

|  |
| --- |
| The Rural and Agri Loan Finance Services |

**Index Page:**

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Title** | **Page No.** |
| Chapter 1 | **Introduction** | **03** |
| 1.1 | **Operating Environment – Hardware and Software** | **08** |
| 1.2 | **Detail Description of Technology Used** | **09** |
| Chapter 2 | **Proposed System** | **15** |
| 2.1 | **Proposed System** | **15** |
| 2.2 | **Objectives of System** | **17** |
| 2.3 | **User Requirements** | **18** |
| Chapter 3 | **Analysis And Design** | **21** |
| 3.1 | **Project Architecture Diagram** | **21** |
| 3.2 | **Data Flow Diagram** | **21** |
| 3.3 | **Entity Relationship Diagram** | **22** |
| 3.4 | **Use Case Diagram** | **23** |
| 3.5 | **Activity Diagram** | **24** |
| 3.6 | **Collaboration Diagram** | **27** |
| 3.7 | **Table Specifications** | **27** |

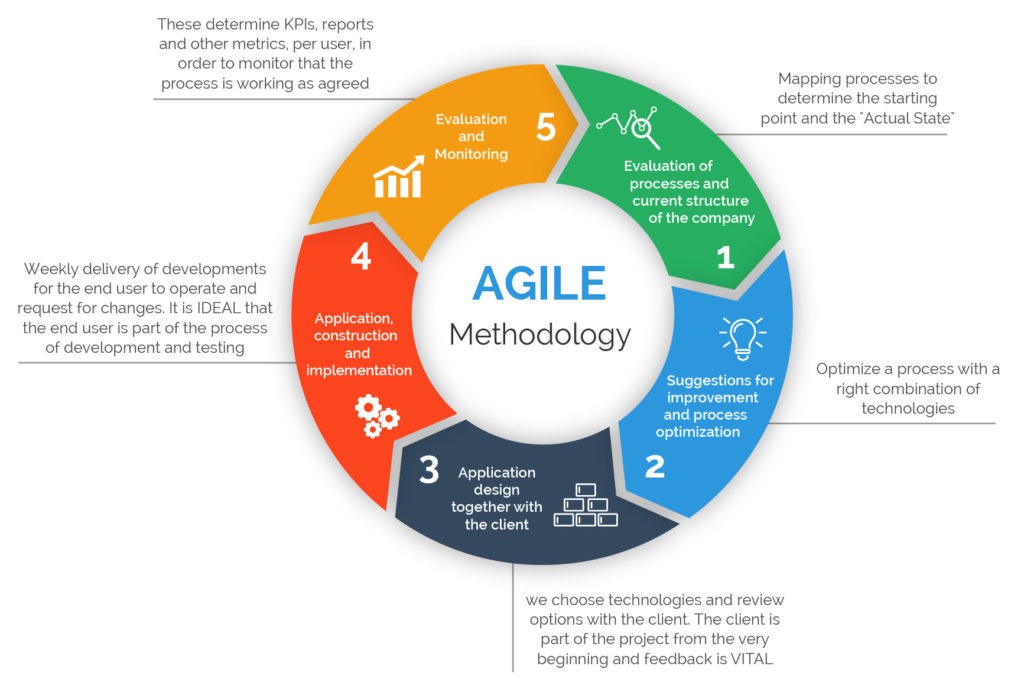
**Chapter 1. Introduction**

* 1. **What is The** **Rural an Agri Finance?**

Rural an Agri Finance is committed to supporting farmers and agricultural businesses across the nation in facing the unforeseen challenges of the future. Rural an Agri provides financial solutions to the farming community to help them enter a better life. Our Relationship Managers understand their needs and help them choose the right type of loan..

* **Concept And Features of Rural an Agri -finance:**

1. Loans for purchase of Tractors and farming equipments
2. Quick processing and hassle-free documentation
3. Convenient and easy repayment options
4. Enabling ownership opportunities for the Rural-Agri class
   1. **Methodology Used For Web Site Development: Agile Methodology**



* 1. **Modules:**

* Enquiry
* Customer Creation
* Cibil Score Check
* Customer Verification
* Sanction Letter
* Disbursement
* Ledger

**Important Roles:**

* **Relationship Executive:**

Relationship executives identify and pursue business relationships with corporate and business unit executives. They create new business opportunities through customer relationships, while also providing value to customers through professional services agreements and solution sales. Relationship executives also establish strategic relationships with internal departments, including sales, marketing and business development teams. They use their internal and external relationships to develop sales strategies and service offerings that enable new opportunities for business and revenue growth.

