* **Chapter 2: - Analysis**

Analysis is the systematic way of examining and evaluating data or information, by breaking it into components parts to uncover their interrelationships. For the development of any system analysis is the most important part. Without analysis the requirements of the system or without knowing anything about the system, if the program starts there is high chance that the system will fail. Different research is done before starting any projects. Basically, analysis is the process of breaking a complex topic or substance into smaller parts in order to gain better understanding of the project.

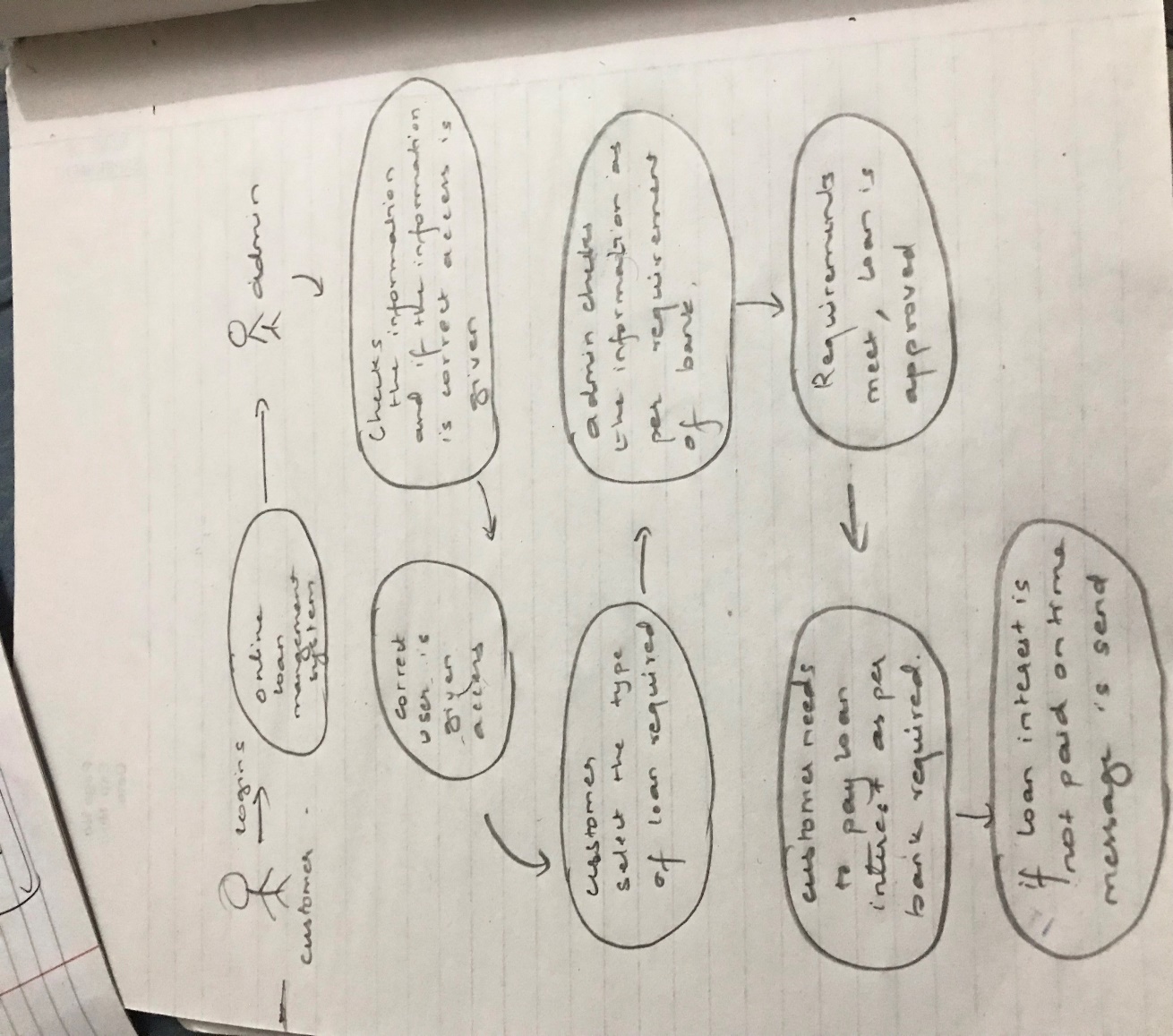
Analysis is done in order to get better understanding of the project. It helps to keep the biasness away from research conclusion with help of statistical data. It also helps to break information into components parts.

