Development of Loan Management System for Longbitz Software Solutions

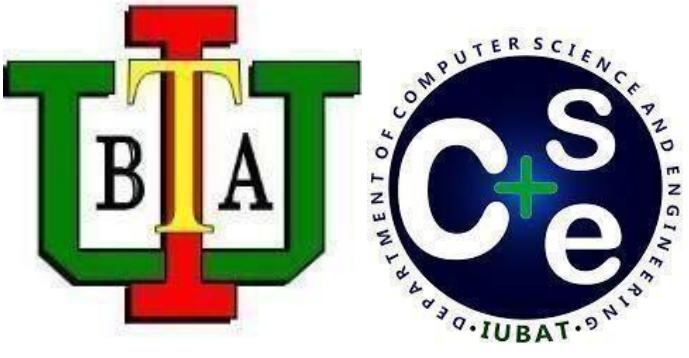
A Practicum Report Submitted By

Shumona Rashid Shikha

ID # 18303034

In Partial Fulfillment of the Requirements for the Award of

Bachelor of Computer Science and Engineering



Department of Computer Science and Engineering

College of Engineering and Technology

IUBAT – International University of Business Agriculture and Technology

Fall 2022

Development of Loan Management System for Longbitz Software Solutions

Shumona Rashid Shikha

ID: 18303034

A practicum report submitted in partial fulfillment of the requirements for the degree of Bachelor of Computer Science and Engineering (BCSE) The practicum has been examined and approved by

_____

Professor Dr.Utpal Kanti Das
Chair and Professor
Dept. of Computer Science and Engineering
IUBAT – International University of Business Agriculture and Technology

_____

Dr. Hasibur Rashid Chayon
Associate Professor and
Coordinator
Dept. of Computer Science and Engineering
IUBAT – International University of Business Agriculture and Technology

_____

Jabunnesa Jahan Sara
Lecturer, Supervisor
Dept. of Computer Science and Engineering
IUBAT – International University of Business Agriculture and Technology

Department of Computer Science and Engineering
College of Engineering and Technology
IUBAT – International University of Business Agriculture and Technology

Fall 2022

Letter of Transmittal

November 29, 2022

The Chairman, Practicum and Placement Board

College of Engineering and Technology - CEAT

IUBAT- International University of Business Agriculture and Technology

4, Embankment Drive Road, Sector 10

Uttara Model Town, Dhaka -1230, Bangladesh.

Subject: Letter of Transmittal.

Sir,

With due respect, I would like to approach you that it is a great opportunity as well as immense pleasure for me to submit this report titled "**Development of Loan Management System for Longbitz Software Solutions**" for the fulfillment of my Practicum course.

It was undoubtedly a splendid opportunity for me to work on this project to actualize my theoretical knowledge and has an enormous exposure with the corporate culture of a renowned company. Now I am looking forward for your kind appraisal regarding this practicum report.

I shall remain deeply grateful to you if you kindly go through this report and evaluate my performance. I hope that you would find the report comprehensive and competent augmented.

Thank you,

_____

Shumona Rashid Shikha

ID # 18303034

Program: BCSE

Letter of Authorization

December 01, 2022

IUBAT- International University of Business Agriculture and Technology

4, Embankment Drive Road, Sector 10

Uttara Model Town, Dhaka -1230, Bangladesh.

Subject: Letter of Authorization.

Shumona Rashid Shikha

ID # 18303034

Program: BCSE

Dear Shumona Rashid Shikha,

You will be happy to know that the project on "**Development of Loan Management System for Longbitz Software Solutions**". I have received in your proposal under my continue internship. Based on your proposal you will have to submit the project as soon as possible. I hope you will successfully complete it on time. After successful completion of the project, you are requested to write a report based on the project.

For any kind of assistance feel free to contact with me.

Co – Supervisor

_____

Dr. Hasibur Rashid Chayon
Coordinator and Associate Professor
Dept. of Computer Science and Engineering
IUBAT – International University of Business Agriculture and Technology

Supervisor

_____

Jabunnesa Jahan Sara
Lecturer, Supervisor
Dept. of Computer Science and Engineering
IUBAT – International University of Business Agriculture and Technology

Student's Declaration

I am Shumona Rashid Shikha , bearing ID# 18303034, student of BCSE - Bachelor of Computer Science and Engineering program, under the College of Engineering and Technology (CEAT) of IUBAT- International University of Business Agriculture and Technology declaring that, this report on the topic of "**Development of Loan Management System for Longbitz Software Solutions**" has been prepared for the fulfillment of the internship CSC 490, Practicum as well as the partial requirement of BCSE-Bachelor of Computer Science and Engineering degree.

The report and the project on **—** "**Development of Loan Management System for Longbitz Software Solutions**" are originally prepared by me. All module and working procedure of this project is being made after proper inspection and internet information.

It has not been prepared for any other purposes, rewards or presentations.

_____

Shumona Rashid Shikha

ID # 18303034

Program: BCSE

Acknowledgements

First and foremost, I'd like to thank Almighty Allah for assisting me in properly completing the report.

My sincere thanks to Prof. Dr. Abdur Rab, Honorable Vice Chancellor, IUBAT-International University of Business Agriculture and Technology to give me an opportunity to submit this practicum report.

My profound respect goes to Professor Dr.Utpal Kanti Das, Chair and Professor, Department of Computer Science and Engineering, IUBAT- International University of Business Agriculture and Technology for approving me to work on the project.

My outmost and sincere gratitude goes to Associate Prof. Dr.Hasibur Rashid Chayon, Coordinator of Department of Computer Science and Engineering, IUBAT- International University of Business Agriculture and Technology for allowing me to complete the project.

I would like to express our gratefulness to my supervisor Jabunnesa Jahan Sara, Lecturer, Department of Computer Science and Engineering, IUBAT- International University of Business Agriculture and Technology who has given me the opportunity to make such a project report for not only in this semester but also throughout my entire education life at IUBAT- International University of Business Agriculture and Technology by giving her valuable suggestions and advices at any time, at any situation. I would able to make this report effectively and properly only for her right direction.

I owe my deepest gratitude to my parents and our family members whose inexhaustible love was indispensable to endure in the tragic moments that confronted me once again with the vulnerability of life.

Lastly, this report would not have been possible without the essential and gracious support of many individuals who encouraged and supported us in any respect to complete this thesis on time. I am also so grateful to them.

Supervisor's Certification

This is to certify that Practicum report on "**Development of Loan Management System for Longbitz Software Solutions**" has been carried out by Shumona Rashid Shikha bearing ID# 18303034 of IUBAT – International University of Business Agriculture and Technology as a partial fulfillment of the requirement of practicum defense course. The report has been fully completed under my guidance and is a record of the accomplished work carried out successfully. To the best of my knowledge and as per his declaration, no parts of this report has been submitted anywhere for any degree, diploma or certification.

Now she is permitted to submit the report. I wish her success in all her future endeavors.

Practicum Supervisor,

_____

Jabunnesa Jahan Sara

Lecturer, Supervisor

Department of Computer Science and Engineering

IUBAT- International University of Business Agriculture and Technology

Departmental Certification

On behalf of the Department of Computer Science and Engineering of International University of Business Agriculture and Technology (IUBAT) we, the undersigned, certify that this practicum report "**Development of Loan Management System for Longbitz Software Solutions**" for the award of Bachelor of Computer Science and Engineering (BCSE) degree was duly presented by Shumona Rashid Shikha (ID No. 18303034) and accepted by the department.

_____

Jabunnesa Jahan Sara

Lecturer, Supervisor

Department of Computer Science and Engineering

IUBAT – International University of Business Agriculture and Technology

_____

Dr. Hasibur Rashid Chayon

Coordinator and Associate Professor

Department of Computer Science and Engineering

IUBAT – International University of Business Agriculture and Technology

_____

Professor Dr.Utpal Kanti Das

Chair and Professor

Department of Computer Science and Engineering

IUBAT – International University of Business Agriculture and Technology

Abbreviations

AFP-Adjusted Function Point

DET-Data Element Type

DFD-Data Flow Diagram

DT-Data Transition

EI-External Input

EIF-External Interfaces File

EO-External Output

EQ-External Query

ER-Entity Relationship

FP-Function Point

FTR-File Type Referenced

TDI-Total Degree of Influence

UFP-Unadjusted Function Point

GSC-General System Characteristics

ILF-Internal Logical File

QA-Quality Assurance

RET-Record Element Type

SDLC-Software Development Life Cycle

TQM - Total quality management

Abstract

The Loan management system is essential and helps to make sure success or failure of any deposit institution. Mortgage loan troubles have constantly been a key note on the chance of loan loss. This system is a platform to provide good communication and communication between the Loan Seeker and the Banks. Loan management systems help automate the entire loan application lifecycle. Depending on requirements, the web application can assist in part. The software can help process loan seekers' information, Loan application, search or filter and manage their own applications, and more. They can also provide lenders with accurate statements, reports and other supporting documents. Moreover, Bank can manage interest rates, duration, skim, loan seekers, loan applications. From this system the loan seeker will be able to apply for the loan according to their preferable criteria. This software is used by the administrator to manage loan seeker, banks and to set loan types, manage applications. This system provides user-friendly interface as well. This system is created by Laravel 9x structure where Hypertext Markup Language, Cascading Style Sheet(CSS), Structure Query Language used.

Table of Contents

Practicum Report Title ..... **ii**
Letter of Transmittal ..... **iii**
Letter of Authorization ..... **iv**
Student's Declaration ..... **v**
Acknowledgement ..... **vi**
Supervisor's Declaration ..... **vii**
Department Certification ..... **viii**
Abbreviation ..... **ix**
Abstract ..... **x**
Table of Contents ..... **xi**
List of Figures ..... **xv**
List of Tables ..... **xvii**
Internship Certification ..... **xviii**
Chapter 1: Organization Overview **1**
1.1 Organizational Overview 2
1.2 Organization Services 2
1.3 Organization Vision 2
1.4 Organization Mission 2
1.5 My position in this Organization 2
1.6 Organizational Structure 3
Chapter 2: Project Introduction **4**
2.1 Introduction 5
2.2 Background of study 5
2.3 Objective 5
2.3.1 Broad Objective 6
2.3.2 Specific Objective 6
2.4 Proposed System Benefits 6
2.5 Methodology 7
2.5.1 Data Sources 7
2.6 Limitation of the Project 7
2.7 Process Model 7
2.7.1 Reason for choosing Incremental Process Model 8
2.8 Feasibility Study 8

2.8.1 Technical Feasibility 9
2.8.2 Economic Feasibility 9
2.8.3 Operational Feasibility 10
Chapter 3: Requirement Engineering 11
3.1 Requirement Analysis 12
3.2 Requirement Engineering 12
3.2.1 User Requirement 13
3.2.2 System Requirement 14
3.2.3 Functional Requirement 16
3.2.4 Non-Functional Requirement 17
3.2.5 Hardware Requirement 18
3.2.6 Software Requirement 18
3.2.7 Use Case Symbol 18
3.2.8 Use Case Diagram 20
Chapter 4: System Planning 21
4.1 Scope of Project 22
4.2 Function of Proposed System 22
4.2.1 Function Description 22
4.3 System Project Planning 22
4.4 System Project Estimation 23
4.5 Function Oriented Metrics 24
4.6 Identifying Complexity 25
4.7 Identifying Complexity of transition function 26
4.8 Identifying Complexity of data function 28
4.9 Unadjusted function point contribution 29
4.10 Performance and Environmental impact 30
4.11 Counting Function Point 30
4.12 Process Based estimation 32
4.12.1 Effort Distribution 33
4.12.2 Project schedule chart 34
4.13 Cost Estimation 34
4.13.1 Personal Cost 34

