**Veer Narmad South Gujarat University, Surat.**

**Department of Information and Communication Technology**

**M.Sc. (Information Technology) Programme**

Project Report

**8th Semester**

**M.Sc. (Information Technology)**

**5 Year Integrated Course**

**Year 2024 – 2025**

**Loan Management System**

|  |  |
| --- | --- |
| Guided By :  **Mrs.Shreya (Internal Guide)** | Submitted By :  **Falguni Dabhi**  **(E19110018000610018)**  **Aditi Bharti**  **(E20110018000610009)** |

### **Index**

|  |  |  |
| --- | --- | --- |
| Sr. No | Topics | Page no |
| 1 | Introduction  1.1 Profile  1.2 Customer Profile  1.2.1 Current System – about the existing system (manual or s/w) | 4  5  6 |
|  | 1.2.2 Customer Detail | 6 |
| 2 | Proposed System  2.1 Scope  2.2 Objective  2.3 Constraints – things that cannot be done  2.3.1 H/w Constraints  2.3.2.S/W Constraints  2.4 Advantages  2.5 Limitation – things that are not there, but could be achieved | 7  7  7      7  7 |
| 3 | Environment Specification  3.1 Hardware & Software Requirements – things that are needed for your s/w  3.2 Development Description | 8    8 |
| 4 | System Planning  4.1 Feasibility Study  4.2 Software Engineering Model  4.3 Risk Analysis  4.4 Project Schedule  4.4.1 Task Dependency  4.4.2 Timeline Chart  4.4.3 Project Table | 9  10  11    12  13  14 |
| 5 | System Analysis  5.1 Detailed SRS  5.2UML Diagram  5.2.1 Use Case Diagram  5.2.2 CRC  5.2.3 Class Diagram  5.2.4 Activity Diagram | 14    16    19  20 |
|  | 5.2.5 Sequence Diagram  5.3 E-R Diagram | 25  28 |
| 6 | Software Design  6.1 Database Design  6.2 Interface Design sitemap followed with page snapshots  6.3 Architecture Design | 29  36    62 |
| 7 | Testing  7.1 Unit Testing  7.2 Integration Testing | 63  63 |
| 8 | Future Enhancement | 68 |
| 9 | Glossary | 68 |
| 10 | Reference | 69 |

**1.1 Project Profile**

|  |  |
| --- | --- |
| Project Title | Loan Management System |
| Group Members | Dabhi Falguni Rameshbhai  Bharti Aditi Raghunandan |
| College | J.P. Dawer Institute of Information Science  and Tech. |
| Duration | Three Months |
| Internal Project  Guide | Mrs. Shreya Ma’am |
| Submitted By | Dabhi Falguni Rameshbhai  Bharti Aditi Raghunandan |
| Submitted To | Department Of ICT, VNSGU |

**System Configuration:**

|  |  |
| --- | --- |
| Project Title | Loan Management System |
| Front End | Bootstrap , Html5 , CSS3, javascript,  Ajax |
| Back End | My SQL ,Java JSF |
| Operating System | Microsoft Windows 10 |
| Language | Java, Ajax ,JSF, Bootstrap , Html5 , CSS3, javascript |
| Database Server | Payara Server |
| Development  Environment | Integrated Development Environment (IDE)- Net Beans 17 |

### **2. System Specification**

* 1. **Scope**

In this system , The scope of this project is to provide good communication between the client and the employee. Client applying for loan and employee will check the client details according to that employee will give loan.

**2.2 Objective**

* + - ***Admin***

Admin can manage Client , Companies , employee and other entities of database.

* + - ***Employee***

Employee will manage their profile, can create customers, can also assigned new call, coordinator, executor. Employee also check about client existing loan and also check about call history.

* + - ***Client***

Client will manage their profile, can register as a member and also decide to take loan and contact about company and loan information.

**2.3 Constraints**

Software Constraints:

* + Any web browser .
  + Internet Connection Required.

* 1. **Advantages:**

1. **Convenience and Accessibility**

* **24/7 Access**: Customers can access their loan accounts, make payments, and apply for new loans anytime, from anywhere, using any internet-enabled device.
* **Remote Application Process**: No need to visit physical branches; loan applications can be completed online, saving time and travel costs.

1. **Faster Loan Processing**

* **Quick Approvals**: Online systems often use automated processes to evaluate applications, leading to faster approval times.
* **Instant Feedback**: Customers receive immediate updates on their loan application status, reducing the anxiety of waiting for decisions.

1. **Easy Account Management**

* **Real-Time Information**: Customers can view real-time updates on their loan balances, payment schedules, and transaction history.
* **Automated Payments**: Options for setting up automatic payments help ensure that customers never miss a due date, avoiding late fees.

1. **Transparent Terms and Conditions**

* **Clear Information**: Online platforms often provide detailed information about loan terms, interest rates, fees, and repayment options, enhancing transparency.
* **Comparative Tools**: Customers can easily compare different loan products and choose the one that best fits their needs.

1. **Lower Costs**

* **Reduced Fees**: Many online loan platforms offer lower processing fees and interest rates compared to traditional banks due to lower operational costs.
* **No Hidden Charges**: Online systems typically offer clear, upfront information on all costs associated with the loan, avoiding hidden fees.

1. **Enhanced User Experience**

* **User-Friendly Interfaces**: Intuitive design and easy navigation make it simple for customers to manage their loans online.
* **Personalized Services**: Data analytics can provide personalized loan offers and recommendations based on individual customer profiles.

1. **Improved Security**

* **Secure Transactions**: Advanced encryption and authentication methods protect sensitive customer information.
* **Fraud Alerts**: Customers can set up alerts for suspicious activity, providing additional security against fraud.

1. **Access to Financial Education**

* **Educational Resources**: Many online loan systems offer educational content and tools to help customers understand their finances and make informed decisions.
* **Credit Score Monitoring**: Some platforms provide free credit score monitoring, helping customers track and improve their credit health.

* 1. **Limitations:**

Not enough call Interaction.

1. **Privacy and Security Concerns**

* **Data Privacy Risks**: Customers must share sensitive personal and financial information online, which can be vulnerable to data breaches and misuse.
* **Identity Theft**: The risk of identity theft can increase if personal information is compromised due to inadequate security measures.

1. **Digital Divide**

* **Limited Access for Some**: Not everyone has reliable internet access or the necessary digital literacy to use online loan services effectively.
* **Exclusion of Older Adults**: Older adults or those less comfortable with technology may find it challenging to navigate online systems.

1. **Potential for Cyber Threats**

* **Hacking and Phishing**: Online platforms can be targets for hackers and phishing attacks, posing risks to customer data and financial security.
* **Scams and Fraud**: Customers may be exposed to fraudulent loan offers and scams that appear legitimate but are designed to steal personal information.

1. **Less Personalized Service**

* **Lack of Human Interaction**: Online systems may lack the personal touch of in-person service, making it harder for customers to get personalized advice and support.
* **Automated Responses**: Automated systems may not address unique customer issues or provide the level of empathy and understanding a human representative can offer.

1. **Dependence on Technology**

* **Technical Issues**: System downtimes, software glitches, or slow internet connections can disrupt access to loan services and information.
* **Device Dependence**: Customers need to have access to and be comfortable using digital devices like computers, smartphones, or tablets.

1. **Complexity of Terms**

* **Understanding Digital Contracts**: Customers may find it difficult to understand and navigate the fine print of digital loan agreements without expert help.
* **Information Overload**: The abundance of information available online can be overwhelming, making it challenging to make informed decisions.

1. **Impulsive Decisions**

* **Ease of Application**: The simplicity of applying for loans online can lead to impulsive decisions without adequate consideration of the terms and conditions.
* **Debt Accumulation**: Easy access to credit might encourage over-borrowing and increase the risk of falling into debt traps.

### **3.Environment Specification**

**́]**

**3.1 Hardware & Software Requirements:**

|  |  |
| --- | --- |
| Processor | P4/Dual Core or Higher Frequency Processor |
| Hard Disc | 50 GB or Higher |
| RAM | 8 GB or Higher |
| Key Board | Microsoft Slim Key Board |
| Mouse | Microsoft Optical Mouse |
| Monitor | CRT, LCD, TFT (anyone can use) |

* **Software Configuration:**
  + - o Apache Net beans 17
    - o Google Chrome, Mozilla Firefox
    - o Xampp

**3.2 Development description:**

* Java
* Jsf
* Ajax
* Operating System (for Development):
* Microsoft Windows
* Javascript
* JQuery
* SQL Server
* HTML5
* CSS3
* BOOTSTRAP

### **4. System Planning :**

**4.1.1 Feasibility Study**

A feasibility study is an evaluation and analysis of the potential of the proposed project which is based on extensive investigation and research to give full comfort to the decisions makers.