Since my system will be providing loan to people as per their need and requirement so for analysis methodology, I will be using soft approach as it focuses on people need’s and requirement.

There are certain steps that should followed while using soft approach:

1. Analyzing the existing information and producing rich pictures.
2. Defining root definition of significant parts of the information system.
3. Producing conceptual models.
4. Comparing the concept of the system with actual system.
5. Defining an implementing the system.
6. **Rich picture:**

Rich picture represents view of the whole system and can enable better planning and understanding of a system. It is basically drawn by hand and includes structures, processes, issues or developments. Here down below is the rich picture of my system to know how the system functions or works.



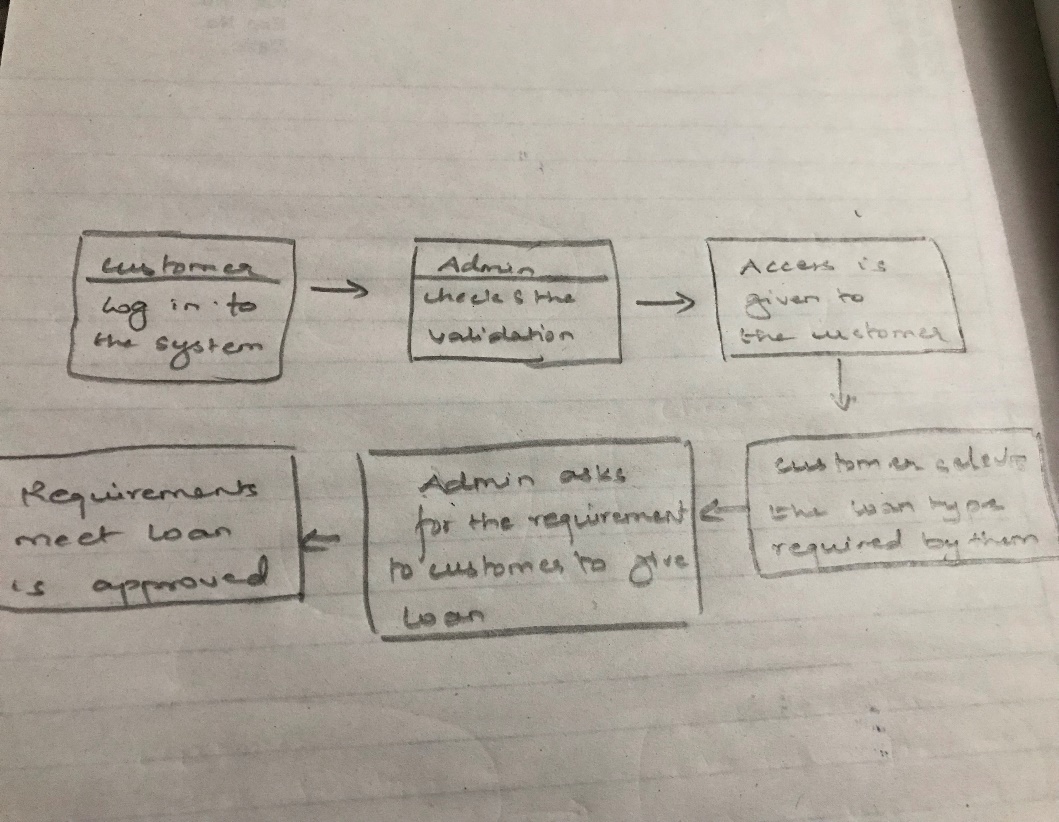
*Fig: Rich picture*

1. **Root definition:**

It helps to clarify the system processes and any problems. It is the short textual statements which describes the aims and functions of the potential system. As per my project admin registers the user information to the system and then only the user’s logins to the system and ask for loan. After all that the bank checks the user information’s and with all the necessary requirements loan approval is given to the user.