* **Operational Executive:**

The Operations Executive is responsible for the part of the workplace team that directly manages the operations and maintenance of facilities. The Operations Executive can report to various parts of the company such as the Facilities Executive, Chief Financial Officer or Chief Operations Officer, but usually has direct access to senior management.

* **Account Head:**

Account head is like the name of person, place or books etc. As we know that lots of transactions take place in a business organization. Now question arises how to recognize those transactions in some definite terms? To recognize the various transactions, the account heads are created. Account head is a name under which particular types of transactions are recorded.

* **Branch Manager:**

A Branch Manager is responsible for coordinating and overseeing all branch operations, including staff hiring, training, and supervision. They come up with appropriate strategies to increase productivity and performance levels in order to achieve the branch’s financial targets.

Furthermore, besides their managerial duties, they are also responsible for attracting new customers and partners, as well as retaining the already existing ones.

* **Credit Manager:**

Credit management is the process of granting credit, setting the terms it's granted on, recovering this credit when it's due, and ensuring compliance with company credit policy, among other credit related functions. The goal within a bank or company in controlling credit is to improve revenues and profit by facilitating sales and reducing financial risks.

A credit manager is a person employed by an organization to manage the credit department and make decisions concerning credit limits, acceptable levels of risk, terms of payment and enforcement actions with their customers

* **Loan Disbursement:**

Disbursement is the act of paying out or disbursing money. Examples of disbursements include money paid out to run a business, cash expenditures, dividend payments, or the amounts that a lawyer might have to pay out on a person's behalf in connection with a transaction. Disbursing money is part of cash flow. If cash flow is negative, meaning that disbursements are higher than revenues, it can be an early warning of potential insolvency.

* **Master Module:**

The module provides utilities for controlling the module status of all modules in the whole installation by a single command. The configuration is done by defining "master modules". "Master modules" are (in most cases) custom modules or features, that provide dependencies to all modules that are needed to run the site

**ABSTRACT**

Rural an Agri Finance is committed to supporting farmers and agricultural businesses across the nation in facing the unforeseen challenges of the future. Rural an Agri provides financial solutions to the farming community to help them enter a better life. Our Relationship Managers understand their needs and help them choose the right type of loan...

**1.2 OPERATING ENVIRONMENT – HARDWARE AND SOFTWARE**

* **Hardware Requirements for Server :**

**Processor**  : Pentium IV 1.8 GHz

**Memory size** : 2 GB RAM

**Storage**  : 40 GB Hard Disk

**Display** : EGA/VGA Color Monitor, 600 x 800 Pixels Resolution, High Color

**Internet Connection** : Required

**Key Board** : Any with minimum required keys

**Mouse** : Any

* **Software Requirements :**

**Operating System** : Windows 7 and above

**Front-End Tool** : HTML, CSS, JS, React JS

**Back-End Tool** : MySQL 8.0

* **Hardware Requirements for Client :**

**Processor** : Pentium III 800 MHz

**Memory size** : 2 MB RAM

**Storage** : 40 GB Hard Disk

**Display** : EGA/VGA Color Monitor, 600 x 800

Pixels Resolution, High Color

**Key Board** : Any with minimum required keys

**Internet Connection** : Required

**Mouse** : Any

**1.3 DETAIL DESCRIPTION OF TECHNOLOGY USED**

* **MySQL:**

**MySQL** is a relational database management system based on SQL – Structured Query Language. The application is **used** for a wide range of purposes, including data warehousing, e-commerce, and logging applications. The most common use for **MySQL** however, is for the purpose of a web database.

**MySQL** is not a **programming language**. Instead, it is a relational database management system (RDBMS). It is used to store data, not to write programs. The SQL **programming language** can be used to program a **MySQL** database.

* **DJANGO:**

**Django** is a high-level Python open-source web framework that encourages rapid development and clean, pragmatic design. **Django** can speed up development of a web application being built in the Python programming language. **Django**  follows the model–template–views architectural pattern.

* **DJANGO REST FRAMEWORK:**

**Django** REST framework is a powerful and flexible toolkit for building Web APIs. Some reasons you might want to use REST framework: The Web browsable API is a huge usability win for your developers. Authentication policies including packages for OAuth1a and OAuth2.

* **REACTJS:**

React.js is an open-source JavaScript library that is used for building user interfaces specifically for single-page applications. It’s used for handling the view layer for web and mobile apps. React also allows us to create reusable UI components.

React allows developers to create large web applications that can change data, without reloading the page. The main purpose of React is to be fast, scalable, and simple.