* **Technical feasibility:**

A large part of determining resources has to do with assessing technical feasibility. It considers the technical requirements of the proposed project. The technical requirements are then compared to the technical capability of the organization. The systems project is considered technically feasible if the internal technical capability is sufficient to support the project requirements.

* **Operational feasibility:**

Operational feasibility is dependent on human resources available for the project and involves projecting whether the system will be used if it is developed and implemented.

Operational feasibility is a measure of how well a proposed system solves the problems and takes advantage of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of system development.

* **Economic feasibility:**

As development tools and software are already we own, there isn’t any burden of buying them. The organization doesn’t have the same system, so it is certainly required for them. So here we do not need to invest extra funds to develop the system. Thus, it is economically feasible to the organization.

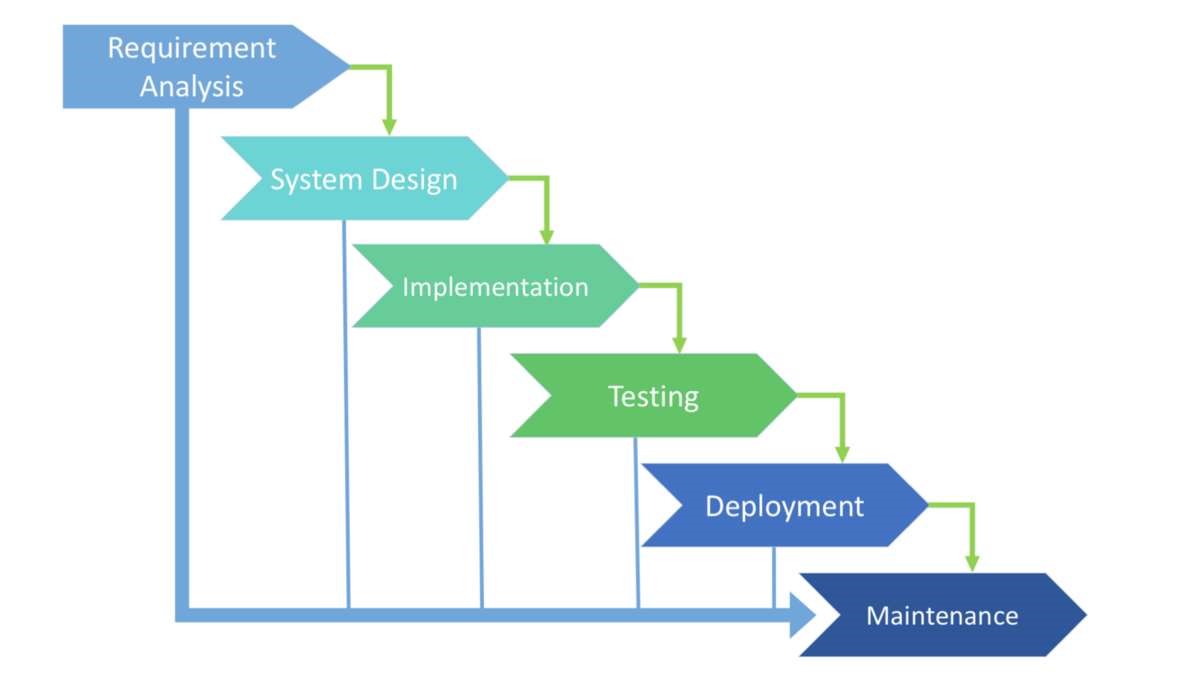
* **Time Schedule feasibility:**

Projects are initiated with specific deadline. We need to evaluate whether the deadlines are mandatory or desirable. Time is the one of the critical factor in the development of any system but this kind of feasibility is hardly perfect in any system.

We have been asked to complete the project within the working days of the organization having period of 3 months approximately. So, we have managed to complete the project before given deadline. In the project planning section, we elaborate our ideas to develop the system within the given period.

## 4.2 Software Engineering Model

We followed Iterative Waterfall Model for Software development .So, we can add new functionalities as per requirement.