1. **Conceptual modals:**

The analyst uses the rich picture and root definition to construct a conceptual ‘ideal’ system that defines. This conceptual model can be used to describe how the system should function and what activities are necessary for the processes to take place.

****

*Fig: Conceptual modal*

1. **Comparing the concept of the system with actual system.**

Differences between the actual system and the model are noted and discussed with management. Required developments of the existing system are discussed. Necessary and feasible solutions are agreed and new system is implemented.

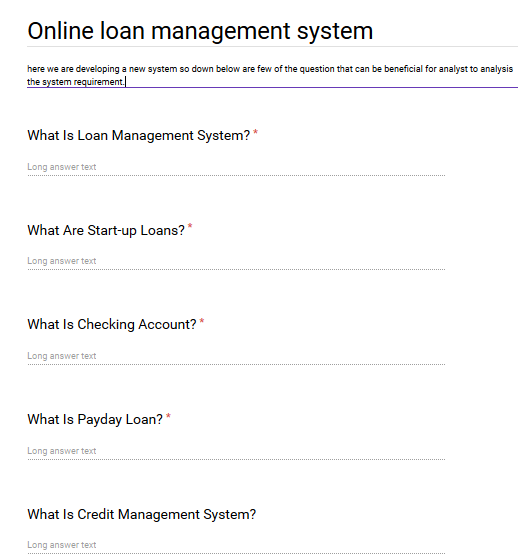
* **Information gathering method:**

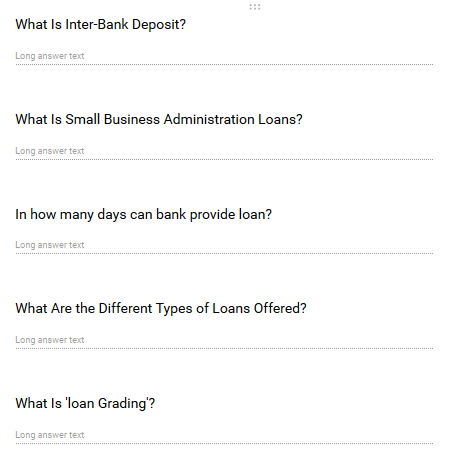
Information gathering method falls under the analysis. Here we gather the information for the development of the project and as per plan the system is built. Information can be collected from different sources like questionnaire, survey, interview e.tc. And here for my system I will be using questionnaire system for gathering the information.

1. **Questionnaire:**

Here in this method different question are prepared and ask to the group of people so that different views/opinions can be collected and system can be built according to it. The question should simple so that the person giving the answer doesn’t feel bore to answer. it should simple so that analyst can get more information.

Here down below are some of the question asked:





* **Feasibility study**: The Studies to determine the advantages or disadvantages, practicability, or capability of accomplishing a projected plan, study, or project is known as Feasibility Study. If the proposed system is not feasible then there is total waste of both time and money. That’s why I had to study the feasibility of the proposed system. The outcome of my studies is-
* Technical Feasibility.
* Economic Feasibility.
* Operational Feasibility.
* Legal Feasibility.

1. **Technical Feasibility:**

Technical feasibility is related to the hardware and software specification required by the system to function properly. For my system the basic requirements are that the system needs a minimum of 2 GB of ram to run all the features smooth and sudden. It needs a minimum 1.3 GHz processor to run smooth as less than that may create problems and computer must be connected to the internet.