**FEATURES AND BENEFITS:**

**CROSS PLATFORM**

**1. Progressive Web Apps**

Use modern web platform capabilities to deliver app-like experiences. High performance, offline, and zero-step installation.

**2. Native**

Build native mobile apps with strategies from Cordova, Ionic, or Native Script.

**3. Desktop**

Create desktop-installed apps across Mac, Windows, and Linux using the same ReactJS you've learned for the web plus the ability to access native OS APIs.

* **SPEED AND PERFORMANCE**

**1. Code Generation**

ReactJS turns your templates into code that's highly optimized for today's JavaScript virtual machines, giving you all the benefits of hand-written code with the productivity of a framework.

**2. Code Splitting**

ReactJS load quickly with the new Component Router, which delivers automatic code-splitting so users only load code required to render the view they request.

**Chapter 2. Proposed System**

2.1 Proposed System

The proposed software will solve all the problems they are facing now. Loan is still perceived as a huge burden for many countries. While loans are known to take off that burden off, it still gives nightmare to people and they avoid availing a personal loan. However, a loan can be a solution to many of your financial problems.

**2.1.1 Purpose of proposed system**

* Interest rates and the loan details are also available at the click of a mouse.
* Customer can apply for a loan and after approved it they can track their details from online.
* This system provides detail about the customers, their loan details, EMI details and its rate details.
* System provides download option to download different type of loan form in MS word document.
* Using with this system admin can find customer easily and it’s a paperless system so workload is reduced.
* The decision process becomes faster and more consistent.
* After registration and login customer can use the system easily and also customer can view any query about loan details as well as EMI details in their profile. So this system saves time.
* Provides good communication for the customer.
* In this system there are used EMI (Equated Monthly instalment) calculators.
* Provides a facility to generate the reports very.

**2.1.2 Advantages of the proposed System**

* Entire activities of the show room are recorded through the system.
* Customer Data is maintained.
* Reports generated will be more useful for management to take the quick business decisions.
* Customer database is maintained which will be helpful for intimating the service completion details and new offers
* Customer follow-ups are maintained which will be an added advantage of this system.
* It helps with your CIBIL Score as CIBIL or Credit Scores are a summary of customer history in loan credits and repayments over a period of time.

**2.2 Objectives of System**

During the past several decades personnel function has been transformed from a relatively obscure record keeping staff to central and top level management function. There are many factors that have influenced this transformation like technological advances, professionalism, and general recognition of human beings as most important resources.

A computer based management system is designed to handle all the primary information required to calculate monthly statements of Customer Record which include monthly statement of any month. Separate database is maintained to handle all the details required for the correct statement calculation and generation.

This project intends to introduce more user friendliness in the various activities such as record updating, maintenance, and searching. The searching of record has been made quite simple as all the details of the Customer can be obtained by simply keying in the identification of that Customer. Similarly, record maintenance and updating can also be accomplished by using the identification of Employee with all the details being automatically generated. These details are also being promptly automatically updated in the master file thus keeping the record absolutely up-to-date.

The entire information has maintained in the database or Files and whoever wants to retrieve can’t retrieve, only authorization user can retrieve the necessary information which can be easily be accessible from the file. The main objective of the entire activity is to automate the process of day-to-day activities of pay.

* **Major Functionalities:**

The system is very simple in design and to implement. The system requires very less system resources and the system will work in almost all configurations. It has got

* **Following features :-**

1) Ensure data accuracy’s.

2) Proper control of the higher officials.

3) Minimize manual data entry.

5) Minimum time needed for the various processing.

6) Greater efficiency.

7) Better service.

8) User friendliness and interactive.

9) Minimum time required.

**2.3 User Requirements**

The system after careful analysis has been identified to be presented with the following modules:

* Loan Application (Enquiry) Module
* CIBIL Check Module
* Loan Sanction Module
* Loan Disbursement Module
* Ledger Creation Module

**Loan Application Module**

* **Relationship Executive:**

Relationship executives identify and pursue business relationships with corporate and business unit executives. They create new business opportunities through customer relationships, while also providing value to customers through professional services agreements and solution sales. Relationship executives also establish strategic relationships with internal departments, including sales, marketing and business development teams. They use their internal and external relationships to develop sales strategies and service offerings that enable new opportunities for business and revenue growth.

* Admin manage loan application which is sent by customer using the system.

**Operational Executive:**

The Operations Executive is responsible for the part of the workplace team that directly manages the operations and maintenance of facilities. The Operations Executive can report to various parts of the company such as the Facilities Executive, Chief Financial Officer or Chief Operations Officer, but usually has direct access to senior management.