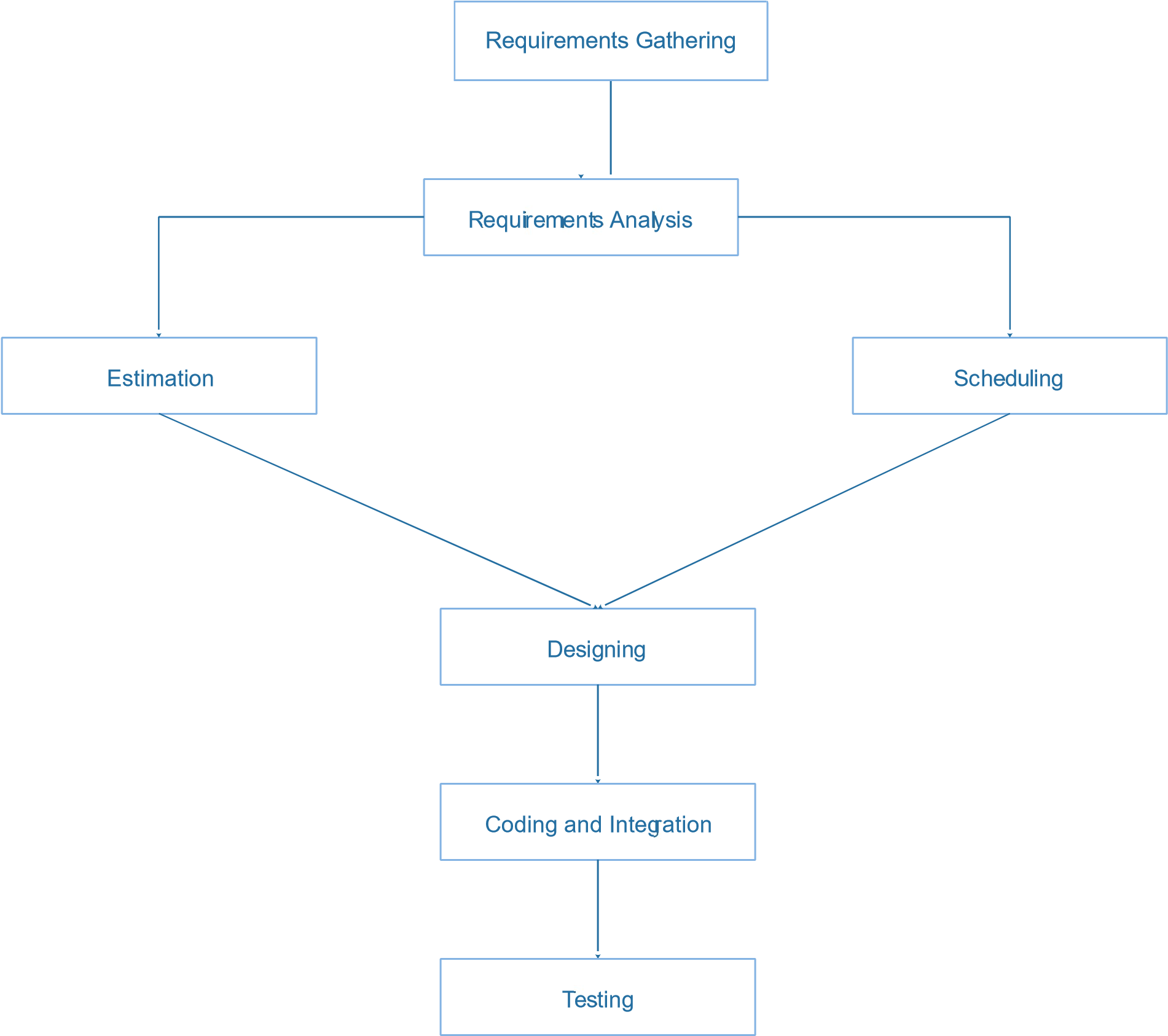


## 4.3 Risk Analysis :

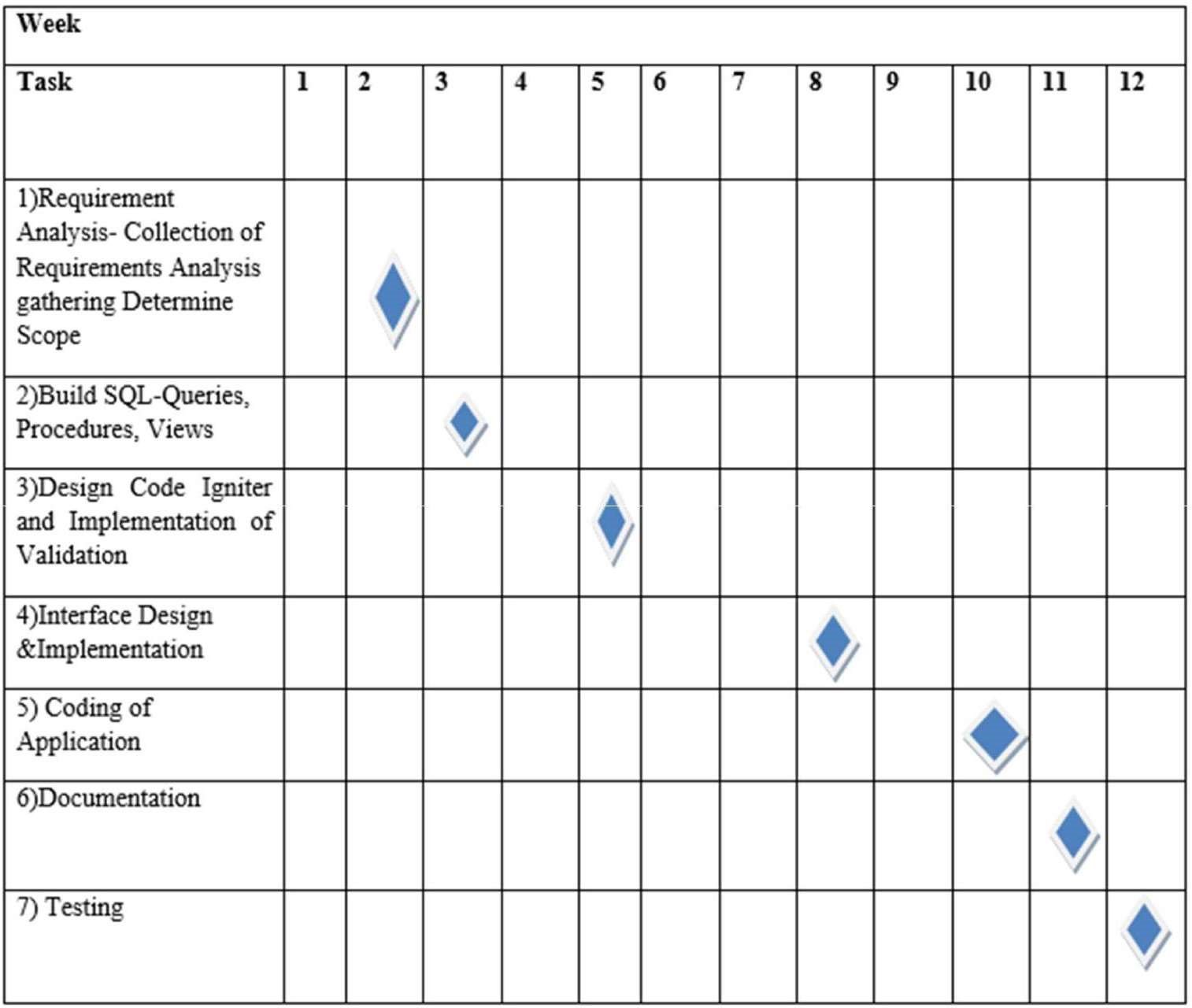
* The probability of any unwanted incident is defined as Risk. In Software Testing, risk analysis is the process of identifying the risks in applications or software that you built and prioritizing them to test. After that, the process of assigning the level of risk is done. The categorization of the risks that takes place, hence, the impact of the risk is calculated.
* Risks in our project
  + Changes in database or model.
  + Issues with designing after updating module.

### **4.4 Project Schedule :**

**4.4.1 Task Dependency**



#### **4.4.2 Timeline Chart**



1. **System Analysis:**

**5.1 Detailed SRS :**

The purpose of SRS is to write the functional and nonfunctional system requirements that represent the characteristic of Loan4You. This system must be designed as user required. So, the complete requirement must be found.

**Features:**

**Admin:**

* Admin Can add , view , update , delete Client.
* Admin Can add , view , update , delete Branch.
* Admin Can add , view , update , delete Bank.
* Admin Can add , view , update , delete Loan Type.
* Admin Can add , view , update , delete Employee.
* Admin Can add , view , update , delete Calls.
* Admin Can add , view ,update , delete Existing Loan.
* Admin Can add , view ,update , delete Company.
* Admin Can add , view ,update , delete Documents.
* Admin Can add , view ,update , delete Login Files.
* Admin Can add, view ,update , delete Call History.

**Client:**

* Client Can do their Login.
* Client Can apply for Loans.
* Client Can view Profile.
* Client Can view Document.
* Client Can view Existing Loan.
* Client Can view Login Files.
* Client Can view Call History.

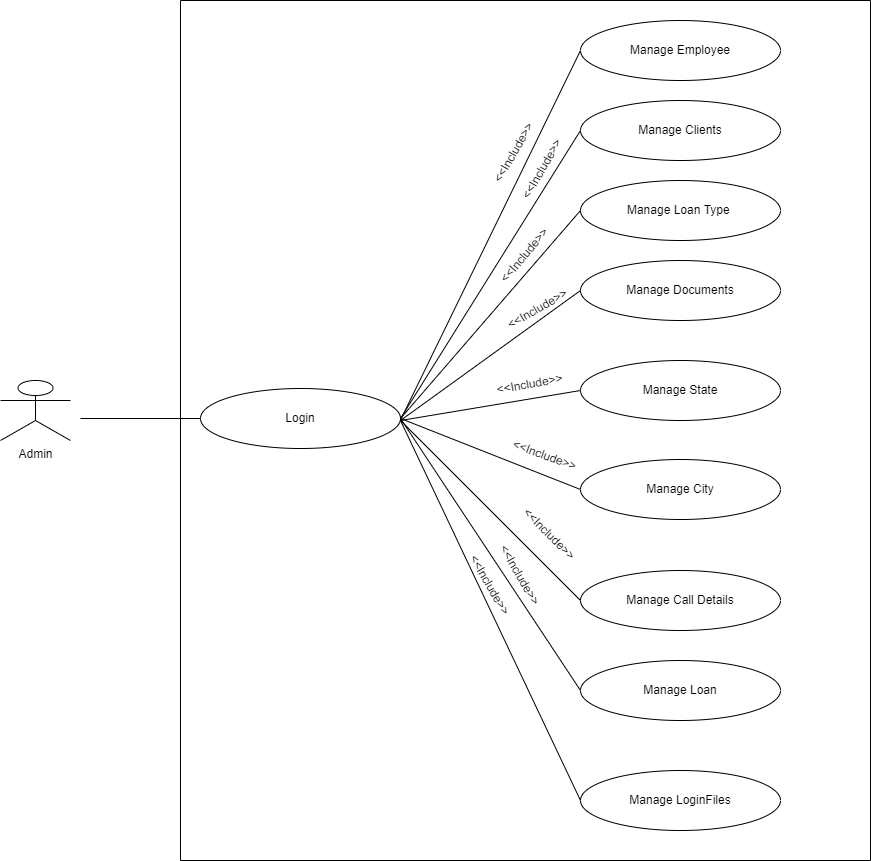
**Employee:**

* Employee Can do their Login.
* Employee Can manage Documents.
* Employee Can manage Client.
* Employee Can manage Calls.
* Employee Can manage Employee.