1. **Economic Feasibility:**

Economic feasibility determines the cost effectiveness of a system. As my system is not so big and it does not require anything rather than a normal computer which must be connected to the internet, it can be cost effective. It does not require huge capital to build the system.

1. **Operational Feasibility:**

Operation Feasibility determines whether the proposed solution is desirable within the existing managerial and organizational framework. The Loan Management System is easy to use. The user does not need any costly training to operate this system. However, the new users must have computer operation knowledge especially on Windows platform. As is common for any new software at the beginning things may appear a little unfamiliar. But the system is gradually used it will be found to be quite easy to operate. And the user will become skilled in it automatically. It will take a few days to learn the operation of this new system.

1. **Legal Feasibility:**

It is related to the ethics. Before starting or implementing any system ethical things should be kept in mind. Here in case of my project that is loan management system there is no any conflicts or any legal conflicts that affects the ethics and harm the system.

* **System requirement specification**

1. **Functional requirement:**

The functional requirement of the system describes what a software should do. It specifies what function or the components of the system can be able to perform.

Following are the functional requirements of the system:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function ID** | **Title** | **Description** | **Purpose** | **Dependency** |
| F1 | Registration | New users get registered by providing their information | To get the features of the system. | F2 |
| F2 | Login | Registered user only can get the access of the system | To use the system in secure way. | F1 |
| F3 | Selecting loan | Users gets access to choose the loan they want. | To select loan. | F2 |
| F4 | Check loan | Admin checks the loan if they can approve or not | To manage the loan securely. | F3 |
| F5 | Delete | If the requirement does not meet, loan does not gets approve | To maintain the loan system | F4 |
| F6 | Send email | Admin sends email to the user if loan interest is not paid on time | To run smoothly the loan interest | F2 |
| F7 | Remove user | If the user is fake admin can remove that. | To manage the user of system | F2 |
| F8 | Edit | If any mistake occurs while inserting data, it can edit. | To down the mistakes | F2 |
| F9 | Update | To update user information if require. | To manage the user system | F2 |
| F10 | Logout | After completion user or admin can log out | To maintain security. | F1,F2 |

1. **Non-functional requirement of the system:**

The requirement that specifies the criteria that can be used to judge the operation of a system rather than specific behavior.it is also called quality attributes of the system.

Here down below are the non- functional requirement of the system:

|  |  |  |  |
| --- | --- | --- | --- |
| Nonfunctional requirement ID | title | description | Purpose |
| NFR1 | Usability | The system interface should be easy to use i.e. it should user friendly. | to make the system easy to use for the user. |
| NFR2 | Availability | The system should be able to access as per user required. | User can get access anywhere or at any time they want. |
| NFR3 | Reliability | The system should be reliable so that user can rely on it without any hesitation. | Make more user. |
| NFR4 | Security | System should not allow any vulnerabilities to enter as it may harm the user information | For privacy purpose. |
| NFR5 | Documentation | Explains what the system is capable to do. | To know the capability of the system. |
| NFR6 | Security | Validation of user email and password | To maintain data security. |
| NFR7 | Performance | The performance should be fast whithout altering user. | Insert, update, delete of user Should be performed fast. |
| NFR8 | Efficiency | Measures the speed or performance of the system. | Performance should be functional. |
| NFR9 | Response time | Time taken to perform the task | Response of the system towards the user should be quick |
| NFR10 | Quality | Functional requirement of the system should be fulfilled. | End product should be fully functional. |

* **Hardware and software specification:**

Hardware components and software components are necessary to run any system without this no system can run.

The basic hardware and software requirement to run the system is mentioned down below:

|  |  |
| --- | --- |
| **Hardware** | Software |
| Minimum of ***2 GB*** of ram to run all the features smooth and sudden. | Operating system: Windows 2007 to Windows 2010 |
| **1.*3 GHz*** processor | Firefox, internet explorer, chrome |
| ***128 GB*** of hard disk. | Database: php MySQL |

* **MoSCoW Prioritization**
* it is one of the best prioritization techniques for managing requirements.