4.13.2 Hardware Coat 35
4.13.3 Software Cost 36
4.13.4 Other Cost 36
4.14 Account Table 36
Chapter 5: Risk Engineering 37
5.1 Risk Management 38
5.2 Stages of Risk 38
5.3 Categories of Risk 39
5.4 The RMMM plan 40
5.5 Project Risks 41

Chapter 6: Analysis Modeling 44
6.1 Software Analysis Pattern 45
6.2 Activity Diagram 45
6.3 Swim lane Diagram 49
Chapter 7: Designing 50
7.1 Design of the system 51
7.2 Data Flow Diagram 56
7.3 Database Design 57
7.4 Entity relationship Model 58
7.5 Database Table Structure 59
Chapter 8: Quality Assurance and Testing 77
8.1 System Quality Management 61
8.1.1 Software Quality management Process 62
8.1.2 Quality Assurance Matrix 62
8.2 System Testing 62
8.2.1 Black-box Testing 62
8.2.2 White-box Testing 63
8.3 System Testing Design 63

Chapter 9: Conclusion 65
9.1 Preface 66

9.2 Practicum and Its Value 66
9.3 Future Plan 66
9.4 Conclusion 67

References 68

List of Figures

Figure 1.1: Organization Structure..... 3
Figure 2.1: Incremental Process Model 8
Figure 3.1: Use Case Symbols.....19
Figure 3.2: Use Case Diagram for Developing of Loan Management System 20
Figure 4.1: Effort Based Estimation.....33
Figure 4.2: Effort Distribution PIE Chart.....33
Figure 4.3: Project Schedule Chart 34
Figure 6.1: Activity Diagram for Admin.....46
Figure 6.2: Activity diagram for Bank 47
Figure 6.3: Activity diagram for Loan Seeker 48
Figure 6.4: Swim lane Diagram for Admin, Bank and Loan Seeker.....49
Figure 7.1: Website Design page.....51
Figure 7.2: Registration interface page for user 51
Figure 7.3: Login interface for User 52
Figure 7.4: Search Loan Type interface.....52
Figure 7.5: Admin login page.....53
Figure 7.6: Interface Design of Admin Dashboard.....53
Figure 7.7: Interface Design of Loan Types Page 54
Figure 7.8: Interface Design of Bank List Page 54
Figure 7.9: Data Flow Diagram for Context Level (level 0) 55
Figure 7.10: Data Flow Diagram for Level 1 56
Figure 7.11: Entity Relationship Diagram 58
Figure 7.12: All Database Table Structure.....59
Figure 8.1: Black box and white box testing source.....63

List of Tables

Table 4.1: Complexity Matrix 24
Table 4.2: Complexity Matrix 2.....24
Table 4.3: Complexity Matrix for UFP 24
Table 4.4: Complexity Matrix 3.....25
Table 4.5: Complexity Matrix for UFP 2 25
Table 4.6: Identifying complexity of transition function 26
Table 4.7: Identifying complexity of data function.....28
Table 4.8: Unadjusted function point contribution of transition function 29
Table 4.9: Unadjusted function point contribution for data functions 29
Table 4.10: Performance and environmental impact 30
Table 4.11: Processed based Estimation 32
Table 4.12: Personnel Cost.....35
Table 4.13: Hardware Cost.....35
Table 4.15: Software Cost.....36
Table 4.16: Estimation of Other Cost.....36
Table 4.17: Accounts Table.....36
Table 5.1: Project Risk (P01).....41
Table 5.2: Project Risk (PR02).....41
Table 5.3: Technical Risk (TR01) 41
Table 5.4: Technical Risk (TR02) 42
Table 5.5: Business Risk (BR01).....42
Table 5.6: Business Risk (BR02).....43
Table:8.1: Test case (1).....63
Table:8.2: Test case (2).....64
Table:8.3: Test case (3).....64
Table:8.4: Test case (4).....64

Internship Certification

Chapter 1: Organization Overview

The very first chapter presents a summary of the company's activities, goals, strategy, and various services. This section goes through a high level of organizational description, the organizational hierarchy, and my responsibilities as an intern here at all.

1.1 Organizational Overview

Longbitz Software Solutions: Longbitz a software development company provides well equipped software development services to the customers. A band of technical experts in helping the potential clients through web and mobile application development. They provide services as a distant digital partner physically, but closer to solve customer problems. Most intimately and with care. They help companies at any stage of the product software development cycle: from R$D and building MVP from scratch, UX analyzing and imporoving.

1.2 Organization Services

- Mobile application Development.
- Web Design and Development
- Dedicated Development Team
- Q/A Testing

1.3 Organization Vision

To expand our digital services beyond the horizon with the power of a growing potential workforce, acute knowledge and supportive business offerings that help the client operator their business faster, better and more update.

1.4 Organization Mission

Their mission is to align digital experiences with brand experience for ensuring maximum benefits to the clients. We blend user experience, design, strategy and execution to form product in such a way that serves the requirements of our customers and the end users satisfaction.

1.5 My position in this Organization

This company involved me as an intern developer. A supervisor guides me in this organization. He's exceedingly helpful and well-informed. He has learned a lot from me. My project was completed successfully and on time. This was only possible with the help of the supervision of my supervisor. Managing the normal hours was also a wonderful experience for me. I also respect the organization's other principles. I'm delighted to be employed in this environment. It has benefited me in preparing for the beginning of my professional life.

1.6 Organizational Structure

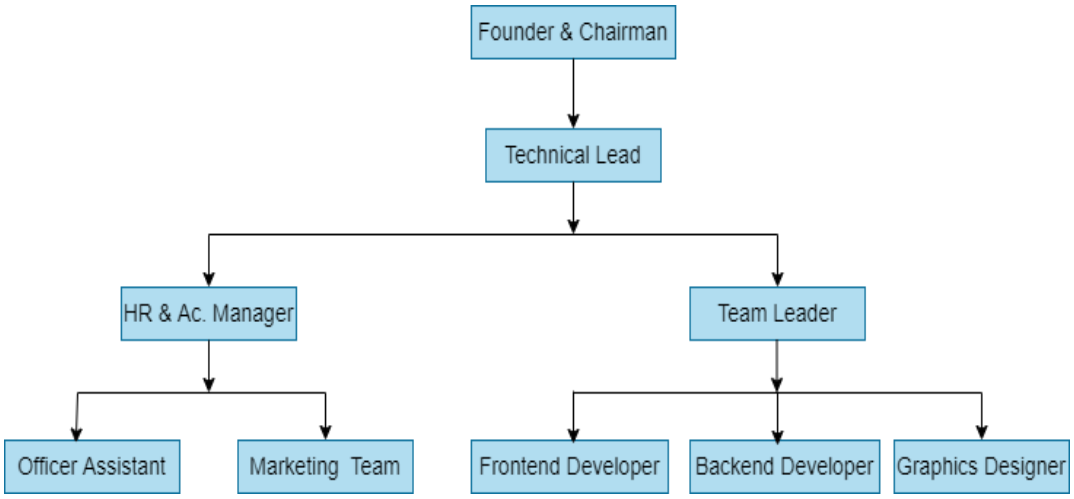


Fig-1.1: Organizational Structure

Chapter 2: Project Introduction

2.1 Introduction

A loan management system is a digital platform that helps automate every stage of the mortgage lifecycle, from software to closing. The normal mortgage administration process is meticulous, time-consuming, and requires amassing and verifying information about applicants, their trustworthiness, and their credibility. Now a days people are facing many problems in taking Loans. The proposed system will solve all problems. In this system, Every Bank will register their loan. So that when the loan seeker will come to our system or visit our system at a time he will be able to see all the Loan from a single page. The loan seeker will be able to prefer his criteria and we will be able to prefer his criteria and we will suggest the Bank Loan according to his criteria. From the Bank list the Loan seeker can apply for the loan. After the bank will be able to check the progress report of their loan. The loan seeker doesn't have to go to the Bank until the Bank accept the paper.

2.2 Background of study

I explored numerous Loan Management Systems Loan management systems help automate the entire loan lifecycle. Depending on requirements, this system can assist in part or whole. The software can help with processing customer information, create new loans, and more. They can also provide lenders with accurate statements and reports. A good loan management system maximizes automation and tools in order to reduce the risk of such errors. The Loan Management feature will help you to automate your customer's entire loan lifecycle**.** Based on the requirements, the software programs will assist you throughout the process. Loan seeker will be provided with accurate loan statements and reports that will keep you updated with present and future installments. The system also manages interest rates and provides various tools for collection automation.

2.3 Objective

From the Loan seeker's point of view

- Justification
- The simple and easiest way to apply for a loan.
- Manage Applications from loan seeker's profile.
- To help people to find their desired loan
- To help hassle-free loan application
- Saving plenty of time.

From the Bank's point of view

- Ensure a completely paperless process.
- Easy to reach customer/loan seekers.
- Manage loans and get to know customer interest.
- Easily manage applications and documents
- Reduces manpower
- Reduce cost as manpower is reduced

2.3.1 Broad Objective

The Loan Management System plays a major role. It's purpose to provide loan to the loan seeker with the information they need to make a comprehensive assessment of whether the loan can be deemed suitable for them, and thus meet the obligations of responsible lending. For this project in data collection phase I collected primary and secondary data. Longbitz Software Solutions provided with all type of primary and secondary data needed to develop the system. The procedures and processes that I followed to develop this system are clearly described in the analysis and design chapter

2.3.2 Specific Objective

Such challenges can be mitigated by an automated computerized solution with a web-based framework.

- The above system captures information pertaining to several Banks and maintains it in a centralized database, culminating in ease of accessibility and continuity. Form a bond between the bank and the loan seeker.
- Enables system characters to successfully collaborate.
- Simplifies the process of creating data.
- To improve quality of work and accuracy.
- To make the system trustable and secure and to make the system user friendly.
- Get instant and detail information at single terminal.

2.4 Proposed System Benefits

A scope statement is a comprehensive representation of the tasks necessary to effectively accomplish or conduct a project. Some benefits of this system are-

- Requires real-time information.
- At a time the loan seeker will be able to see all the loan from a single page.
- The loan seeker will be able to prefer his criteria and the system will suggest the Bank loan according to his criteria.
- Save User's valuable time.
- Information is kept secure.

2.5 Methodology

The development process on "Loan Management System" is completed by following the structure described later on Software Analysis & Design. The study of this project is tentative in nature. It aims to development of a system which makes the service seeking and providing process of admin and users easier. The variables identified to manipulate through a handy inspection and from primary and secondary data.

2.5.1 Data Sources

The foregoing are the information sources for this construction process:

- Primary Data
- Secondary Data

Primary data are collected from the organization which gives me a clear idea how the service seeking process is carried on now-a-days. The organizations experience, face to face interview with the users and web administrators help me to generate primary data.
Secondary data is collected through the real-life experience, studying different articles, existing system. Data and facts collected from different web sites and sources made us understand the project better.

2.6 Limitation of the Project

The limitation of the system is since it will be web based,

- internet access is a bit expensive, and not all people from our country can afford it.
- Lastly, most people are not computer literate so it might be a challenge for them as well.