**CIBIL Check Module**

**Operational Executive:**

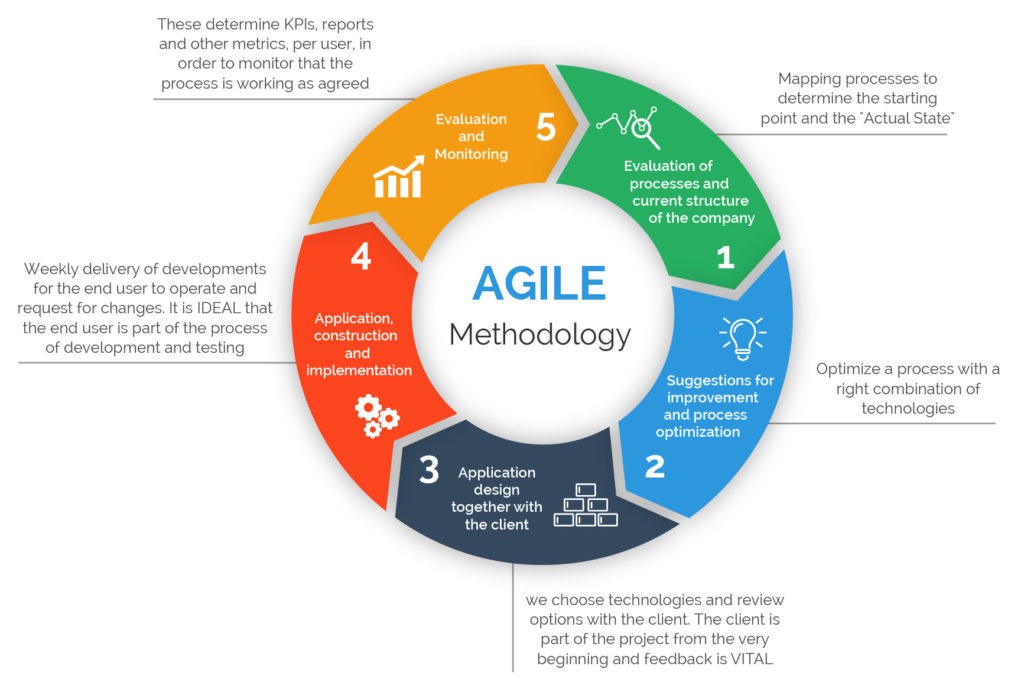
After receiving customers basic information from Relational Executive, the Operational Executive sends that data to Third Party API for CIBIL Check. Third Party API checks the CIBIL Score of Customer on the basis of PanCard number and sends the CIBIL Score to Operational Executive. Upon checking CIBIL Score Operational Executive Puts the Appropriate Remark (Whether the Customer is eligible for taking loan or not) and sends it back to Relational Executive. Relational Executive tells the customer about his eligibility of taking loan and if customer is eligible and interested for taking loan then Collects additional information and documents from customer.

**Loan Sanction Module**

Operational executive will create a sanction letter and send to relationship executive. Relationship executive will send the letter to customer and get customer confirmation and send it back to Operational executive. Now Operational executive will send all loan details and customer details to Account head. Account head will check the disburse amount and dealer details such as account no , bank name, bank IFSC code and transfer the amount to dealer bank account. Hence loan amount is disbursed and disbursement list is created.

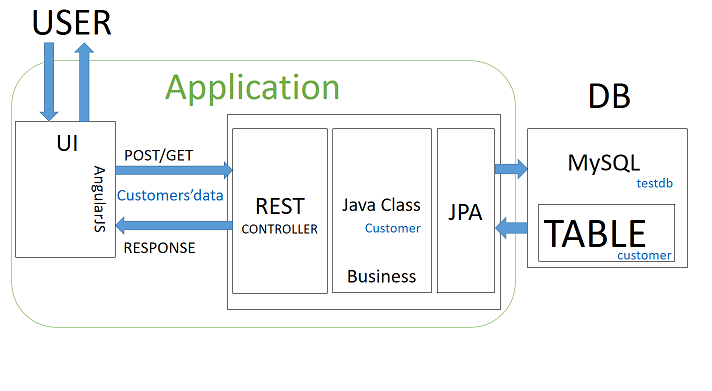
* **Methodology Used for Web Application Development:**

Agile Methodology:

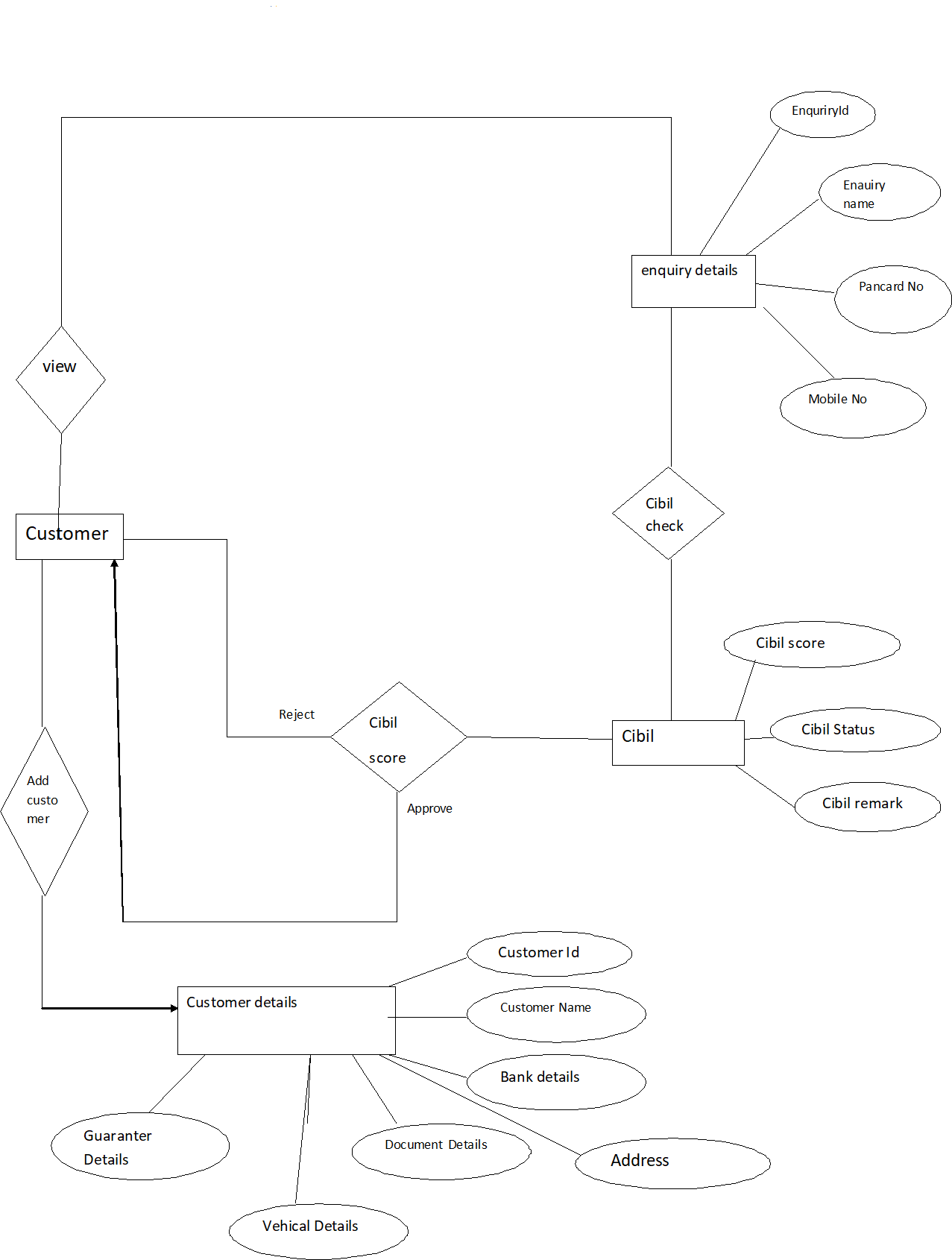


**Chapter 3. Analysis and Design**

**3.1 Project Architecture:**

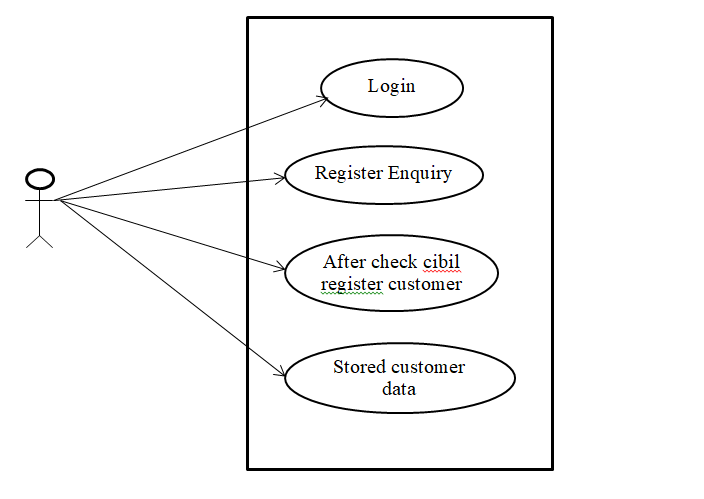


**3.2** **Entity Relationship Diagram:**

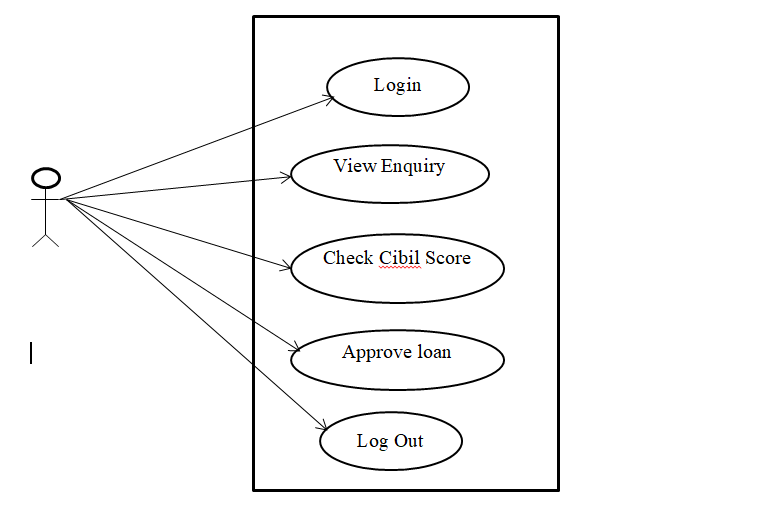


**3.4 USE CASE DIAGRAM**

**RE Login:**



**OE Login:**



**3.5 Activity Diagram:**

**RE Login:**

Log in

**No**

**Yes**

New Enquiry

Log out

Add Customer

Enquiry List

**OE Login:**

Log in

No

Yes

Log out

Enquiry List

New Enquiry

Document List

Sanction Letter

**CM Login:**

Log in

**No**

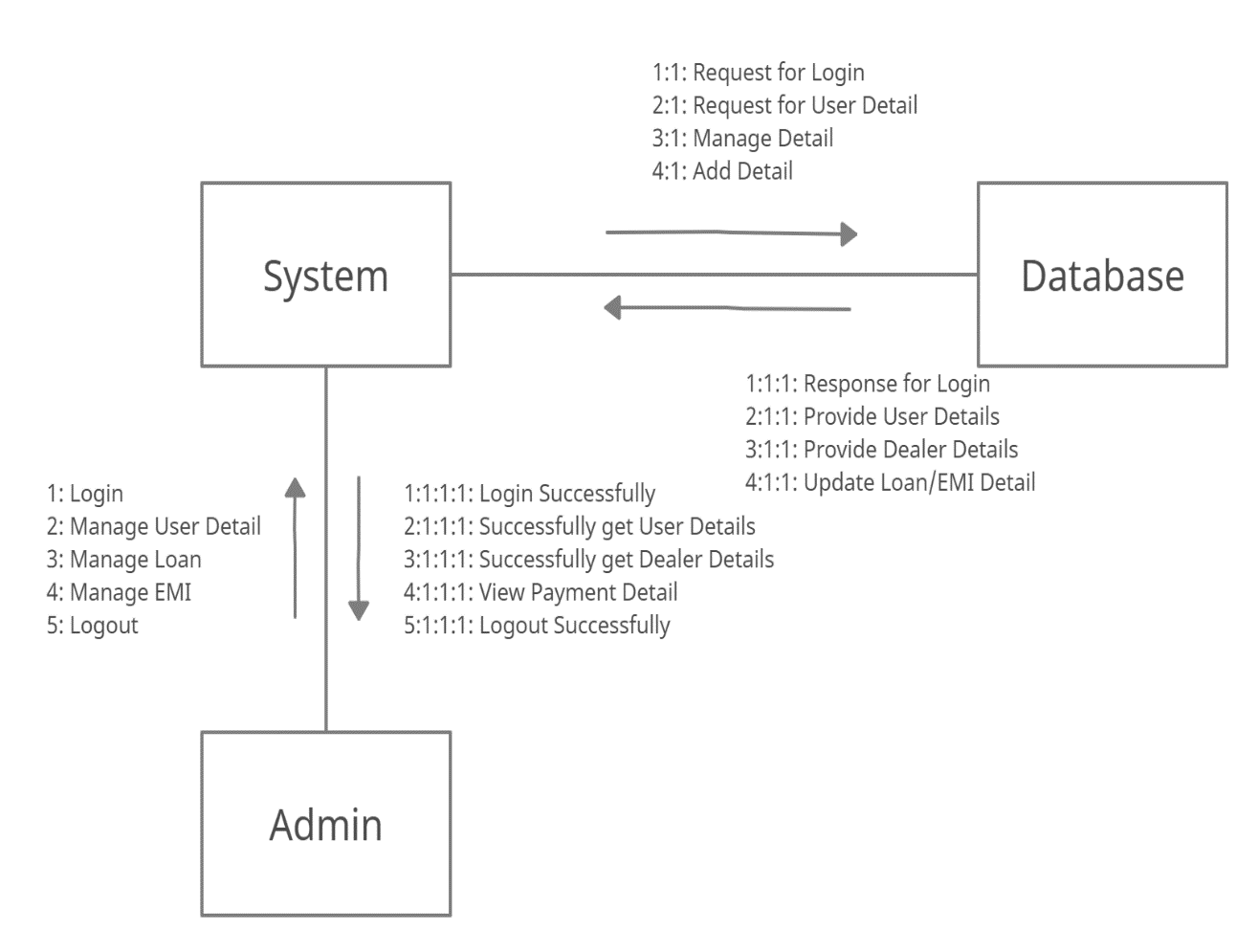
**Yes**

Verification

Log out

**3.6 Collaboration Diagram:**

**Admin Side:**



**3.7 Table Specification:**

**Table Name : Enquiry\_Details**

**Primary Key :** enq\_id

**Description :**These table manage the Enquiry(Customer’s) Details Information

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraint** | **Description** |
| **enq\_id** | Integer | Primary Key | Stores enquiry ID |
| **enq\_name** | String | Not Null | Stores name. |
| **enqpancardno** | String | Not Null | Stores Pancard Number. |
| **enq\_mobileno** | String | Not Null | Stores Mobile number. |
| **enq\_age** | Integer | Not Null | Stores age of customer. |
| **enq\_status** | String | Not Null | Stores status. |
| **enq\_email** | String | Not Null | Stores email address. |