Here the MoSCoW stands for;

M: Must have (without this project cannot run)

S: Should have (must haves over the long run)

C: Could have (low cost tweaking)

W: Won’t have (not necessary)

1. **Must have:** it represents the things that a system must have, without this a system cannot be developed properly. A release product becomes useless without work of an initiative and the initiative is most likely a “must have”.
2. **Should have:** the initiative is below must have. They play an important to develop a product but they are not vital. Even if the things in should have are not included in the system the system will work. However, if it is included, they provide significant value to the product.
3. **Could have:** The initiative is not necessary to develop the function of the system. There will be less impact in the system even if it is left out. It is nice to have could have initiative on the system. The initiative in could have are the first one to deprioritized.
4. **Won’t have:** Here the initiative is not necessary while releasing a product. Even if won’t have initiative are included in the system there won’t much impact in the system. But sometime it can be beneficial in the future.

Here down below is the MoSCoW Prioritization for my system:

**Functional requirement:**

|  |  |  |
| --- | --- | --- |
| ID | Functional requirement | MoSCoW |
| F1 | Registration | Must have |
| F2 | Login | Must-have |
| F3 | Selecting loan | Must have |
| F4 | Check loan | should have |
| F5 | Delete | Must have |
| F6 | Send email | Must have |
| F7 | Remove user | Should have |
| F8 | Edit | Could have |
| F9 | Logout | Must have |
| F10 | update | Should have |

**Non-functional requirement**

|  |  |  |
| --- | --- | --- |
| ID | Non-functional requirement | MoSCoW |
| NFR1 | Usability | Could have |
| NFR2 | Availability | should have |
| NFR3 | Reliability | Should have |
| NFR4 | Security | Should have |
| NRF5 | Documentation | Should have |
| NRF6 | Security | Must have |
| NRF7 | Efficiency | Must have |
| NRF8 | Response Time | Must have |
| NRF9 | Quality | Should have |
| NRF10 | Performance | Must have |

* **System architecture:**  Here the system architecture describes conceptual model which explains about the behavior, structure and view of the system in diagrammatic form. As to develop the system I have choose Laravel with PHP core, this architecture shows the use of PHP script. As my system is web based this architecture is best one to show how the system functions overall. This architecture shows through the use of browser PHP script is executed. Also, in my system all data are stored in database server and even the architecture also explains the same. Architecture is needed to see the structure of the system, implement design according to plan and to manage quality and requirement of the system.