2.7 Process Model

There are plenty of circumstances when the baseline requirements documentation seems fairly

unambiguous, but somehow the ultimate complexity of the construction activity precludes an entirely chronological strategy. Furthermore, it's possible that there's a compelling need for us to serve consumers with something like a narrow set of application functionality promptly, and afterwards optimize and enlarge upon the features in successive firmware upgrades. The incremental improvements model is a software development paradigm in which the model is specified, implemented, and evaluated the competitive (a tiny bit more each time) till the referendum result is accomplished. It necessitates both maintenance and enhancement. Whenever the prototype has complied with all of its guidelines, it is referred to as being accomplished. Such a paradigm incorporates elements of the waterfall methodology with the core principle of original association.

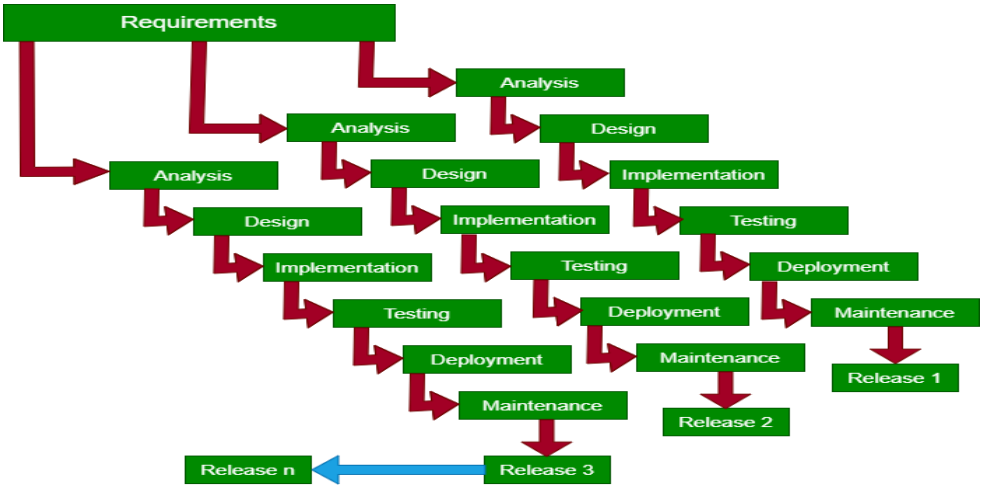


Figure-2.1: Incremental Process Model

2.7.1 Reason for choosing Incremental Process Model

- Improved workflow help is available.
- It reduces rework in software development.
- It saves time since it eliminates rework in coding.
- Reduction in the risk of implementation failure.
- Setting distribution objectives more conveniently.
- This model allows users more flexibility.
- Each phase has specific deliverables and a review process.

2.8 Feasibility Study

The goal of a feasibility study is to establish and understand the feasibility of a project.

The major goal of a feasibility analysis is to determine the project's risks and possibilities, as well as its chances of success. A feasibility study determines whether or not the proposed structure is feasible or feasible for the organization. This indicates whether or not the projects we will perform are worthwhile. There are three main areas for examining and generating ideas on another framework. For something like the formation of revolutionary schemes, there must be 3 main focus areas to consider. Three significant relevant factors are acknowledged even as exploring this same service's survivability to establish for certain if it is sustainable if it is brought back.

- Technical feasibility
- Economic feasibility
- Operational feasibility

2.8.1 Technical Feasibility

The system must first be assessed from a technical point of view. An outline design of the system requirement in terms of input, output, programs, and procedures must be used to determine its feasibility. Technical feasibility addresses concern about hardware actual competence, dependability and convenience and the knowledges of the development team.

Technical issues raised during the investigation are:

- Is it possible to develop the proposed system using the current technical resource?
- If not, can current technical resources be upgraded or added to in a manner that fulfills the request under consideration?
- Is there technology in existence that meets the specifications?

The project is developed such a way so that the necessary functions and performance are achieved within the constraint.

2.8.2 Economic Feasibility

This evaluation usually includes a cost-benefit analysis of the project, which helps organizations in determining the project's viability, time, cost, and benefits before allocating financial resources. It also functions as an independent project assessment and

improves project credibility by assisting decision-makers in determining the proposed project's positive economic benefits to the organization.

The cost and utility of the emerging system must be justified. Criteria to ensuring that work is focused on the project that will yield the best results and returns as soon as possible. The cost of developing a new system is one of the elements that influences its development.

The following are some of the important financial questions asked during preliminary investigation:

- The costs investigate the entire system.
- The financial benefits, such as lower expenditures or less costly errors.
- The hardware and software expenses.

Since the suggested system was constructed as part of my practicum defense, so there are no manual costs associated with it. Furthermore, all of the resources required to develop the system are already in place, and indicating that the system is economically viable for development

2.8.3 Operational Feasibility

Operational feasibility determines if the human resources are available to operate the system once it has been installed. Users that do not want a new system may prevent it from becoming operationally feasible.
If users are virtually wed to the present system, see no problems with it, and generally not involve in requesting a new system, resistance to implementing the new system will be strong. The new system has a low chance to become operational.

This includes the following questions:

- Is there sufficient support for the users?
- Will the proposed system able to fulfill the user needs?

The project would be beneficial because it satisfies the objectives when it developed and installed. All behavioral aspects are considered carefully and conclude that the project is behaviorally and operationally feasible.

Chapter 3: Requirement Engineering

A method for determining the infrastructure that a system may need to provide to its customers, as well as the constraints under which it operates and is created, the need reflects the user's need for a mechanism that completes a specified activity, including controlling equipment.

3.1 Requirement Analysis

Requirements analysis is a software engineering task that links system engineering and system design together. Requirements analysis enables a software engineer to define software allocation and construct models of the data, functional, and behavioral domains that software will meet. Requirement analysis gives a representation of the system's information, function, and behavior to the software designer.

The first stage of the software development process is requirements analysis. It encompasses those tasks that go into determining the needs or conditions to meet for a new or altered product, taking account of the possibly conflicting requirements of the various stakeholders, such as beneficiaries or users.

There are some phases of requirement analysis which is described below:

- Requirement Initiation
- Requirement Elicitation
- Requirement Elaboration
- Requirement Negotiation Requirement Specification
- Requirement Verification

3.2 Requirement Engineering

This same work by detecting and yielding client requirements but instead explicitly stating enterprise applications is renowned as scope management. There are various interpretations of requirements elicitation; however, people would mostly rely on assertions about which criteria necessitate establishing that which people require from a computing device, rather than attempting to understand how much their expectations translate to in terms of design. The system development life cycle, which also fixates mostly on procedures for creating this

same mechanism, something which the current users crave, is tightly intertwined with requirements elicitation.

- User requirements
- System requirements
- Functional requirements
- Non-functional requirements
- Hardware Requirements
- Software Requirements

3.2.1 User Requirements

Administrator:

- Administrator may log in and create an account.
- The administrator can manage their profile.
- Administrator has access to the Banks dashboard.
- Administrator can view their own activities.
- Administrator may handle loan appliers applications.

Bank:

- Bank can register and login.
- Bank can register their loan.
- Bank can provide their own banks information.
- Banks can change their own password and email address.
- Banks can manage loans and get to know loan seekers interest.

Loan Seeker:

- Loan Seeker may register and login.
- Loan Seeker can manage his profile.
- Loan Seeker can seek the loan.
- Loan Seeker can prefer criteria for loan.
- Loan Seekers can apply for the loan.

3.2.2 System Requirements

Administrator:

- ✓ Administrator may log in and create an account.
    - To login, the administrator must click the " login" button in the admin login bar below. A login form will then be displayed.
    - The administrator must complete the form and click the "Log in" button.
    - If the admin's information matches what is in the database, administrators have access to their panels. If not, the system will notify you with a warning message and urge you to enroll as a new user.
- ✓ The administrator can manage their profile.
    - Through use of the administration panel, if an administrator clicks on their User name from the header they might see the personal system profile.
    - When you click the user name the system displays the entire information table.
    - An "Update Profile" button will be available. An admin can modify his/her profile details using this button.
    - An "Update password" button will be available. The admin can change the password using this button.
- ✓ Administrator has access to the banks dashboard.
    - When viewing the banks dashboard, administrators can see the total number of banks, total posts, total pending, how many loans posts occur in a day, and total accepted posts.
    - After that, the administrator can access and amend a student's profile.
- ✓ Administrator can view their own activities.
    - By selecting the admin profile button, admins may view their own activities, such as how much work they have done in a week or month.
- ✓ Administrator may handle loan appliers application.
    - Admin can view loan appliers application information and decide whether or not the loan seeker is suitable for this loan.

Bank:

- ✓ Bank can register and login.
    - To begin, there is a "Registration" menu bar banks must provide accurate information in order for their profiles to be registered.
    - The banks must then login to his profile using the correct information provided during registration.
- ✓ Bank can update his profile.
    - After logging in, the bank can update his personal profile by clicking the "about" option.
    - The bank must supply their own authentic profile photo.
    - The bank must include their address and institute.
    - After that, the bank may go to the "settings" tab and input his new password and email address to login to his account.
- ✓ Bank can supply their own information.
    - Here banks can provide banks information, name, logo, email, loan criteria, loan type.
    - Then loan seeker can select the expectation loan from bank.
- ✓ Bank can change his own password and email address.
    - By selecting the form the bank can change old password and the set new password
    - Bank can then input a new email address for login.
    - The Bank must then click the "Update" button.
- ✓ Banks can manage loans and get to know loan seekers interest
    - The Bank can manage loan, loan types and criteria according to the loan type.
    - The loan seeker may then view the banks profile and take loan from bank.

Loan Seeker:

- ✓ Loan seeker may register and login.
    - At first, there is a "Registration" menu bar, where loan seeker must provide accurate information in order to register their profile.
    - After that, the loan seeker must login to his profile using the correct information provided at registration.
- ✓ Loan Seeker can manage his profile.
    - After logging in, Loan seeker can update their personal information.
    - Loan seeker must input their actual address or location.
    - After that, the loan seeker may click on the "login" button and input his new password and email address to login to his account.
- ✓ Loan seeker can seek the loan.
    - Loan seeker may search for loan by selecting their criteria, loan types.
- ✓ Loan Seeker can prefer for loans according to their Criteria.
    - Loan Seeker can prefer loan on the website by clicking the search button.
    - The loan seeker must provide all updated information for the website post.
    - If any information is incorrect, the loan seeker can correct it.

3.2.3 Functional Requirements

Admin:

- ✓ Registration
- ✓ Login
- ✓ View own profile
- ✓ Manage system
- ✓ View Banks profile
- ✓ View Loan seekers profile
- ✓ Delete Banks profile
- ✓ Delete Loan Seekers profile and Logout

Banks:

- ✓ Registration
- ✓ Login
- ✓ View profile
- ✓ Update profile
- ✓ View Loan Seekers Post
- ✓ Set Criteria
- ✓ See Loan Appliers
- ✓ Logout

Loan Seekers:

- ✓ Registration
- ✓ Login
- ✓ View profile
- ✓ Update profile
- ✓ Set Criteria in the website
- ✓ See the Update Information.
- ✓ See the banks information
- ✓ Logout

3.2.4 Non-Functional Requirements

- ✓ The minimum length of password should be Eight
- ✓ User cannot make more than one account using one email.
- ✓ Registered user has to retype email and match the password in login.
- ✓ User name & password for admin and user saved in database for security purpose.
- ✓ Password will be encrypted.

3.2.5 Hardware Requirements

This same mentioned console isn't really a necessary amount for implementing automation, but more a staging ground for maintaining the system flawlessly and courteously. Then it also is taken into consideration the doable portion of pedestrians which might transmit through all the console.