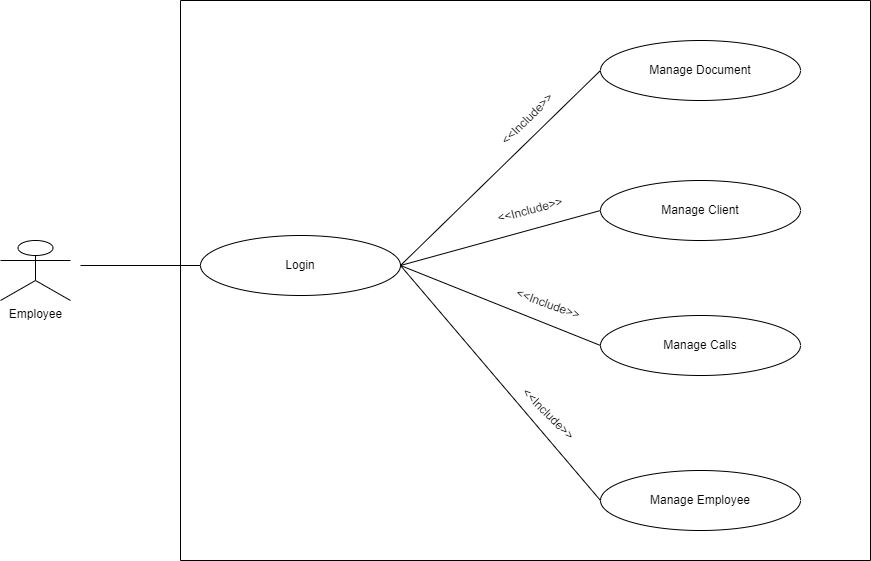
**5.2 UMl Diagrams**

**5.2.1 Use Case Diagram**

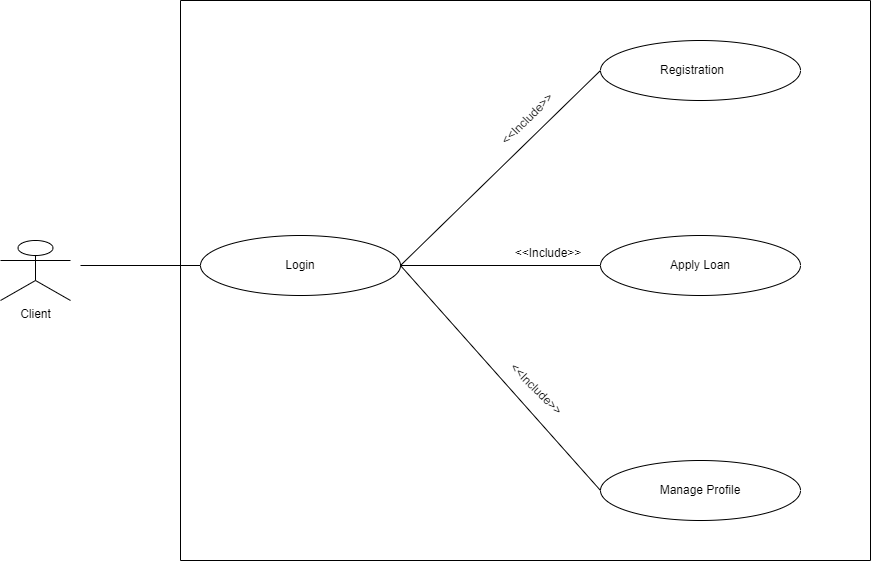
**Admin**



* + - * **Employee**

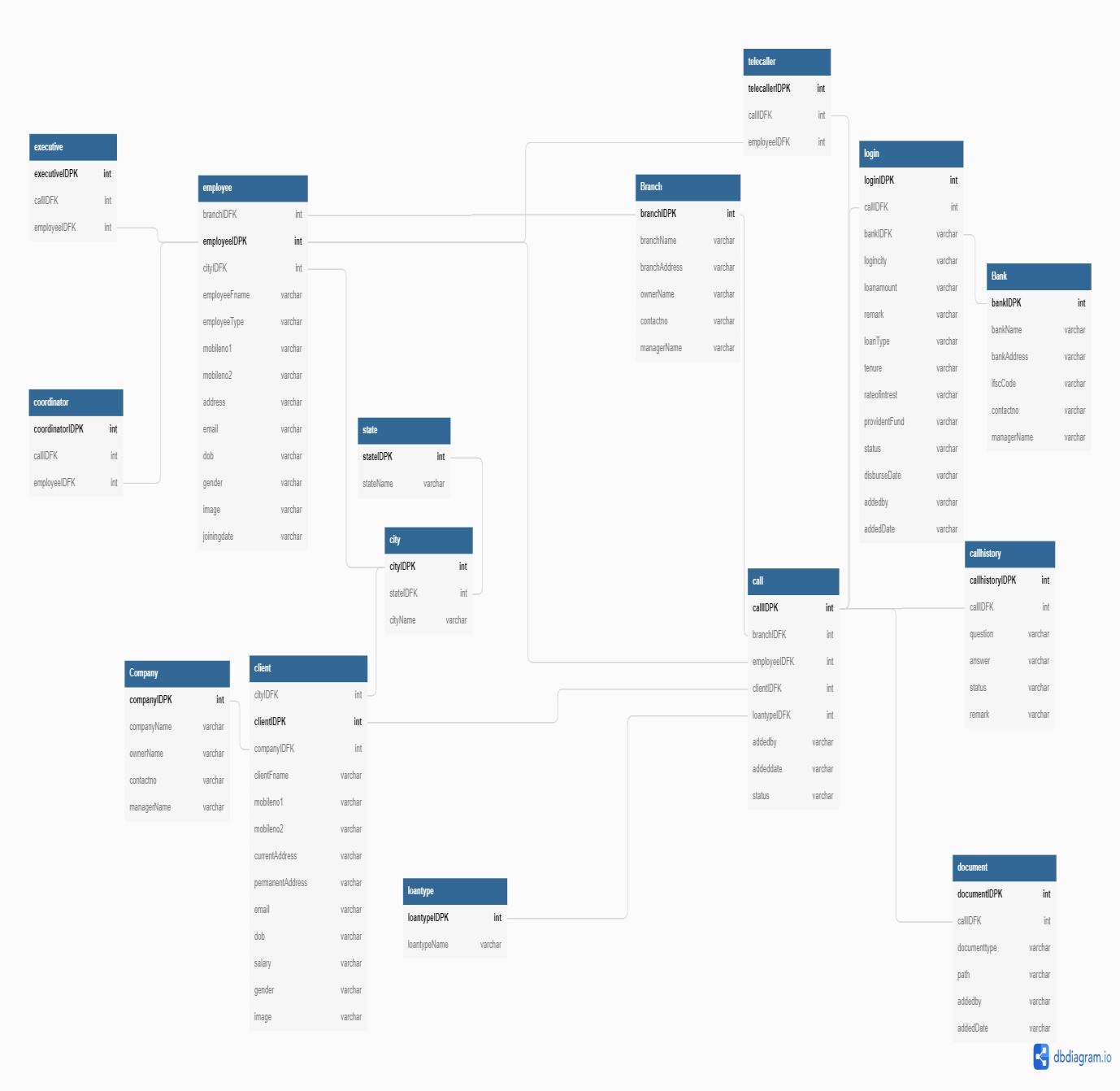


* + - * **Customer**



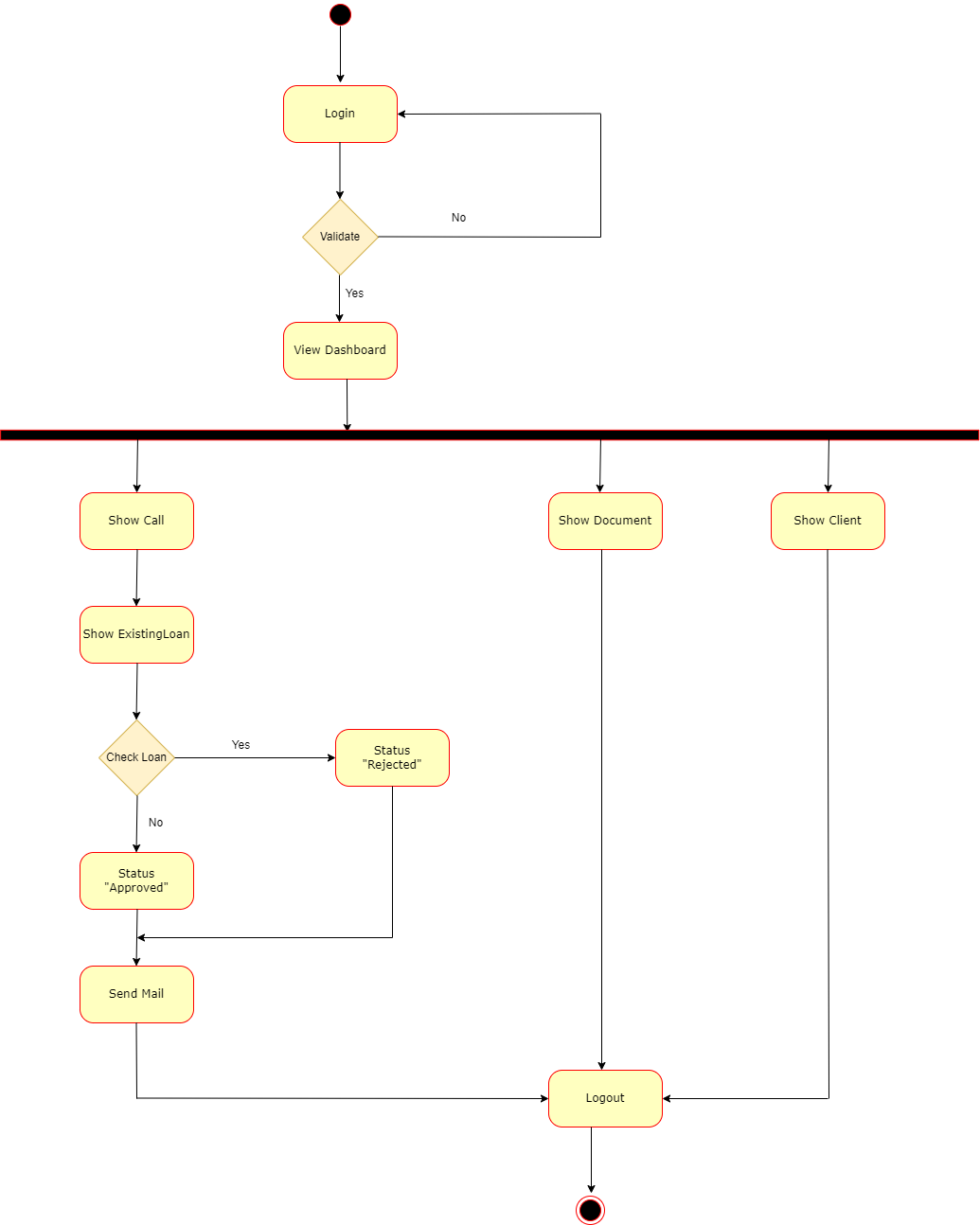
**5.2.2**

**Class Diagram**

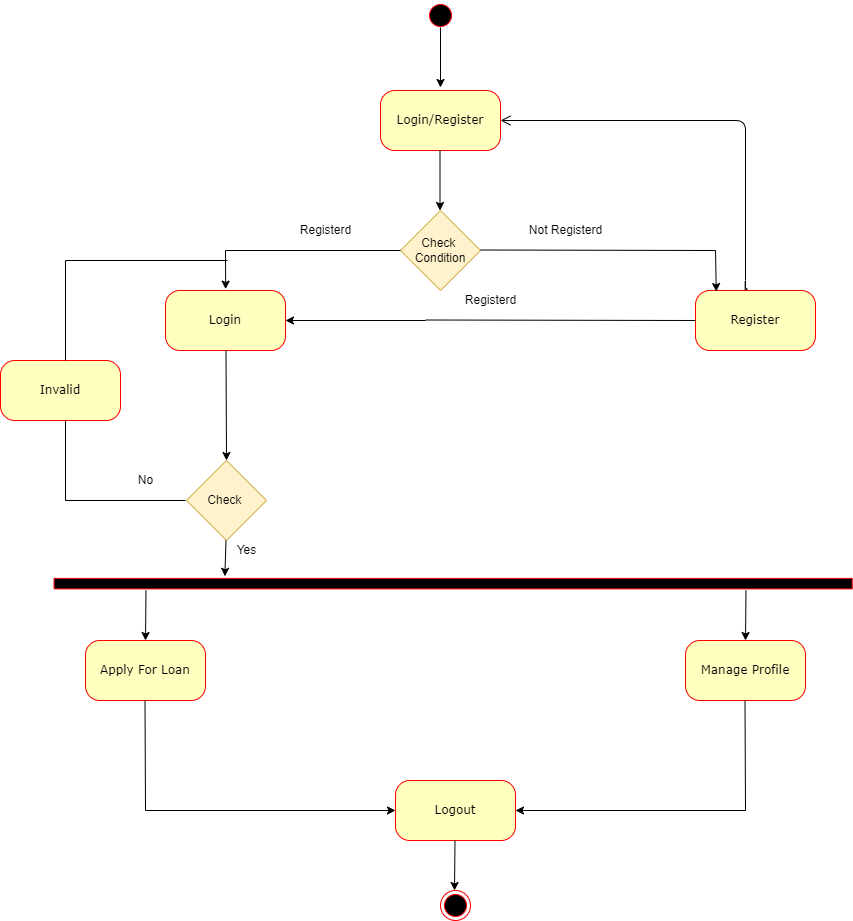


**5.2.4 Activity Diagrams:**

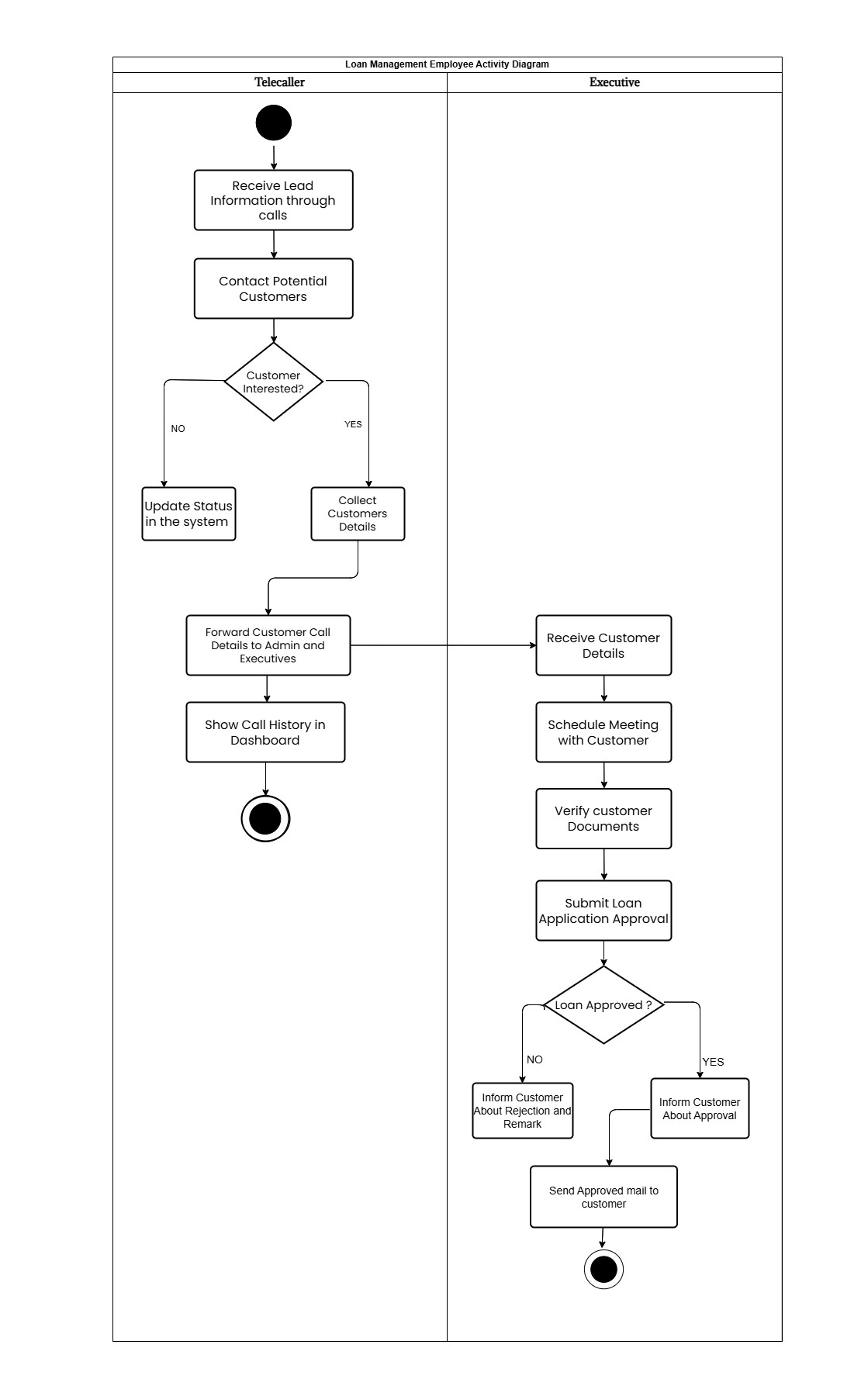
**Admin Activity Diagram:**



**Client Activity Diagram:**



**Employee Activity Diagram:**



**Login Activity Diagram**

Login

Enter UserName and Password

Validate UserName and Passward

V

a

l

i

d

Invalid

Error

Message

Display HomePage

**Registration Activity Diagram**

Registration

Enter

Name,Contactno,Emailid,Password,etc

Validate Data

User Registered

Error

Message

V

a

l

i

d

I

n

v