HTML

Browser

Desktop

PHP script

Web server

HTTP SQL

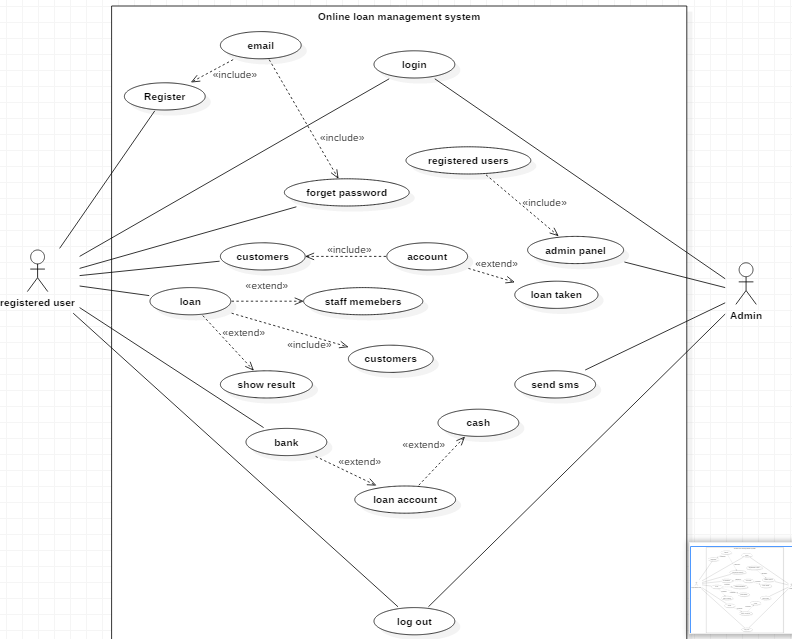
Database

Database server

HTML tables

*Fig: System architecture*

* **Use-case diagram:**



*Fig: Use-case diagram*

* **Name:** Online loan management system
* **Participating actor**: Registered user and Admin
* **Entry condition:**
* Customer must have an account in the bank database to get the access.
* Customer must have all requirements fulfilled in order to get loan.
* **Event flow**:
* User registers to the system using email and password.
* Admin verifies the user and give access.
* Admin can access his profile and as well as he can search users and can see the detail.
* Admin can only add user and give access.
* Admin can manipulate data as per system requires.
* Customer search for loan they need.
* Admin asks for the information as per bank requires and approves the loan if all the requirements matches.

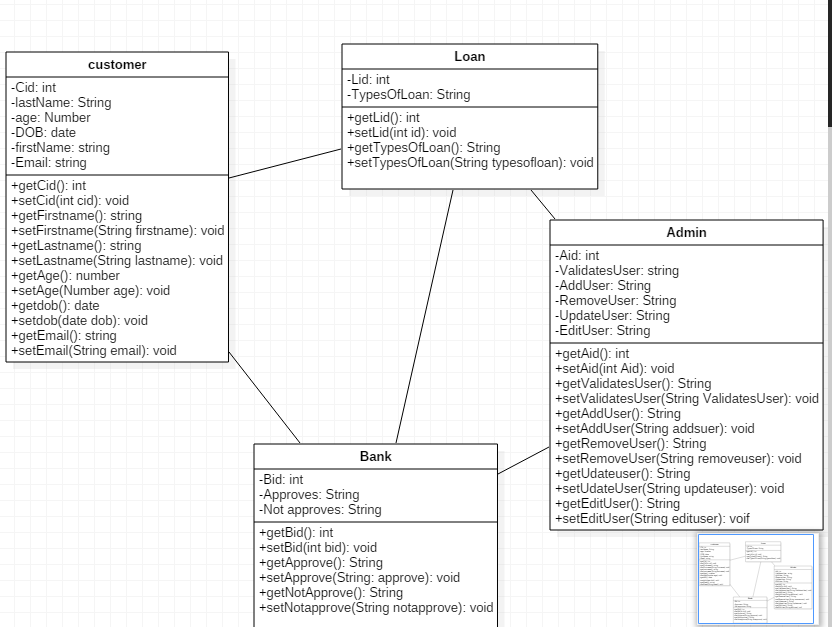
* **Exit condition**
* Loan is approved by the bank.
* **Exceptional case**
* If bank requirement is not fulfilled by the customer then loan can’t be approved.
* **Candidate class list and Diagrams:**

**Scenario:**  the system name is online loan management system. The main purpose of this system is to maintain the record of the customers who have taken loan from the bank. this system is designed in way that the data are kept confidentially. Customers must have an account in the bank in order to get loan. customer must register and then only they can get access. Admin can only add update delete the user information. New and unique id is given to customer whoever take the loan from bank. the customer logins in to the system and selects the type of loan want by them after that admin asks for the information and if only the information matches the requirements of the bank the customer can take loan. And if it does not match loan is not approved.

* **NLA/initial class diagram:**

NLA is basically done to find out the possible classes from the given paragraph by following NLA rule. Main goal of NLA is to find out the candidate class / actual class.

|  |  |  |
| --- | --- | --- |
| Noun | Verb | attribute |
| Customer | Add  Remove  Update  Edit | CId  Name  Email  Password  D.O.B  Address  Age |
| Admin | validates | Aid  Add user  Remove user |
| Bank | approve | Bid  Approves  Not approve |
| loan | Loan want | Lid  Types of loan |



*Fig: Class diagram*