- Intel(R) Core(TM) i3-4030U CPU
- GB (DDR3) RAM
- 128 GB SSD.
- Web Server: Xampp Server Bitnami 8.0.10
- Data set Engine: MySQL

3.2.6 Software Requirements

- Windows 10
- Xampp Server
- Visual Studio Code
- Web Browser (Google Chrome)

3.2.7 Use case symbols

➢ Vital particles in the use case:

1. **Artist**: An action star that's outside or situational factors into the framework.

2. **Use case**: The use case seems to be a characterization of a component of the system or implementation. It'll be pulled back inside the morphology of such a racecourse and categorized for its purpose.

3. **Framework Demarcation Line**: A framework is a rectangular prism chain of events that occur when an interacts with it. This seems to be an alternative measure that seems to be advantageous for visualizing tremendous frameworks.

4. **Correlation**: An association seems to be an interaction, one use case and now a protagonist.

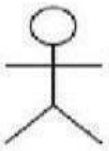
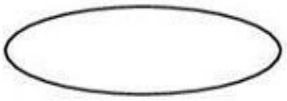
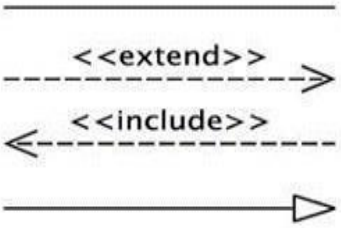
Symbol	Reference Name
	Actor
	Use case
	Relationship

Figure 3.1: Use Case Symbols

3.2.8 Use case diagram:

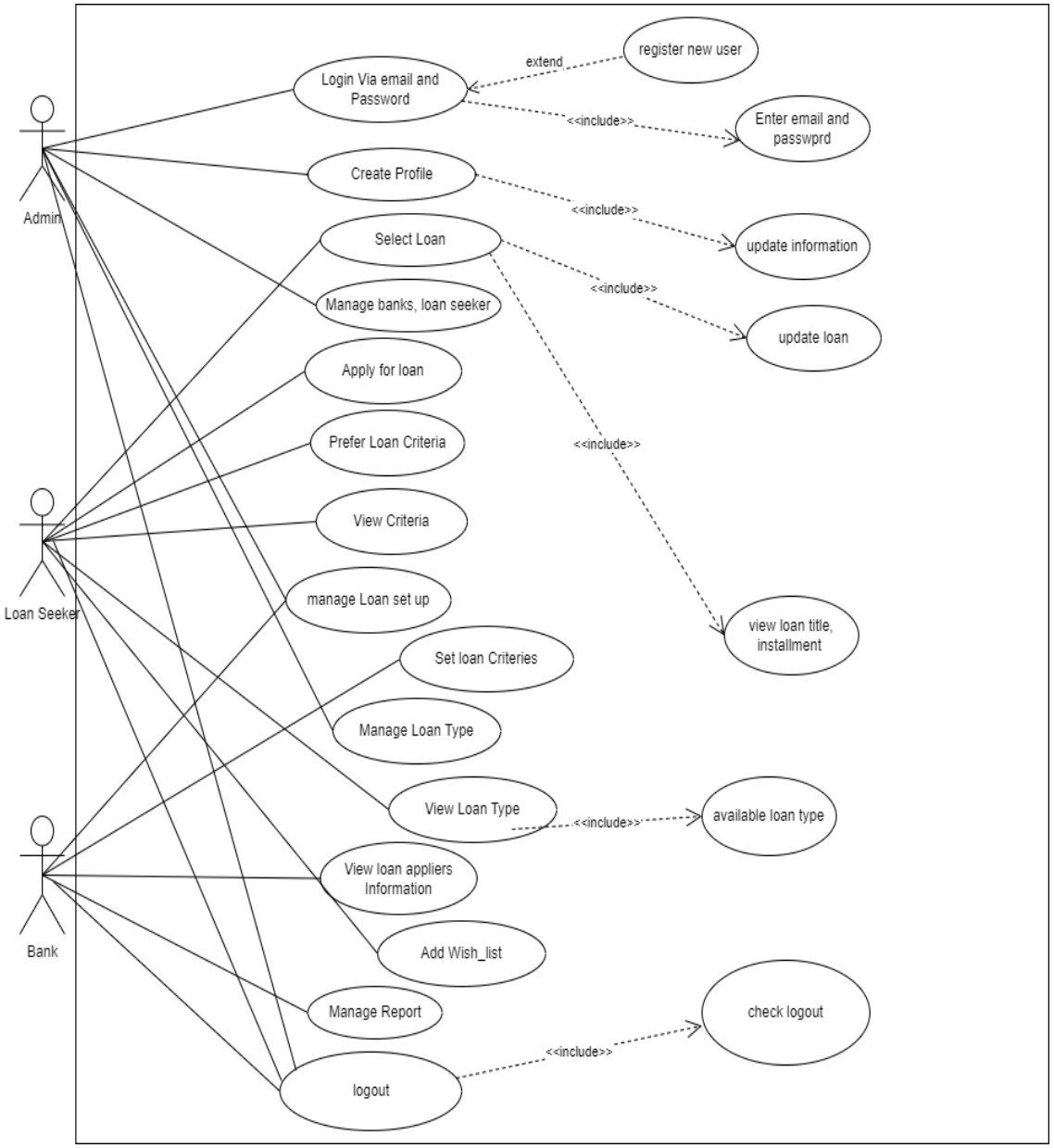


Figure 3.2: Use Case Diagram for Developing of Loan Management System

Chapter 4: System Planning

This same chapter on planning processes exemplifies the components of a framework for the "Development of Loan Management System for “Longbitz Software Solutions". One such section also contains critical path evaluation, initiative dissemination, and just an overall framework benchmark.

4.1 Scope of Project

A project scope is a detailed description of the work that must be done in order to properly finish or deliver a project. The first and most critical activity in software project planning is determining the scope of the software. The software scope statement must be binding. The software should be able to manage the database as well as the members, operators, and package information. The power of reusability has been granted to this project. The scope of the project is to fulfill our clients demand. The system is simple to use even for non-technical people.

4.2 Function of Proposed System

4.2.1 Function Description

Create an Account	F1
Login to the System	F2
Insert Bank	F3
Go to Bank	F4
Update Bank Info	F5
Insert Loan Seeker	F6
Go to Loan Seeker	F7
Prefer Loan Criteria	F8
Insert Administrator	F9
Go to Administrator	F10
Update Administrator Info	F11
View Loan Appliers Information	F12
Set Loan Criteria	F13

4.3 System Project Planning

Once initiating any endeavor, it is crucial to estimate the research to be carried, the same infrastructure that is going to be considered necessary, the length of time that will move from concept to execution, and thus to scrutinize the endeavor to discern its practicability. The

reference implementation organization starts with a sequence of words and deeds referred to as the application program management. I approximate the research to be carried, the materials that will also be considered necessary, and also the length of time it takes from ideation to execution, and subsequently I scrutinize the endeavor to verify whether it would be sustainable. These initiatives of this agency's technology planning phase that already have come afterwards are as regards:

- ✓ System Project Estimation
- ✓ Function Point Estimation
- ✓ Based Estimation
- ✓ Effort Distribution
- ✓ Task Scheduling Project
- ✓ Schedule Chart Cost
- ✓ Estimation

4.4 System Project Estimation

A reference implementation estimate's precision is contingent on a variety of considerations, which would include:

- ✓ A precisely approximated demand for the product to be constructed.
- ✓ This is the magnitude at which the planning phase demonstrates the architectural squad's or computing squad's competence.
- ✓ The sustainability of the quality attributes and the framework for the process of software development.

The much more striking feature of such a piece of code, which I will now confront, is the source code diameter assumption. The much more significant characteristic of such a piece of code that I'll need to confront is the source code size guesstimate. Whereas if source code dimension isn't really linearly proportional, it'll still aggravate a myriad of subjects, such as time constraints, infrastructure issues, and so forth. Prior to actually accurately predicting the composition of the product, I must safeguard that now the plan context has always been limited as even the campaign moves.

4.5 Function Oriented Metrics

Table 4.1: Complexity Matrix

EI	1-4 DETs	5-15 DETs	16 or more DETS
1 FTR	Low	Low	Average
2 FTRs	Low	Average	High
3orMoreFTRs	Average	High	High

Table 4.2: Complexity Matrix 2

EO/EQ	1-5 DETs	6-19 DETs	20 or more DETS
1 FTR	Low	Low	Average
2 to 3 FTRs	Low	Average	High
4 or More FTRs	Average	High	High

Table 4.3: Complexity Matrix for UFP

Complexity	Transaction Function Type	Transaction Function Type
	EI/EQ	EO
Low	3	4
Average	4	5
High	6	7

Table 4.4: Complexity Matrix 3

ILF/EIF	1-19 DETs	20-50 DETs	51 or more DETS
1 RET	Low	Low	Average
2 to 5 RETs	Low	Average	High
6 or More RETs	Average	High	High

Table 4.5: Complexity Matrix for UFP 2

Complexity	Transaction Function Type	Transaction Function Type
	ILF	EIF
Low	7	5
Average	10	7
High	15	10

4.6 Identifying Complexity

This same duty of weighing reusable components should be included in the implementation framework. Weighing latent variables must be planned. An initial discriminant function tally must be implemented in order to offer tailoring for estimate purposes.

Transactional Functions:

- ✓ External Inputs [EI]
- ✓ External Outputs [EO]
- ✓ External Queries [EQ]

Data Functions:

- ✓ Internal Logical Files [ILF]
- ✓ External interface files [EIF]

Value Adjustment Factor (VAF) = (0.65+ (.01X TDI))

UFP = UFP (Data Function) + UFP (Transaction Function)

Adjusted Function Point **(AFP) = UFP X VAF Effort for PHP = AFP x Productivity**

4.7 Identifying complexity of transition function

Table 4.6: Identifying complexity of transition function

Transition function	**Fields/File involve**	**FTRs**	**DETs**
1. Create an account (EI)	**Fields**- role, name, email, image, address, password, contact, submit **File Name**-users	1	8
2.Login to the system(3*EI)	**Fields**- email, password, login **File Name:** users	1	3
3. Insert Bank (EQ)	**Fields**- name, email, address, contact, logo, status, submit **File Name:** users	1	6
4.Go to Bank (EI)	**Fields**- name, email, address, role, password, image, update **File Name:** users	1	7
5. Update Bank Info (EO)	**Fields**- name, contact, address, logo**,** status **File Name:** post	1	5
6. Insert Loan Seeker(EQ)	**Fields**- name, email, address, contact, image **File Name:** users	1	5

7. Loan Seeker can see his apply approval (EO)	**Field-** id, loan_id, user_id, name, status, action **File Name:**loan seeker , post	2	6
8. Administrator can approve Bank (EI)	**Fields**- id, name, address, logo, contact, email **File Name:** applied_post	1	6
9. Bank can set loan Criteria (EI)	**Fields**- loan_id, criteria_id File Name: user	1	2
10.Administrator can manage own profile (EI)	**Fields**- about, admin_image, name, contact, address, email **File Name:** admin	1	6
11.Administrator rejected bank approval from list (EI)	**Fields**- id, name, address, logo, contact, email **File Name:** bank	1	6
12.View Loan appliers Information	**Fields**- id, name, date of birth, email, phone number, address, marital status, living duration, company name, designation, monthly income, years of experience, status, remarks **File Name:** user	1	15

