, 130, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(18, 18, 18)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addComponent(jButton1)

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

.addGroup(layout.createSequentialGroup()

.addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED\_SIZE, 112, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jButton2)

.addGap(59, 59, 59))))

.addGroup(layout.createSequentialGroup()

.addContainerGap()

.addComponent(jScrollPane1, javax.swing.GroupLayout.DEFAULT\_SIZE, 540, Short.MAX\_VALUE)

.addContainerGap())

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addGap(28, 28, 28)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel1)

.addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)))

.addGroup(layout.createSequentialGroup()

.addContainerGap()

.addComponent(jButton2)))

.addGap(18, 18, 18)

.addComponent(jButton1)

.addGap(18, 18, 18)

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 122, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap(147, Short.MAX\_VALUE))

);

pack();

}// </editor-fold>

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

String accnum = jTextField1.getText();

try {

Database db = new Database();

String q = "Select \* from accounts where accountnum = '"+ accnum +"'";

ResultSet rs = db.executeQuery(q);

String un = null;

String bal = null;

String transmode = null;

String deb\_cred= null;

while(rs.next())

{

un = rs.getString("username");

bal = rs.getString("balamount");

transmode = rs.getString("mode");

deb\_cred = rs.getString("amount\_debt\_credit");

DefaultTableModel model = (DefaultTableModel) jTable1.getModel();

String [] rowdata = new String [4];

rowdata[0] = un;

rowdata[1] = bal;

rowdata[2] = transmode;

rowdata[3] = deb\_cred;

model.addRow(rowdata);

}

} catch (SQLException ex) {

Logger.getLogger(TransactionHistory.class.getName()).log(Level.SEVERE, null, ex);

} catch (ClassNotFoundException ex) {

Logger.getLogger(TransactionHistory.class.getName()).log(Level.SEVERE, null, ex);

}

}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

account acc = new account();

acc.setVisible(true);

dispose();

}

/\*\*

\* @param args the command line arguments

\*/

public static void main(String args[]) {

/\* Set the Nimbus look and feel \*/

//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

/\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

\* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

\*/

try {

for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(TransactionHistory.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(TransactionHistory.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(TransactionHistory.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(TransactionHistory.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

/\* Create and display the form \*/

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new TransactionHistory().setVisible(true);

}

});

}

// Variables declaration - do not modify

private javax.swing.JButton jButton1;

private javax.swing.JButton jButton2;

private javax.swing.JLabel jLabel1;

private javax.swing.JScrollPane jScrollPane1;

private javax.swing.JTable jTable1;

private javax.swing.JTextField jTextField1;

// End of variables declaration

}

**Java Data Base**

/\*\*

\*

\* @author keerthan k n

\*/

import java.sql.\*;

import java.util.\*;

public class Database {

String jdbcDriver = "";

String dbURL = "";

String username = "";

String password = "";

Connection connection;

//Load the Driver(Class.forName(jdbcDriver); where jdbcDriver = "com.mysql.jdbc.Driver";)

// Get the connection ( connection = DriverManager.getConnection(dbURL, username, password);)

//prepare the statement or query ( PreparedStatement st = connection.prepareStatement(query);)

//close the connection (connection.close();)

public Database() throws SQLException, ClassNotFoundException

{

jdbcDriver = "com.mysql.jdbc.Driver";

// jdbcDriver = "oracle.jdbc.driver.OracleDriver";

dbURL = "jdbc:mysql://localhost:3306/bank" ;

//dbURL= "jdbc:oracle:thin:@//server.local:1521/employee";

username = "root";

password = "root";

Class.forName(jdbcDriver); //set Java database connectivity driver

connection = DriverManager.getConnection(dbURL, username, password);

}

public ResultSet executeQuery(String query)throws SQLException

{

PreparedStatement st = connection.prepareStatement(query);

return st.executeQuery();

}

public int executeUpdate(String statement)throws SQLException

{

PreparedStatement st = connection.prepareStatement(statement);

return st.executeUpdate();

}

public void close()

{

try

{

connection.close();

}

catch (SQLException sqlException)

{

//sqlException.printStackTrace();

//connection = null;

}

}

protected void finalize()

{

close();

}

**TESTING**

**Software testing**

Software testing is an investigation conducted to provide stakeholders with information about the quality of the software product or service under test. Software testing can also provide an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation. Test techniques include the process of executing a program or application with the intent of finding software bugs (errors or other defects), and verifying that the software product is fit for use.

Software testing involves the execution of a software component or system component to evaluate one or more properties of interest. In general, these properties indicate the extent to which the component or system under test:

• meets the requirements that guided its design and development

• responds correctly to all kinds of inputs

• performs its functions within an acceptable time

• is sufficiently usable

• can be installed and run in its intended environment and

• achieves the general result its stakeholders desire.