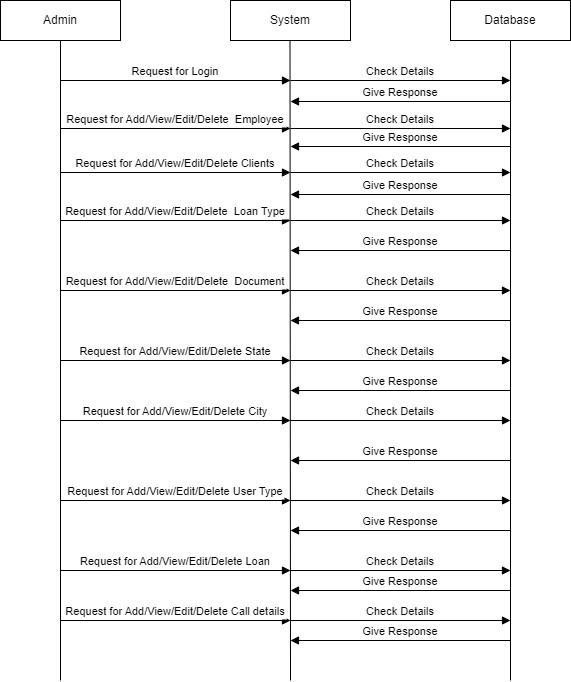
a

li

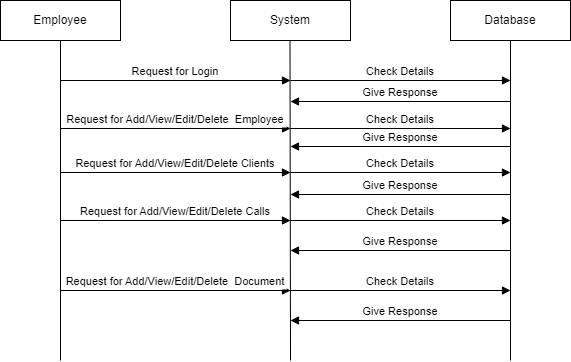
d

**5.2.5 Sequence Diagrams:**

* **Admin:**

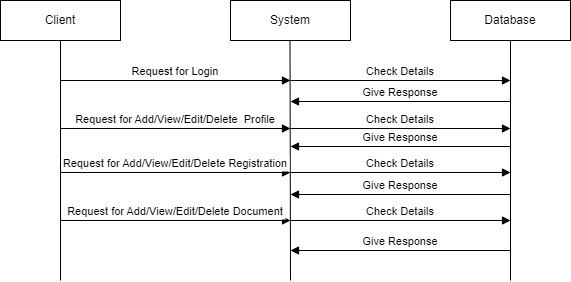


* **Employee:**

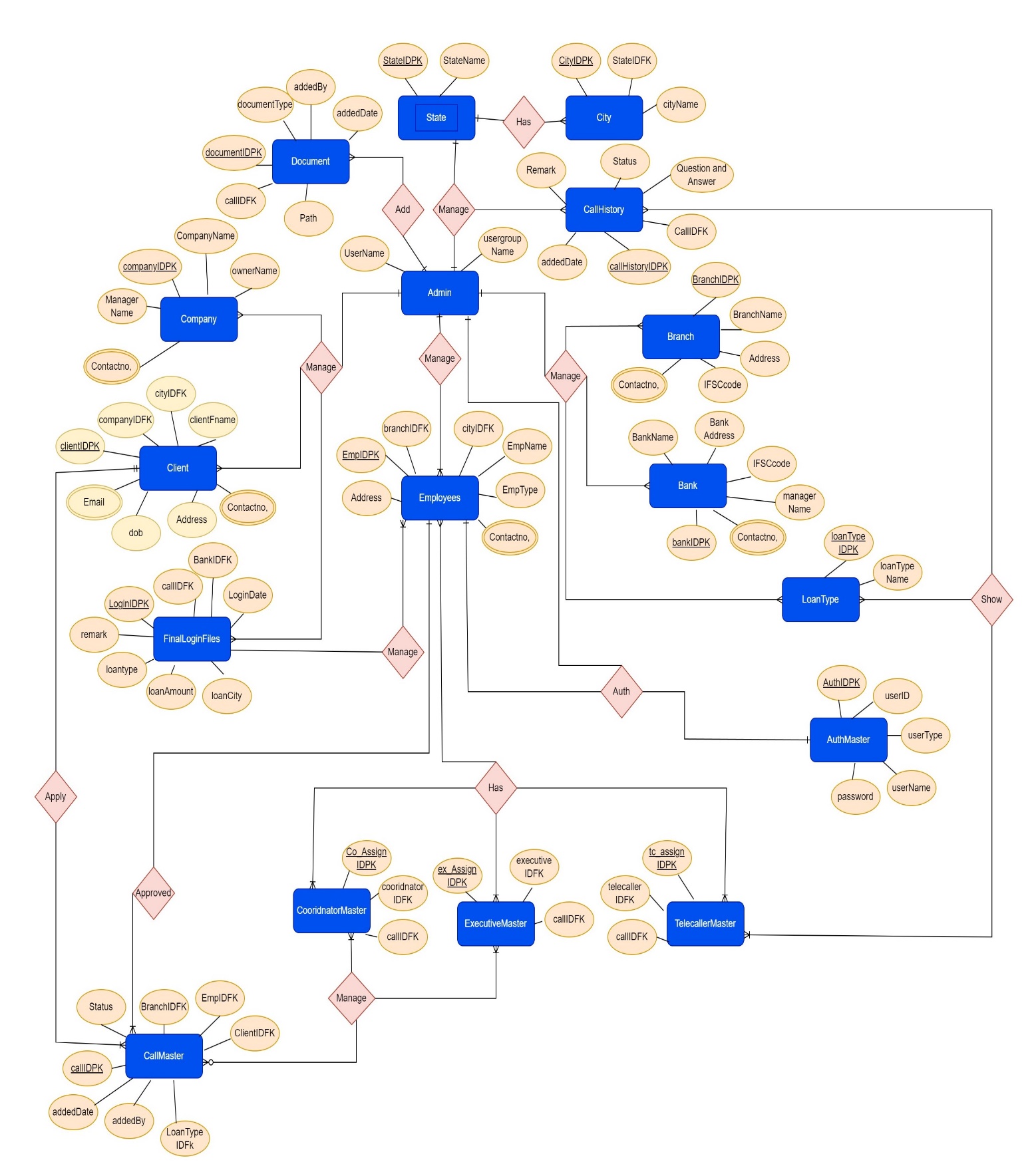


## Client(Customer)

## 



**5.3 E-R Diagram**



**Software Design:**

**6.1 Database Design**

### **State\_Master**

|  |  |  |  |
| --- | --- | --- | --- |
| ELEMENT NAME | ELEMENT DESCRIPTION | DATA TYPES | CONSTRAINT |
| StateIdPK | State id | Int | Primary Key |
| StateName | State Name | Varchar | Not Null |

### **Branch\_Master**

|  |  |  |  |
| --- | --- | --- | --- |
| ELEMENT NAME | ELEMENT DESCRIPTION | DATA TYPES | CONSTRAINT |
| BranchIdPK | Branch id | Int | Primary Key |
| BranchName | Name of Branch | Varchar | Not Null |
| BranchAddress | Address of Branch | Varchar | Not Null |

### **Bank\_Master**

|  |  |  |  |
| --- | --- | --- | --- |
| ELEMENT NAME | ELEMENT DESCRIPTION | DATA TYPES | CONSTRAINT |
| BankIdPK | Bank id | Int | Primary Key |
| BankName | Name of Bank | Varchar | Not Null |
| BankAddress | Address of Bank | Varchar | Not Null |
| IFSCcode | Ifsc code | Varchar | Not Null |
| ManagerName | Name of the manager | Varchar | Not Null |
| ContactNo | ContactNo of the manager | Varchar | Not Null |

### **Company\_Master**

|  |  |  |  |
| --- | --- | --- | --- |
| ELEMENT NAME | ELEMENT DESCRIPTION | DATA TYPES | CONSTRAINT |
| CompanyIdPK | Company id | Int | Primary Key |
| CompanyName | Name of Company | Varchar | Not Null |
| OwnerName | Name of the owner | Varchar | Not Null |
| ContactNo | Contact no | Varchar | Not Null |
| Manager Name | Name of the manager | Varchar | Not Null |