4.8 Identifying complexity of data function

Table 4.7: Identifying complexity of data function

Data function	**Fields/File involve**	**RETs**	**DETs**
1. admin (ILF)	**Fields-** id, name , email, contact address, password, image, role	1	8
2.loan applied_info (ILF)	**Fields-** loan_id, user_id, name, email, status, living duration, year of experience, phone, address, monthly income, company name, designation	1	13
3.criteria (ILF)	**Fields-** id, title, description, type_id, status	1	5
4.loan (ILF)	**Fields-**id, title, status, type_id, details, bank_id, loan_amount, number of amount	1	8
5.bank (ILF)	**Fields-** id, name, email, contact, address, password, logo, status	1	8
6.loan seeker (ILF)	**Fields-** id, name, email, contact, address, password, image, role	1	8

4.9 Unadjusted function point contribution

Table 4.8: Unadjusted function point contribution of transition function

Transition function	FTRs	DETs	Complexity	UFP
1. Create an account (EI)	1	8	Low	3
2.Login to the system(3*EI)	1	3	Low	3
3. Insert Bank (EQ)	1	6	Low	3
4. Go to Bank (EI)	1	7	Low	3
5. Update Bank Info (EO)	1	5	Low	3
6. Insert Loan Seeker	1	5	Low	3
7. Loan Seeker can see loan apply approval (EO)	2	6	Average	5
8. Bank can set loan Criteria (EI)	1	2	Low	3
9. Administrator can manage own profile (EI))	1	6	Low	3
10. Administrator rejected bank approval from list (EI)	1	6	Low	3
11. .View Loan appliers Information	1	15	Low	3
12. Administrator can approve Bank(EI)	1	6	Low	3
Total				38

Table 4.9: Unadjusted function point contribution for data functions

Data function	RETs	DETs	Complexity	UFP
1. admin (ILF)	1	8	Low	7
2. loan applied_info(ILF)	1	13	Low	7
3. criteria (ILF)	1	5	Low	7
4. loan (ILF)	1	8	Low	7
5. bank (ILF)	1	8	Low	7
6. loan seeker (ILF)	1	8	Low	7
Total				42

4.10 Performance and Environmental impact

Table 4.10: Performance and environmental impact

GSC (General System Characteristics)	TDI
1. Information Communications	4
2. Uniformly distributed Data Analysis	2
3. Deployment	4
4. Frequently Employed Layout	2
5. Purchase Percentage	2
6. Web - based data submission	1
7. Consumer productivity	2
8. Download Patch	1
9. Sophisticated Mission	2
10. Versatile afterwards	2
11. Simplicity of Implementation	4
12. Convenience of Administration	3
13. Too many Venues	3
14. Influence The behaviors	1
Total Degree of Influence (TDI) (Range 0 - 70 -> influence size ±35%)	33

4.11 Counting Function Point

Value Adjustment Factor (VAF) = (0.65 + (0.01*TDI))

= (0.65 + (0.01*33))

= 0.98

UFP = UFP (Data Function) + UFP (Transition Function)

= 42+38

=80

Adjusted Function Point (AFP) =UFP*VAF

= 80*0.98

= 78.4

Efforts for Project = AFP * Productivity

= 78.4 × 15.5

= 1215.2 per hour

A single person sets in eight hours each and every day=1215.2/8

= 151.9days

A group has two guys in it,

=151.9/2

= 75.95days

= Which is around 76 days

25 days from each month are days at work,

=76/25

=3.04months

3 Months is required to finish the project.

4.12 Processed Based Estimation

Table 4.11: Processed based Estimation

Function Name	RG (Client Visit)	Data Analysis	Risk Analysis	Engineering		Developing		Testing	Implementation And Support	Total
				Planning	Design	Coding	Documentation			
F1	0.2	0.9	0.1	0.1	0.6	1.62	0.6	0.9	0.4	5.42
F2	0.6	1.0	0.01	0.01	0.82	2.5	0.82	1.44	0.42	7.62
F3	0.2	0.8	0.1	0.2	0.6	1.9	0.6	0.9	0.4	5.7
F4	0.2	0.7	0.1	0.1	0.5	2.01	0.5	0.7	0.79	5.6
F5	0.3	0.8	0.02	0.1	0.5	1.5	0.5	0.8	0.3	4.82
F6	0.2	1.0	0.1	0.14	0.5	2.0	0.7	1.37	0.5	6.51
F7	0.4	1.8	0.1	0.3	0.5	1.6	0.5	0.8	0.6	6.6
F8	0.4	1.1	0.1	0.3	0.7	2.3	0.7	1.5	0.5	7.6
F9	0.4	0.9	0.1	0.2	1.5	1.1	0.5	0.9	0.3	5.9
F10	0.2	1.2	0.2	0.49	0.88	1.3	1.51	1.4	0.5	7.68
F11	0.4	0.8	0.5	0.5	0.5	1.4	0.6	0.6	0.5	5.8
F12	0.4	1.6	0.23	0.1	0.7	2.1	0.87	1.2	0.5	7.7
F13	0.5	0.6	0.1	0.1	0.5	2.4	0.4	1.0	0.45	6.05
									Total	83
Total	4.4	13.2	1.76	2.64	8.8	23.73	8.8	13.51	6.16	
Percentage	5%	16%	2%	3%	10%	31%	10%	16%	7%	100%

4.12.1 Effort Distribution

This same reference implementation forecasting model consequences an extrapolation of the physical servers expected to accomplish the incremental methodologies in this project. 40% of the total of the filled to the brim software quality has already been earmarked for implementation, 35% for interpretation, prototype, and the remaining 25% for requirements specification and empower.

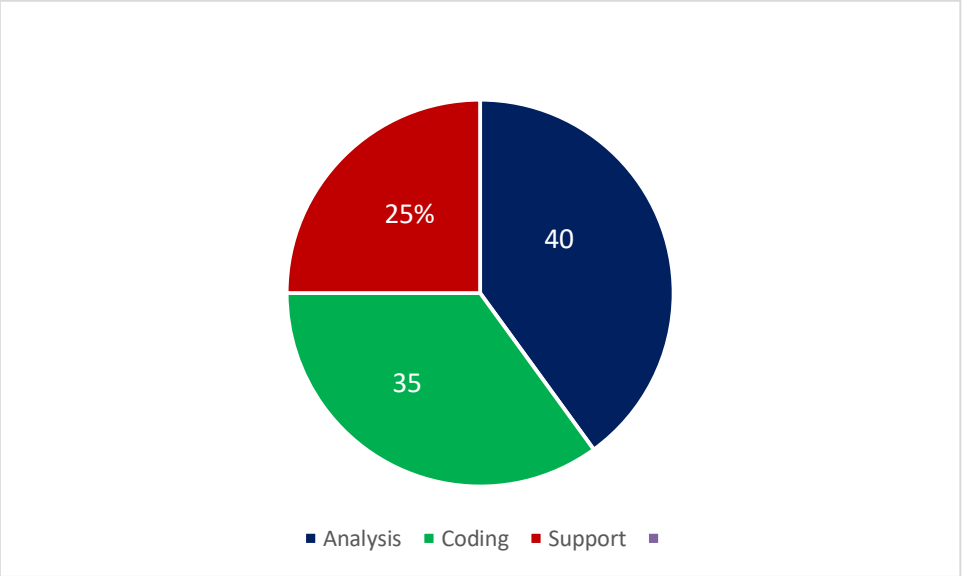


Figure 4.1: Effort Based Estimation

A detailed view of the distribution chart is illustrated bellow:

We recognize it in the Spread Sheet., among 35% CC we spend 20% for planning and collect all requirement gathering, 15% for design the data and information, 15% for code, 10% for testing and 7% for implementing.

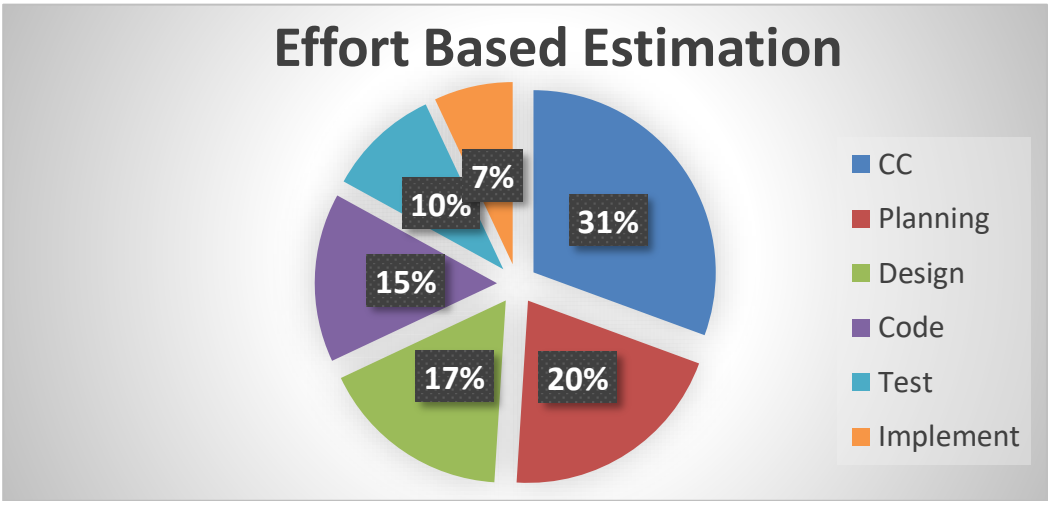


Figure 4.2: Effort Distribution PIE Chart

4.12.2 Project Schedule Chart

Overall, paradigm development seems to be a treasure trove of workloads. Most such duties ought to be pursued in a concise and collaborative way. This same project timeline serves as the system developer's guide. The timetable for this project is as follows:

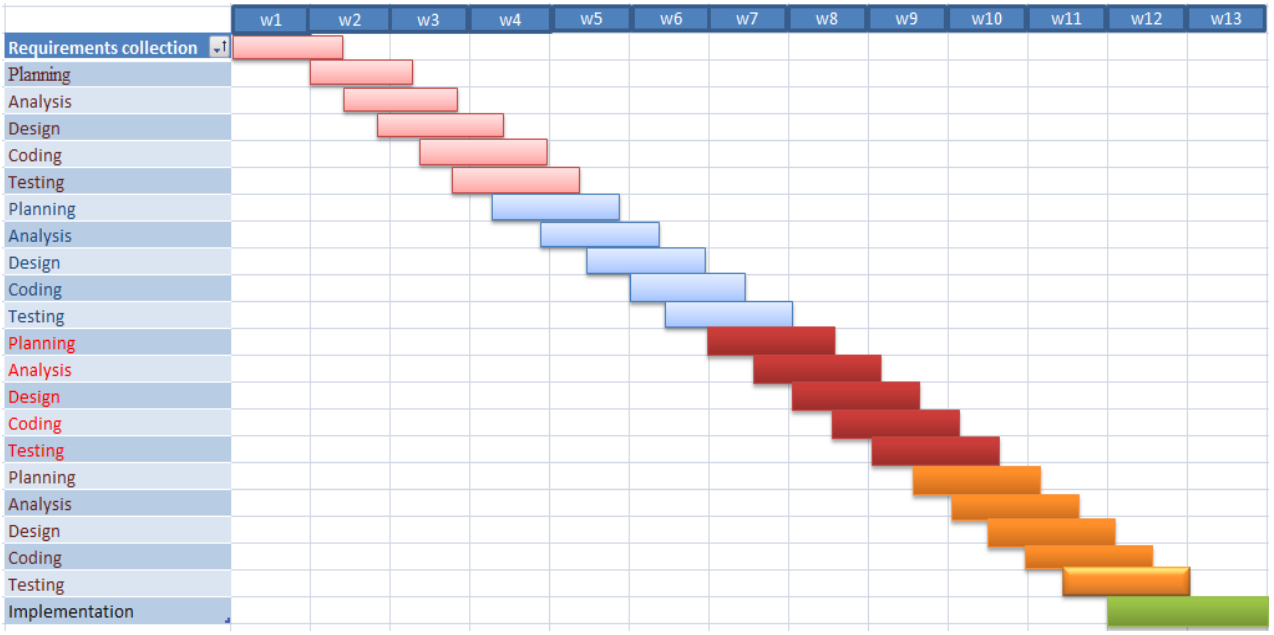


Figure 4.3: Project Schedule Char

4.13 Cost Estimation

An expenditure estimate seems to be an interpolation of an agency's overheads. There have only been several other components to contemplate unless dealing with different types of this job. Illustrated below,

- Personal costs
- Software costs
- Hardware costs
- Other costs

4.13.1 Personnel cost

- Number of days in a year = 365
- Number of government holidays in a year =24

- Number of weekly holidays in a year =52
- Total number of working days to develop the project =365-(52+24) =289 days
- Total number of working days per months to develop the project =289/12 =24.083 days
- Organization working hours per day = 8 hours
- Organization working hours per month = 24.083*8 = 192.664hours

Table 4.12: Personnel Cost

Designation	**Person**	**Months**	**Salary**
System Analyst	1	1	13000
Planner	1	1	12000
Designer	1	1	18000
Coder	2	1	40000
Tester	1	1	12000
Total			95000BDT

4.13.2 Hardware cost

Cost of the computer that used to complete the project.