| **enq\_vehicle\_name** | String | Not Null | Stores name of vehicle. |

**Table Name : Cibil Score**

**Primary Key :** CibilId

**Description :**These table manage the Cibil’s(Customer’s) Details Information.

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraint** | **Description** |
| **cibilId** | Integer | Primary Key | Stores Cibil ID |
| **customerPanNo** | String | Not Null | Stores Pancard no. |
| **Score** | Integer | Not Null | Stores cibil score. |
| **Cibil\_status** | String | Not Null | Stores cibil status. |
| **Remark** | String | Not Null | Stores remark. |

**Table Name : Customer Details**

**Primary Key :** CustomerId

**Description :**These table manage the Customer’s Details Information.

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraint** | **Description** |
| **customerId** | Integer | Primary Key | Stores Customer id. |
| **customerName** | String | Not Null | Stores customer name. |
| **customerMobileno** | Integer | Not Null | Stores mobile no. |
| **customerLoanAmount** | Integer | Not Null | Stores loan amount. |
| **customerAge** | Integer | Not Null | Stores customer age. |
| **customerDateofBirth** | String | Not Null | Stores Dob of customer. |
| **customerEmail** | String | Not Null | Stores email address. |
| **customerPancardno** | String | Not Null | Stores pancard number |
| **Address** | Address | Foreign Key | It gives reference to the addressID. |
| **Vehical** | VehicalDetails | Foreign Key | It gives reference to the vehicleID. |
| **Bank** | Bank | Foreign Key | It gives reference to the bankId. |
| **Guaranter** | GuaranterDetails | Foreign Key | It gives reference to the guaranterId. |
| **loandetails** | Previousloandetails | Foreign Key | It gives reference to the preloanId. |

**Table Name : Address Details**

**Primary Key :** addressId

**Description :**These table manage the Address Details Information.

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraint** | **Description** |
| **addressId** | Integer | Primary Key | Stores address id. |
| **localHouseNo** | String | Not Null | Stores local house no. |
| **localArea** | String | Not Null | Stores local area. |
| **localLandmark** | String | Not Null | Stores local landmark. |
| **Localcity** | City | Foreign Key | It gives reference to the cityId. |
| **permanantHouseNo** | String | Not Null | Stores permanent houseno |
| **permanantArea** | String | Not Null | Stores permanent area. |
| **permanantLandmark** | String | Not Null | Stores permanent landmark. |
| **permananatCity** | City | Foreign Key | It gives reference to the cityId. |
| **Country** | Country | Foreign Key | It gives reference to the countryId. |