### **Loantype\_Master**

|  |  |  |  |
| --- | --- | --- | --- |
| ELEMENT NAME | ELEMENT DESCRIPTION | DATA TYPES | CONSTRAINT |
| LoanTypeIdPK | LoanType id | Int | Primary Key |
| LoanTypeName | Name of Loan | Varchar | Not Null |

### **Document\_Master**

|  |  |  |  |
| --- | --- | --- | --- |
| ELEMENT NAME | ELEMENT DESCRIPTION | DATA TYPES | CONSTRAINT |
| DocumentIdPK | Document id | Int | Primary Key |
| CallIDFK | CallIDFK | Int | NotNull |
| DocumentType | DocumentType | Varchar | Not Null |
| Path | Image of the  Document | Varchar | Not Null |
| Addedby | addedby | Varchar | Not Null |
| Addeddate | addeddate | Varchar | TimeStamp |

### **UserGroup\_Master**

|  |  |  |  |
| --- | --- | --- | --- |
| ELEMENT NAME | ELEMENT DESCRIPTION | DATA TYPES | CONSTRAINT |
| UserGroupname | User group Name | Varchar | Not Null |
| UserName | User Name | Varchar | Not Null |

### **City\_Master**

|  |  |  |  |
| --- | --- | --- | --- |
| ELEMENT NAME | ELEMENT DESCRIPTION | DATA TYPES | CONSTRAINT |
| CityIdPK | City id | Int | Primary Key |
| StateIdFk | State Id | Int | Foreign Key |
| CityName | City Name | Varchar | Not Null |

### **CallAssignTelecaller\_Master**

|  |  |  |  |
| --- | --- | --- | --- |
| ELEMENT NAME | ELEMENT DESCRIPTION | DATA TYPES | CONSTRAINT |
| CTAssignIdPK | Telecaller id | Int | Primary Key |
| TelecallerIdFk | Employee id | Int | Foreign Key |
| CallIdFk | Call Id | Int | Foreign key |

### **CallAssigncoordinator\_Master**

|  |  |  |  |
| --- | --- | --- | --- |
| ELEMENT NAME | ELEMENT DESCRIPTION | DATA TYPES | CONSTRAINT |
| CCOAssignIdPK | Call Coordinator id | Int | Primary Key |
| CoordinatorIdFk | Employee id | Int | Foreign Key |
| CallIdFk | Call Id | Int | Foreign key |

### **CallAssignExecutive\_Master**

|  |  |  |  |
| --- | --- | --- | --- |
| ELEMENT NAME | ELEMENT DESCRIPTION | DATA TYPES | CONSTRAINT |
| CEAssignIdPK | Call Executive id | Int | Primary Key |
| ExecutiveIdFk | Employee Id | Int | Foreign Key |
| CallIdFk | Call id | Int | Foreign Key |

### **CallHistory\_Master**

|  |  |  |  |
| --- | --- | --- | --- |
| ELEMENT NAME | ELEMENT DESCRIPTION | DATA TYPES | CONSTRAINT |
| CallHistoryIdPK | Call History id | Int | Primary Key |
| CallIdFk | Call Id | Int | Foreign Key |
| Question | Question | Varchar | Not Null |
| Answer | Answer | Varchar | Not Null |
| Status | Status No | Varchar | Not Null |
| Remark | Remark of the call | Varchar | Not Null |
| Addeddate | addeddate | Varchar | Not Null |

### **Employee\_Master**

|  |  |  |  |
| --- | --- | --- | --- |
| ELEMENT NAME | ELEMENT DESCRIPTION | DATA TYPES | CONSTRAINT |
| EmployeeIdPK | Employee id | Int | Primary Key |
| BrancIDFK | Branch id | Int | Foreign Key |
| CityIdFk | City Id | Int | Foreign Key |
| EmployeeFname | Employee first Name | Varchar | NOT NULL |
| EmployeeType | Email Type | Varchar | NOT NULL |
| MobileNo1 | Contact Number | Varchar | NOT NULL |
| MobileNo2 | Contact Number | Varchar | N |
| Address | Address of Employee | Varchar | NOT NULL |
| Email | Email Of Employee | Varchar | NOT NULL |
| Dob | Dob of Employee | Varchar | NOT NULL |
| Gender | Gender of Employee | Varchar | NOT NULL |
| Image | Image of Employee | Varchar | NOT NULL |
| JoiningDate | Date of Employee | Varchar | NOT NULL |

### **Client\_Master**

|  |  |  |  |
| --- | --- | --- | --- |
| ELEMENT NAME | ELEMENT DESCRIPTION | DATA TYPES | CONSTRAINT |
| ClientIdPK | Client id | Int | Primary Key |
| CompanyIdFk | Company Id | Int | Foreign Key |
| CityIdFk | City Id | Int | Foreign Key |
| ClientFname | Employee first Name | Varchar | NOT NULL |
| Mobile1 | Email Type | Varchar | NOT NULL |
| Mobile2 | Contact Number | Varchar | NOT NULL |
| CurrentAddress | Address of Employee | Varchar | NOT NULL |
| PermanentAddress | Email Of Employee | Varchar | NOT NULL |
| Dob | Dob of Employee | Varchar | NOT NULL |
| Email | Gender of Employee | Varchar | NOT NULL |
| Salary | Image of Employee | Varchar | NOT NULL |
| Gender | Gender | Varchar | NOT NULL |
| Image | Images of Customer | Varchar | NOT NULL |

### **ExisitingLoan\_Master**

|  |  |  |  |
| --- | --- | --- | --- |
| ELEMENT NAME | ELEMENT DESCRIPTION | DATA TYPES | CONSTRAINT |
| ExsitingLoanIdPK | Employee id | Int | Primary Key |
| BankIdFk | Bank id | Int | Foreign Key |
| ClientIDFK | Client id | Int | Foreign Key |
| CallIDFK | Call id | Int | Foreign Key |
| LoanType | LoanType | Varchar | Not NULL |
| CallIdFk | Call Id | Int | Foreign Key |
| ClientIdFk | Client Id | Int | Foreign Key |
| EMI | EMI | Varchar | NOT NULL |
| PendingEMI | Pendingemi | Varchar | NOT NULL |
| Interest | Interest | Varchar | NOT NULL |
| Amount | Total Amount | Varchar | NOT NULL |
| Tenure | Tenure | Varchar | NOT NULL |
| PendingEMI | Pending EMI | Varchar | NOT NULL |
| Intrest | Intrest | Varchar | NOT NULL |
| Amount | Amount | Varchar | NOT NULL |
| Tenure | Tenure | Varchar | NOT NULL |
| Outstanding | Outstanding | Varchar | NOT NULL |

### **Login\_Master**

|  |  |  |  |
| --- | --- | --- | --- |
| ELEMENT NAME | ELEMENT DESCRIPTION | DATA TYPES | CONSTRAINT |
| LoginIdPK | Login id | Int | Primary Key |
| CallIdFk | Call Id | Int | Foreign Key |
| BankIdFk | Bank Id | Int | Foreign Key |
| LoginDate | Login date | Varchar | Foreign Key |
| LoginCity | Login City | Varchar | NOT NULL |
| LoanAmount | Loan Amount | Varchar | NOT NULL |
| LoanType | Loan Type | Varchar | NOT NULL |
| Remark | Remark | Varchar | NOT NULL |
| Tenure | Tenure | Varchar | NOT NULL |
| RateOfIntrest | Rate of Intrest | Varchar | NOT NULL |
| ProvidentFund | Pf | Varchar | NOT NULL |
| Insurance | Insurance | Varchar | NOT NULL |
| FinalRemark | Final Remark | Varchar | NOT NULL |
| RejectedBy | Rejected By | Varchar | NOT NULL |
| Status | Status | Varchar | NOT NULL |
| DisburseDate | Date | Varchar | NOT NULL |
| AddedBy | addedby | Varchar | NOT NULL |
| AddedDate | addeddate | Varchar | NOT NULL |

### **Call\_Master**

|  |  |  |  |
| --- | --- | --- | --- |
| ELEMENT NAME | ELEMENT DESCRIPTION | DATA TYPES | CONSTRAINT |
| CallIdPK | Login id | Int | Primary Key |
| BranchIdFk | BranchId | Int | Foreign Key |
| EmployeeIdFk | TeamLeaderid | Int | Foreign Key |
| ClientIDFK | Client id | Int | Foreign Key |
| LoanTypeIDFK | Loantypeid | Int | Foreign Key |
| AddedBy | AddedBy | Varchar | NOT NULL |
| AddedDate | AddedDate | Varchar | NOT NULL |

### **Auth\_Master**

|  |  |  |  |
| --- | --- | --- | --- |
| ELEMENT NAME | ELEMENT DESCRIPTION | DATA TYPES | CONSTRAINT |
| AuthIdPK | Auth id | Int | Primary Key |
| Username | User Name | Varchar | NOT NULL |
| Password | Password | Varchar | NOT NULL |
| UserType | UserType | Varchar | NOT NULL |
| UserId | Clientid | Varchar | NOT NULL |

**6.2 Interface Design sitemap followed with page snapshots**

### **6.3Architecture Design**

The MVC architectural pattern has existed for a long time in software engineering. All most all the languages use MVC with slight variation, but conceptually it remains the same.