Table 4.13: Hardware Cost

Name	Price
DELL Laptop	50,000
Mouse	350
Total	50350 BDT

4.13.3Software cost

Table 4.14: Software Cost

Name	Amount
Windows 10	No Cost
MS Office 2016	No Cost
XAMPP	No Cost
MySQL	No Cost
VS Code	No Cost
Google Chrome	No Cost

4.13.4 Others Cost

Table 4.15: Estimation of Other Cost

Name	Price
Internet Bill	3000
Transport	3000
Mobile	800
Extra	15000
Total	8300 BDT

4.14 Account Table

Table 4.16: Accounts Table

Name	Price
Personnel Cost	95000 BDT
Hardware Cost	50350 BDT
Software Cost	0 BDT
Other Cost	8300 BDT
Total	1,53,650 BDT

Chapter 5: Risk Engineering

5.1 Risk Management

A risk is a major situation that may or may not occur. Analyzing possible hazards in a project is essential. Many difficulties might arise if the risks of a software project are not appropriately examined and estimated. Anyone creating any form of system will confront it, and it must be controlled.

5.2 Stages of Risk

A risk analysis and management process help a software team identify and manage uncertainty. Many issues can afflict a software project. A risk is a prospective problem; it may or may not occur. Regardless of the conclusion, it's a good idea to identify it, analyze its likelihood of occurrence, estimate its impact, and develop a contingency plan should the problem materialize. Risk analysis and management are a set of processes that assist software developers in understanding and managing uncertainty. The following stages are taken to build a risk management model:

Identification: The practice of recognizing possible risks or hazards through data collection is known as risk identification. There are several data collection and manipulation technologies and methodologies available. The team is collecting data and characterizing possible dangers to Web sites using both automated and human methodologies. Web crawling is one efficient method of gathering information on the status of Web pages and sites:

Classification: The process of building a structured model to categorize risk and fitting observable risk features and occurrences into the model is known as risk categorization. The team used both quantitative and qualitative methodologies to identify and categorize the hazards to Web pages, Web sites, and hosting servers.

Assessment: The process of determining relevant risk scenarios or sequences of events that potentially result in harm or loss, as well as the likelihood of these events, is known as risk assessment. Rosenthal defines a general risk assessment standard as "clear, coherent, consistent, complete, comprehensive, unbiased, uniform, balanced, defensible, sustainable, adaptable, and supported by appropriate and sufficient instruction."

Analysis: The potential effect of risk patterns or scenarios, the potential extent of loss, and the direct and indirect costs of recovery are all determined through risk analysis. This stage analyzes

vulnerabilities, takes into account the organization's willingness to tolerate risk given the potential repercussions, and creates mitigation strategies.

Implementation: Implementing risk management policies, processes, and mechanisms describes how to manage and respond to identified hazards. The program should be designed to balance the value of assets with the direct and indirect expenses of avoiding or recovering from damage or loss. To properly maintain a web-based system, we must consider the following factors:

- The hardware and software environment, including any operating system and Web server updates, security patch installation, removal of insecure services, use of firewalls, and so on.
- Administrative tasks such as contracting with dependable service providers and renewing domain name registration, among others.
- Network configuration and upkeep, such as load balancing, traffic control, and usage monitoring.
- Backup and archiving policies and procedures, such as backup media selection, media replacement intervals, backup frequency, and storage location
- The physical location of the server, as well as its susceptibility to fire, flood, earthquake, electric power anomalies, power outages, temperature variations, theft, and vandalism.

5.3 Categories of Risk

There are several types of hazards that must be addressed in each software project. This software project took into account the following risk areas:

- **Project risks:** These hazards jeopardize the project's schedule. If these risks materialize, it is likely that the project's timeline will be pushed back and prices will rise. Potential budgetary, scheduling, human resource, customer, and requirement issues, as well as their influence on the software project, are identified as project risks.
- **Technical risks:** These risks jeopardize the quality and timeliness of the program being developed. Implementation may become difficult or impossible if a technological risk becomes a reality. Potential design, implementation, interface, verification, and

maintenance issues are identified as technical risks. Furthermore, specification ambiguity, technical uncertainty, and technological obsolescence are risk issues.

- **Business risks:** These risks jeopardize the viability of the program being developed. The business risks might include market risks, such as designing a system that no one wants. Strategic hazards include developing a system that no longer fits within the company's overall business plan. Management risks losing senior management support owing to a shift in focus or a shift in personnel. Budget hazards include the loss of budgetary or personnel commitment.

5.4 The RMMM Plan

- **Risk Mitigation:** Proactive planning for risk avoidance.
- **Risk Monitoring:** Determining whether or not expected risks materialize, ensuring preventative measures are implemented correctly, collecting data for future risk analysis, and attempting to determine which risks caused which problem
- **Risk Management:** Actions to be taken in the event that mitigation steps have failed and the risk has become a live problem.

Type of Impact: Catastrophic (1), Marginal (2), Tolerable (3), Critical (4).

Type of Probability: very low (<10%), low (10–25%), moderate (25–50%), high (50–75%), very high (>75%)

Project Risks: In my system, the bellow mentioned projects risks I needed manage.

5.5 Project Risks:

Table5.1 Project Risk (PR01)

Project Risk(PR01)	Date:
Name	Changes the requirements
Probability	Low (25%)
Impact	Marginal (2)
Description	User may change their requirement
Mitigation and Monitoring	The corporation redefines requirements based on time or commercial demands. Regular meetings will be conducted with the firm. This assures us that the product we are creating is a solution to a problem.
Management	Emergency meeting between both parties to identify new project requirement and goals.
Status	Not occur.

Table5.2 Project Risk (PR02)

Project Risk (PR02)	Date:
Name	Poor Quality Documentation
Impact	Catastrophic
Description	Quality of documentation may poor
Mitigation and Monitoring	A meeting will be organized on a regular basis to give documentation proposals and themes. Documentation progress will also be monitored at each meeting.
Management	We will convene a meeting to discuss document quality enhancement. The insertion of new subjects into the documentation will be delegated to the appropriate individual.
Status	Monitoring it.

Technical Risks

Table5.3 Technical Risk (TR01)

Technical Risks (TR01)	Date:
Name	Computer Crash
Impact	Catastrophic
Description	Computer may crash due to several reasons.
Mitigation & Monitoring	We should take proper follow up of computers. We also take regular data backup

	every day and we can use IPS to stop unexpected shutdown.
Management	If our computer has been crashed then we will restore backup.
Status	We are not facing such kind of problem yet.

Table5.4 Technical Risk (TR02)

Technical Risk (TR02)	**Date:**
Name	Technology doesn't meet specifications.
Impact	Catastrophic
Description	Customer doesn't have the technology to their desired specification.
Mitigation & Monitoring	Ensures that the product we are creating and the customer's specifications are equal.
Management	The client should be contacted quickly, and whatever efforts are necessary to correct the situation should be made. A meeting between the development team and the client is preferable to discuss this problem in depth.
Status	We are not facing such kind of problem yet.

Business Risks

Table5.5 Business Risk (BR01)

Business Risk (BR01)	Date:
Name	Insufficient Budget
Impact	Marginal
Description	If the budget is low project may not complete
Mitigation & Monitoring	The project requires a streaming server, which is expensive to set up. To lessen the budget risk, we located numerous alternative streaming options.
Management	Refinement in project goal. A new plan for regulate the budget.
Status	We are not facing such kind of problem yet.

Table5.6 Business Risk (BR02)

Business Risk (BR02)	Date:
Name	Not pay the installment of Software Cost.
Impact	Catastrophic
Description	Customer doesn't pay for the installment of software cost.
Mitigation & Monitoring	We must guarantee that all customers communicate well and that the full installation is done.
Management	The only option would be to investigate the cause and devise a remedy.
Status	The risk has not been arisen yet.

Chapter 6: Analysis Modeling

Analysis modeling uses a combination of text and diagrammatic forms to portray data, function, and behavior requirements in a way that is reasonably simple to grasp and, more importantly, simple to check for accuracy, completeness, and consistency. This section contains resources for traditional and object-oriented analysis (OOA) methodologies, as well as UML resources.

6.1 Software Analysis Pattern

Objectives of analysis Pattern

- Domain Investigation.
- Describe the client's needs.
- Lay the groundwork for the development of a software design.
- Create a list of criteria that can be validated after the software is completed.

6.2 Activity Diagram

Activity diagrams are graphical representations of processes of sequential activities and actions that allow for selection, iteration, and concurrency. Activity diagrams are meant to depict both computational and organizational processes in the Unified Modeling Language. Activity diagrams depict the entire control flow.