**7.2 Verification**:

Verification is a process of evaluating the intermediary work products of a software development lifecycle to check if we are in the right track of creating the final product. For the verification of this product we can follow the following steps:

1. Login

2. Create an Account

3. View details of an Account

4. Credit amount to the Account

5. View the Transfer details

6. View the Withdraw details

7. Loading the bank customer data grid

**7.3 Validation**

Confirming that the product meets the requirements of the customer and fulfilling his need is validation. It is a process that is used to evaluate whether a product, service, or system complies with regulations, specifications, or conditions imposed at the start of a development phase. Verification can be in development, scale-up, or production. This is often an internal process.

Validation is intended to ensure a product, service, or system results in a product, service, or system that meets the operational needs of the user. For a new development flow or verification flow, validation procedures may involve modeling either flow and using simulations to predict faults or gaps that might lead to invalid or incomplete verification or development of a product, service, or system A set of validation requirements specifications, and regulations may then be used as a basis for qualifying a development flow or verification flow for a product, service, or system . Additional validation procedures also include those that are designed specifically to ensure that modifications made to an existing qualified development flow or verification flow will have the effect of producing a product, service, or system that meets the initial design requirements, specifications, and regulations; these validations help to keep the flow qualified. It is a process of establishing evidence that provides a high degree of assurance that a product, service, or system accomplishes its intended requirements. This often involves acceptance of fitness for purpose with end users and other product stakeholders. This is often an external process

**SYSTEM MAINTENANCE**

**INTODUCTION**

Once the software is delivered and deployed, it enters the maintenance phase. Software needs to be maintained not because some of its components wear out and need to be replaced, but because there are often some residual errors remaining in the system that must be removed as they are discovered.

The objectives of this maintenance work are to make sure that the system gets into work all time without any bug. Provision must be for environmental changes which may affect the computer or software system. This is called the maintenance of the system. Nowadays there is the rapid change in the software world. Due to this rapid change, the system should be capable of adapting these changes. In our project the process can be added without affecting other parts of the system. Maintenance plays a vital role. The system is liable to accept any modification after its implementation. This system has been designed to favor all new changes. Doing this will not affect the system ‘s performance or its accuracy. Maintenance covers a wide range of activities [1], including correcting coding and design errors, updating documents and test data also upgrading user support.

Maintenance can be classified as:

* Corrective maintenance
* Adaptive maintenance
* Preventive maintenance

**CORRECTIVE MAINTENANCE**

Almost all developed software has residual errors. Corrective maintenance means repairing of processing or performance failures or making changes because of previously uncorrected problems or false assumptions.

**ADAPTIVE MAINTENANCE**

Even without bugs, software frequently undergoes change because it must be upgraded and enhanced to include more features and provide services. Once software is deployed, the environment in which it operates changes with time. This changing requires changing or modification in software, called Adaptive maintenance.

**PREVENTIVE MAINTENANCE**

Preventive maintenance is used to ensure the risks of future are minimized. This is one of the best ways to react to risk proactively.

Many activities classified as maintenance are actually enhancements. Maintenance means restoring something to its original condition. Unlike hardware, however software does no wear out; it is corrected. In contrast, enhancement means adding [2], modifying or redeveloping the code to support changes in the specifications. It is necessary to keep up with changing user needs and the operational environment.

**FUTURE ENHANCEMENT**

One of the major future enhancements is making our system online, which would further increase the possibilities of using the project in more effective ways. For the moment Banking System is created to work as an independent platform in itself. The project has a very vast scope in future. The project can be implemented on web-based application as a future enhancement. Project can be expanded further based on the requirements, as it is very flexible in terms of expansion.

In future we would like to add voice assistance feature to our project to build much more user-friendly experience and also future enhancements include city updates, travel history, job related alerts and vacancy notifications, travel recommendations on user interest and many more functionality can be added depending upon the user requirements and specifications.

**CONCLUSION**

An attempt has been made to develop an Bank Management System which meets necessary requirements of the user successfully. Since it is user friendly it enables the user to interact efficiently.

Though the system still contains a lot of scope of improvement in it, but its overall look an feel is a rough picture on an existing automation system.

## 

## BIBLIOGRAPHY

During this development of our system, we have taken the reference from books, journals and few websites, which we would like to mention in this section.

BOOKS REFERED:

1. Java programming, Srikanth S
2. Software engineering, Ashwini S Diwakar
3. Database Management System, James s Larson
4. System Analysis and Design, Kenneth E

WEB REFERENCES:

1. [www.mysql.com](http://www.mysql.com/)
2. [www.google.com](http://www.google.com/)
3. [www.youtube.com/udemy](http://www.youtube.com/)
4. [www.programmingknowledge.com](http://www.programmingknowledge.com/)
5. [www.softwareengineering/concepts/](http://www.softwareengineering/concepts/)
6. [www.softwareconcepts/testing/](http://www.softwareconcepts/testing/)