**Table Name : Guaranter Details**

**Primary Key :** guaranterId

**Description :**These table manage the Guaranter’s Details Information.

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraint** | **Description** |
| **guaranterId** | Integer | Primary Key | Stores guaranterId. |
| **guranterName** | String | Not Null | Stores guaranter name. |
| **guaranterAddress** | String | Not Null | Stores guaranter address |
| **guaranterRealtionshipwithcustomer** | String | Not Null | Stores relationship with customer. |
| **guaranterMobileNo** | Long | Not Null | Stores mobile no. |
| **aadharCardNo** | Long | Not Null | Stores aadhar card number. |
| **jobDetails** | String | Not Null | Stores job description. |

**Table Name : Previous Loan Details**

**Primary Key :** previousloanId

**Description :**These table manage the Previous loan Details Information.

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraint** | **Description** |
| **previousloanId** | Integer | Primary Key | Stores loan id. |
| **previousLoanamount** | Integer | Not Null | Stores previous loan amount. |
| **previousLoanStatus** | String | Not Null | Stores loan status. |
| **Tenure** | Integer | Not Null | Stores tenaure. |
| **paidAmount** | Integer | Not Null | Stores paid amount. |
| **remainingAmount** | Integer | Not Null | Stores remaining amount. |
| **bank** | Bank | Foreign Key | It reference to the bankId. |

**Table Name : Bank Details**

**Primary Key :** bankId

**Description :** These table manage the Bank’s Details Information.

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraint** | **Description** |
| **bankId** | Integer | Primary Key | Stores bank id. |
| **bankName** | String | Not Null | Stores bank name. |
| **accountNumber** | Long | Not Null | Stores account number. |
| **bankIfsc** | String | Not Null | Stores bank ifsc code. |
| **bankMicr** | String | Not Null | Stores bank micr code. |
| **bankAddress** | String | Not Null | Stores bank address. |

**Table Name : Document Details**

**Primary Key :** documented.

**Description :**These table manage the Document’s Details Information.

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraint** | **Description** |
| **documentId** | Integer | Primary Key | Stores document id. |
| **customerId** | Integer | Not Null | Stores customer id. |
| **Pancard** | Byte[] | Not Null | Stores pancard image. |
| **Aadharcard** | Byte[] | Not Null | Stores aadharcard image. |
| **Phote** | Byte[] | Not Null | Stores photo. |
| **Signature** | Byte[] | Not Null | Stores signature. |
| **Postdatedcheque** | Byte[] | Not Null | Stores postdated cheque. |
| **Thumb** | Byte[] | Not Null | Stores thumb image. |
| **Bankstatement** | Byte[] | Not Null | Stores bank statement image. |
| **itrfile** | Byte[] | Not Null | Stores itr file image. |
| **Salaryslip** | Byte[] | Not Null | Stores salary slip image. |

**Table Name : Vehicle Details**

**Primary Key :** vehicalId

**Description :**These table manage the Vehicle’s Details Information.

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraint** | **Description** |
| **vehicalId** | Integer | Primary Key | Stores vehicle Id. |
| **modelNo** | String | Not Null | Stores model name. |
| **Dealer** | String | Not Null | Stores dealer name. |
| **Price** | String | Not Null | Stores price of vehical. |
| **onRoadPrice** | String | Not Null | Stores on road price of vehical. |

**Table Name: City**

**Primary Key:** cityId

**Description:** This table manage the City’s Information

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type(Size)** | **Constraint** | **Description** |
| **cityId** | Integer | Primary Key | Store City ID |
| **cityName** | String | Not Null | Stores City Name |
| **cityPincode** | Int | Not Null | Stores City pincode. |
| **stated** | State | Foreign Key | It gives references to the stated |

**Table Name: State**

**Primary Key:** stateId

**Description:** This table manage the All-State Related Information

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraint** | **Description** |
| **stated** | Integer | Primary Key | Stores State ID |
| **stateName** | String | Not Null | Stores State Name |
| **countryId** | Country | Foreign Key | It gives reference to country \_id |

**Table Name: Country**

**Primary Key:** countryId

**Description:** This table manage the Countries’ Information

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraint** | **Description** |
| **countryId** | Integer | Primary Key | Stores Country ID(Auto increment) |
| **countryName** | Varchar (40) | Not Null | Stores Country Name |