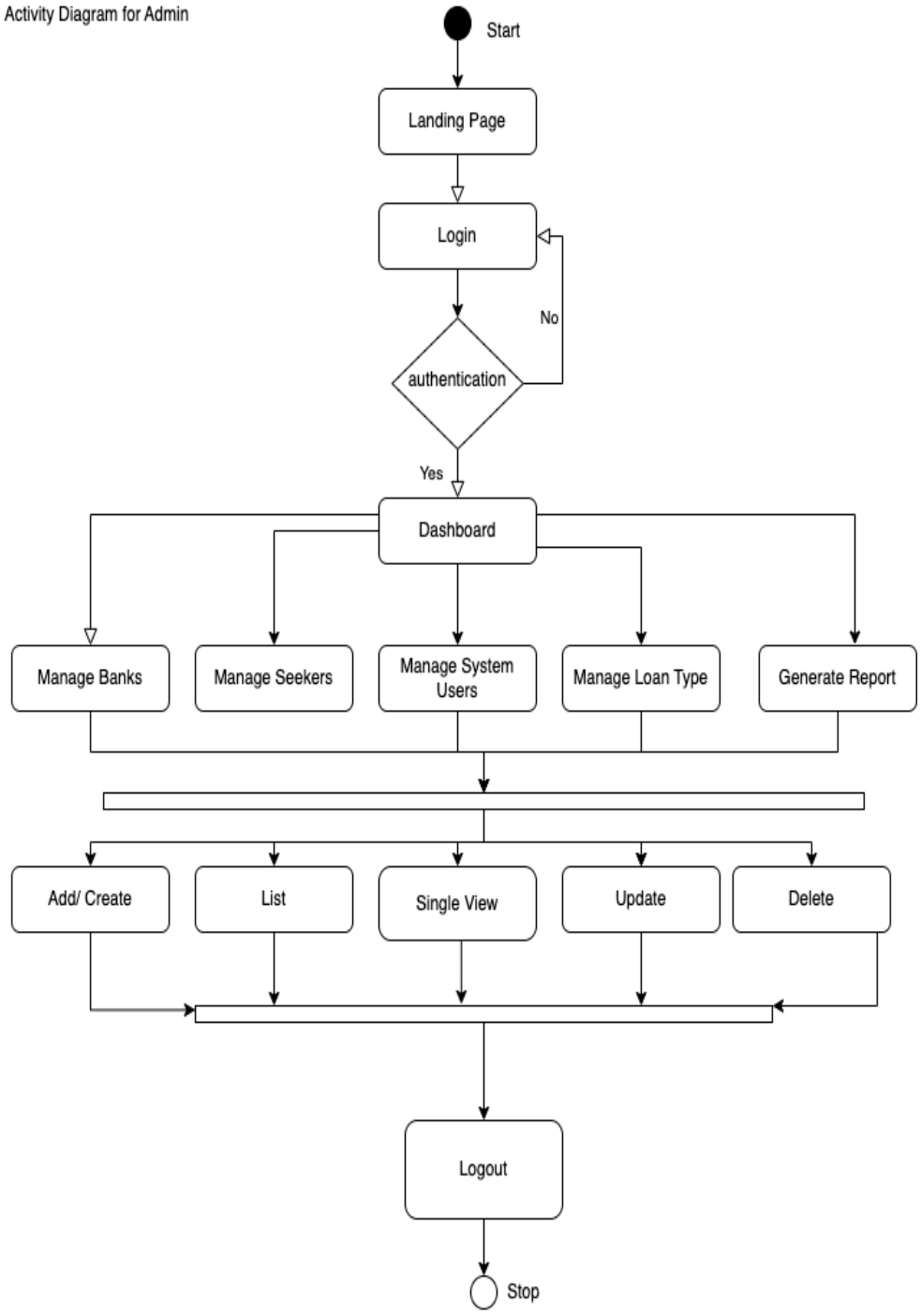


Figure 6.1: Activity Diagram for Admin

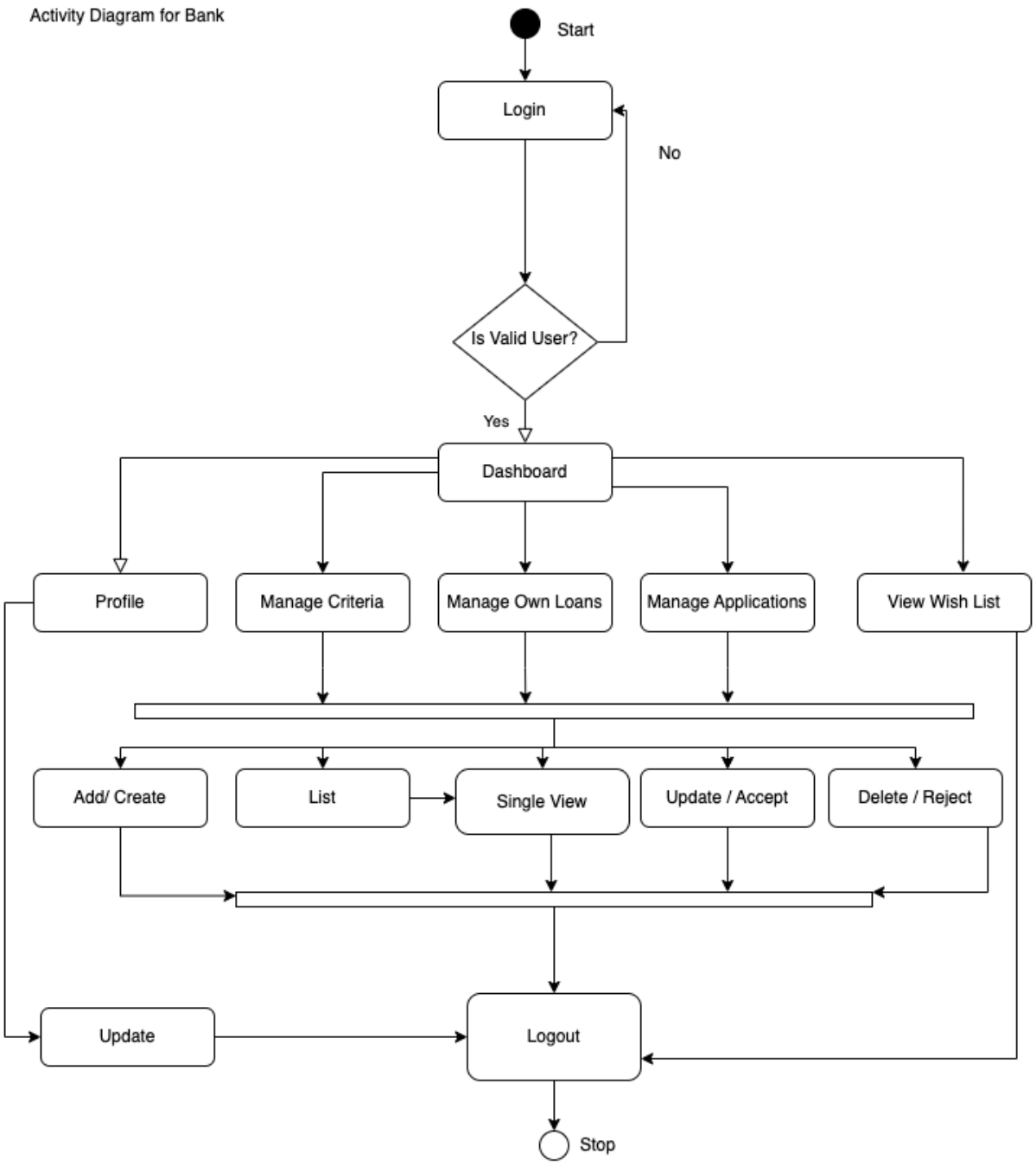


Figure 6.2: Activity diagram for Bank

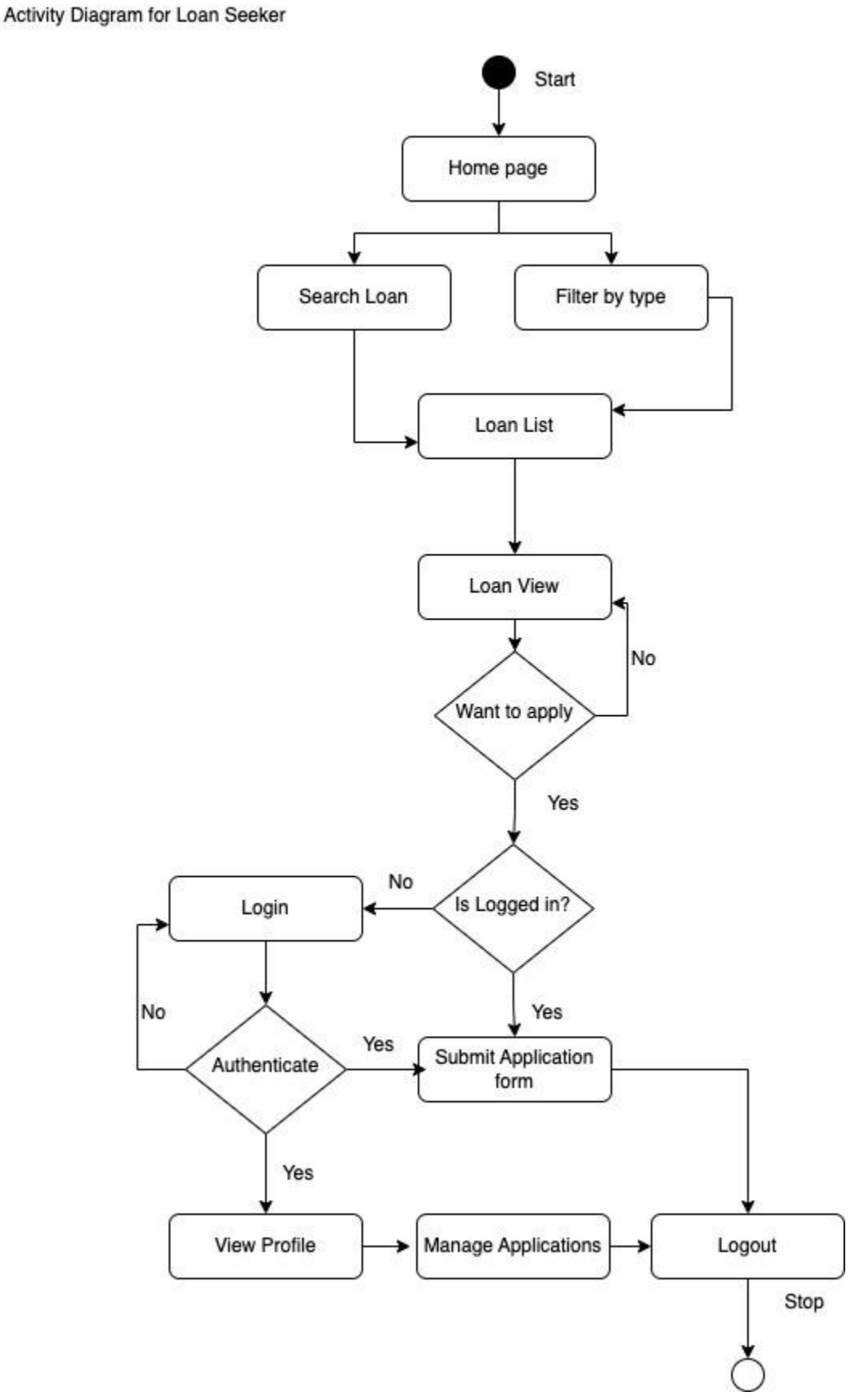


Figure 6.3: Activity diagram for Loan Seeker

6.3 Swim Lane Diagram

A swim lane diagram is a form of flowchart that shows who is responsible for what in a process. Using the metaphor of lanes in a pool, a swim lane diagram improves clarity and accountability by putting process stages inside the horizontal or vertical "swim lanes" of a particular person, work group, or department. It depicts the connections, communication, and handoffs that occur between various lanes, and it may be used to reveal waste, redundancy, and inefficiency in a process.