MVC stands for Model, View, and Controller. MVC separates an application into three components - Model, View, and Controller.

Model: Model represents the shape of the data. A class in C# is used to describe a model. Model objects store data retrieved from the database.

Model represents the data.

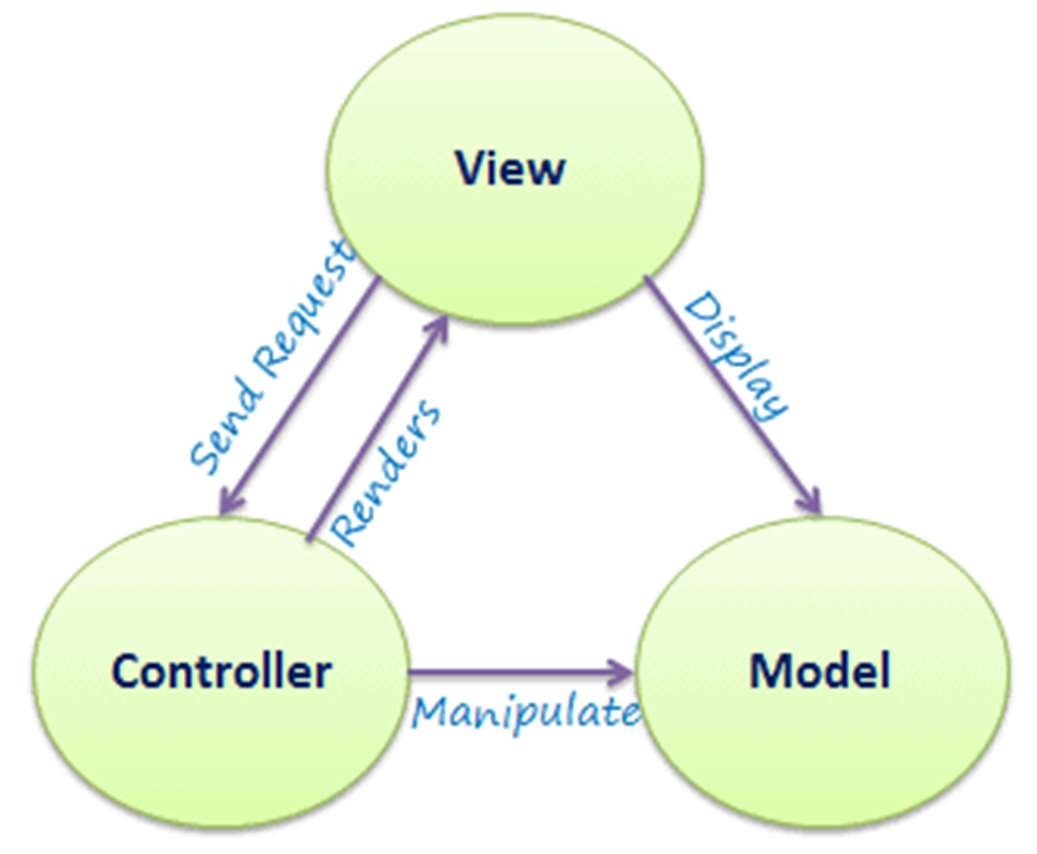
View: View in MVC is a user interface. View display model data to the user and also enables them to modify them. View in ASP.NET MVC is HTML, CSS, and some special syntax (Razor syntax) that makes it easy to communicate with the model and the controller.

View is the User Interface.

Controller: The controller handles the user request. Typically, the user uses the view and raises an HTTP request, which will be handled by the controller. The controller processes the request and returns the appropriate view as a response.

Controller is the request handler.

The following figure illustrates the interaction between Model, View, and Controller.



### **7. Testing**

**7.1 Unit Testing**

This testing focuses on verification effort on the smallest unit of software design-the component or module. Using the component - level design description as s guide, important control paths are tested to uncover errors within the boundary of the module. The relative complexity of tests and the error those tests uncover is limited by constrained scope established for the unit testing . The unit testing focuses on internal processing logic and data structure with boundaries of the component. This type of testing can be conducted in parallel for multiple components.

**7.2 Integration Testing**

* Once the modules were unit tested, integration testing was carried out to ensure that the modules worked properly when interfaced with each other.
* The main objective of integration testing was to ensure that ,
* No data was lost across the interface.
* No side effects were encountered while integrating the modules.
* Sub functions when combined produced the desired result.
* Numbers of input parameters are equal to number of arguments.
* Tests scenarios which were created in a specific format to carry out functional testing were extensively used during this stage.
* Each time a new module was added as a chapter of integration testing, regression testing was carried to ensure that changes have not propagated any unidentified side effects.
* **Admin Side:**

|  |  |
| --- | --- |
| Test Case | 1 |
| Description | Login |
| Pre-Condition | Admin should know username and password. Username and password should not be empty. |
| Test action | Allow access or deny access |
| Error Message | Invalid user. |

|  |  |
| --- | --- |
| Test Case | 2 |
| Description | View/edit/delete data |
| Pre-Condition | Admin should login itself for view/edit/delete the data |
| Test action | Edit/View/Deletion of data |
| Error Message | Please enter details. |

|  |  |
| --- | --- |
| Test Case | 3 |
| Description | Add/Edit/Delete LoginFile |
| Pre-Condition | LoginFIle details |
| Test action | New LoginFile insert |
| Error Message | Invalid LoginFile details |

**Employee(Telecaller)**

|  |  |
| --- | --- |
| Test Case | 1 |
| Description | Login |
| Pre-Condition | Admin should know username and password. Username and password should not be empty. |
| Test action | Allow access or deny access |
| Error Message | Invalid user. |

|  |  |
| --- | --- |
| Test Case | 2 |
| Description | View data |
| Pre-Condition | Admin should login itself for view the data |
| Test action | View of data |

|  |  |
| --- | --- |
| Test Case | 3 |
| Description | Add Call |
| Pre-Condition | Calldetails |
| Test action | New Call insert |
| Error Message | Invalid Call details |

**Employee(Executive)**

|  |  |
| --- | --- |
| Test Case | 1 |
| Description | Login |
| Pre-Condition | Admin should know username and password. Username and password should not be empty. |
| Test action | Allow access or deny access |
| Error Message | Invalid user. |

|  |  |
| --- | --- |
| Test Case | 2 |
| Description | View data |
| Pre-Condition | Admin should login itself for view the data |
| Test action | View of data |

|  |  |
| --- | --- |
| Test Case | 3 |
| Description | Add Call |
| Pre-Condition | Call details |
| Test action | New Call insert |
| Error Message | Invalid Call details |

**8. Future Enhancement**

* Emi payment.

**9. Glossary**

 Dependency in a project network is a link amongst a project's terminal elements.

 Duration of a project's terminal element is the number of calender periods it takes from the time to execution of element starts to the moment it is completed.

 Management in business and human organization activity is simply the act of getting people together to accomplish desired goals. Management comprises planning, organizing , staffing , leading or detection and controlling and organization or efforts for the purpose of accomplishing a goal.

 Planning in organization and public policy is both the organizational process of creating and maintaining a plan and the psychological process of thinking about the activites required to create a desired goal on some scale.

 Project A temporary endeavor undertaken to create a unique product, service or result.

 Risk management is a management specialism aiming to reduce different risks related to preselected domain to the level accepted by society . It may refer to numerous types of threts caused by environment,technology ,humans, organization and politics.

 Scope of a project in project management is the sum total of all of its products and their requirements or features.

 Timeline is a graphical representation of a chronological sequence of events , aslo referred to as a chronology . It can also mean a schedule of activities, such as a timetable.

 Work in project management is the amount of effort applied to produce a deliverable or to accomplish a task(a terminal element).

**10 References:**

* [**https://stackoverflow.com/**](https://stackoverflow.com/)
* **https://themeforest.net/**
* **https**[**://ww**](http://www.codexworld.com/)**w.c**[**odexworld.com/**](http://www.codexworld.com/)
* **https://developers.google.com/**