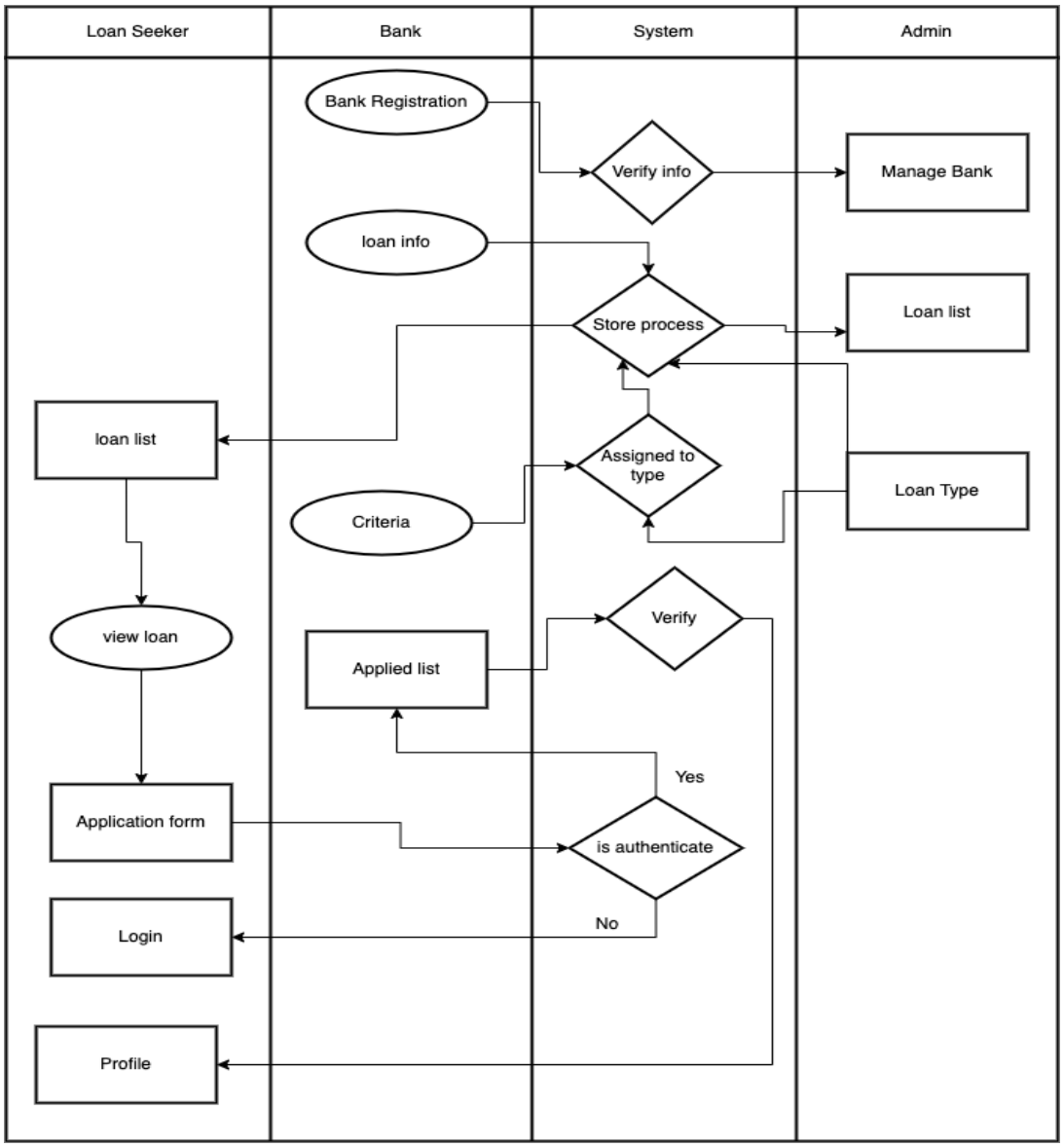


Figure 6.12: Swim Lane Diagram for Admin ,Bank and Loan Seeker

Chapter 7: Designing

7.1 Design of the system

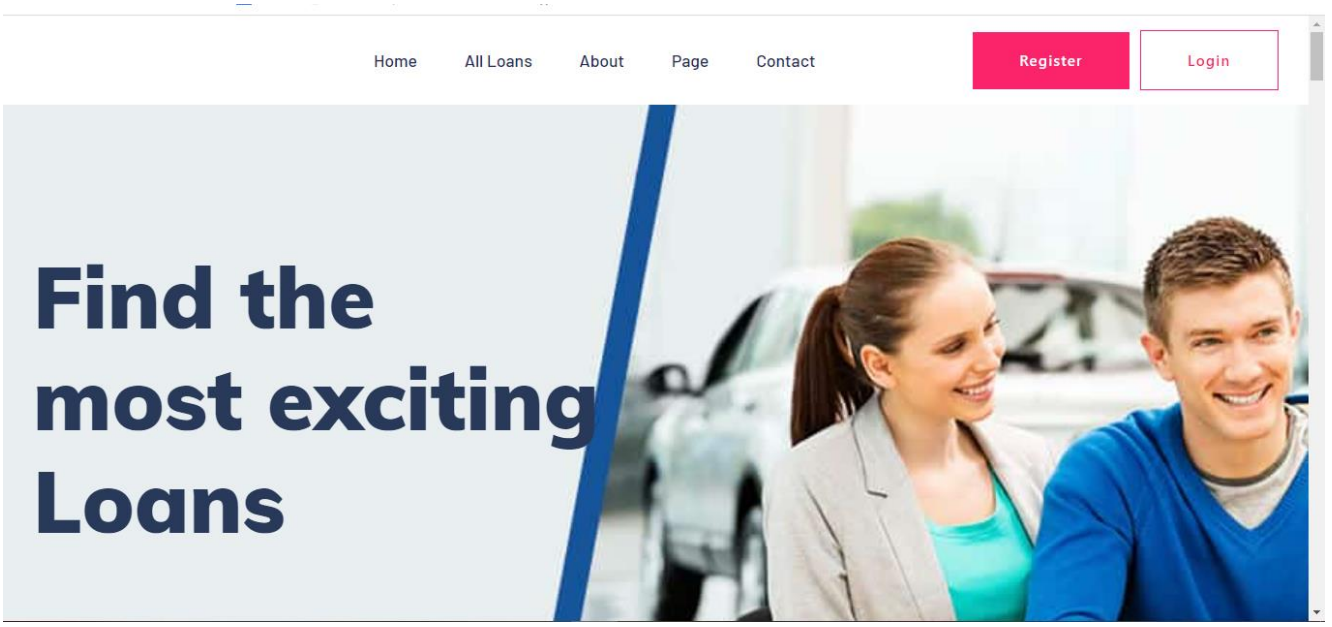


Figure 7.1: Website design page

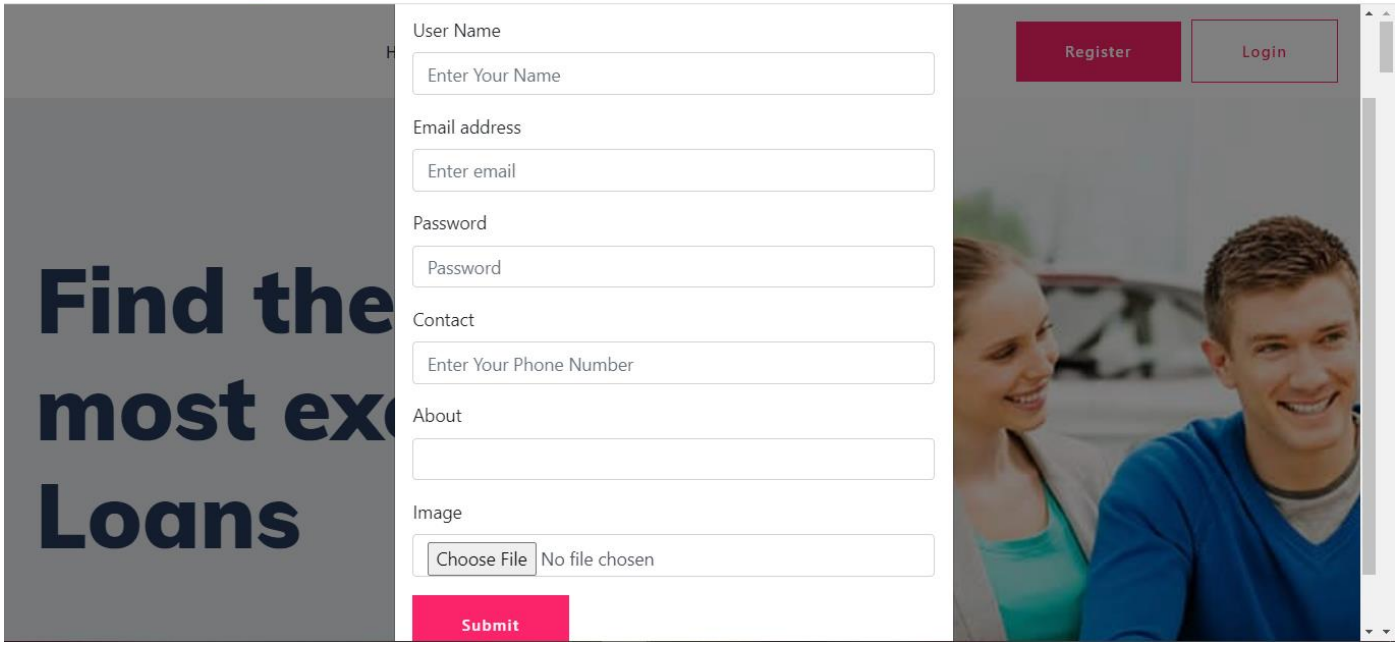


Figure 7.2: Registration interface for user

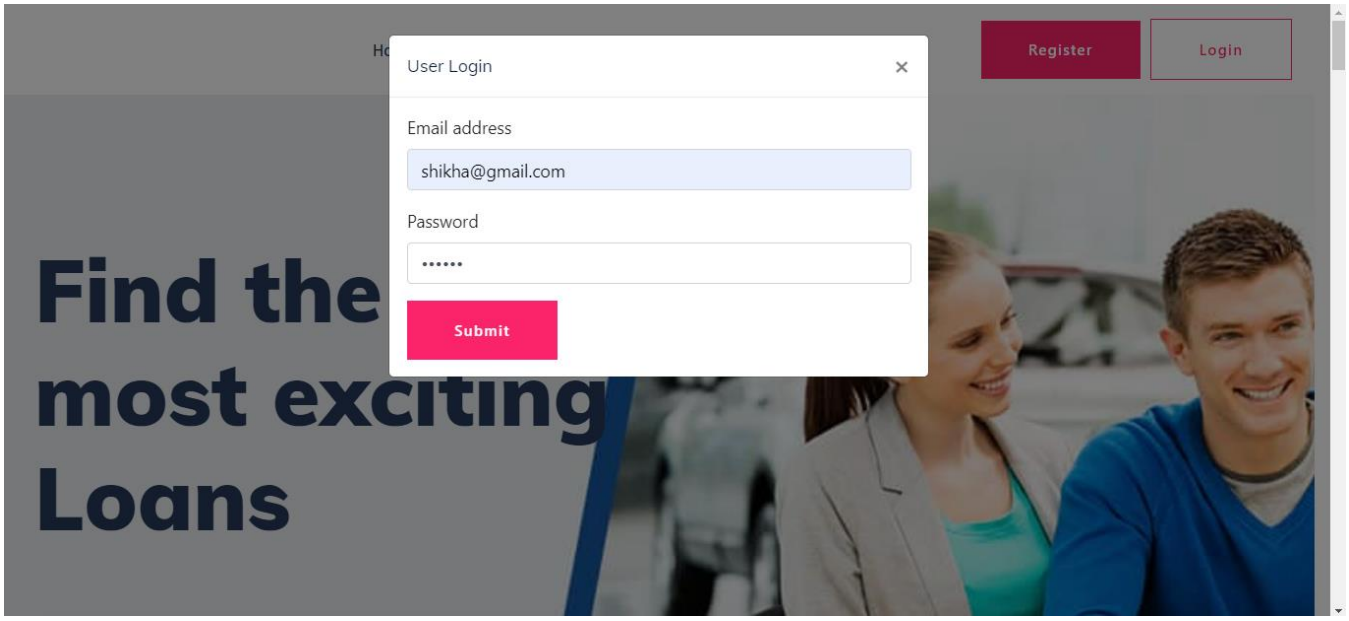


Figure 7.3: Login interface for User

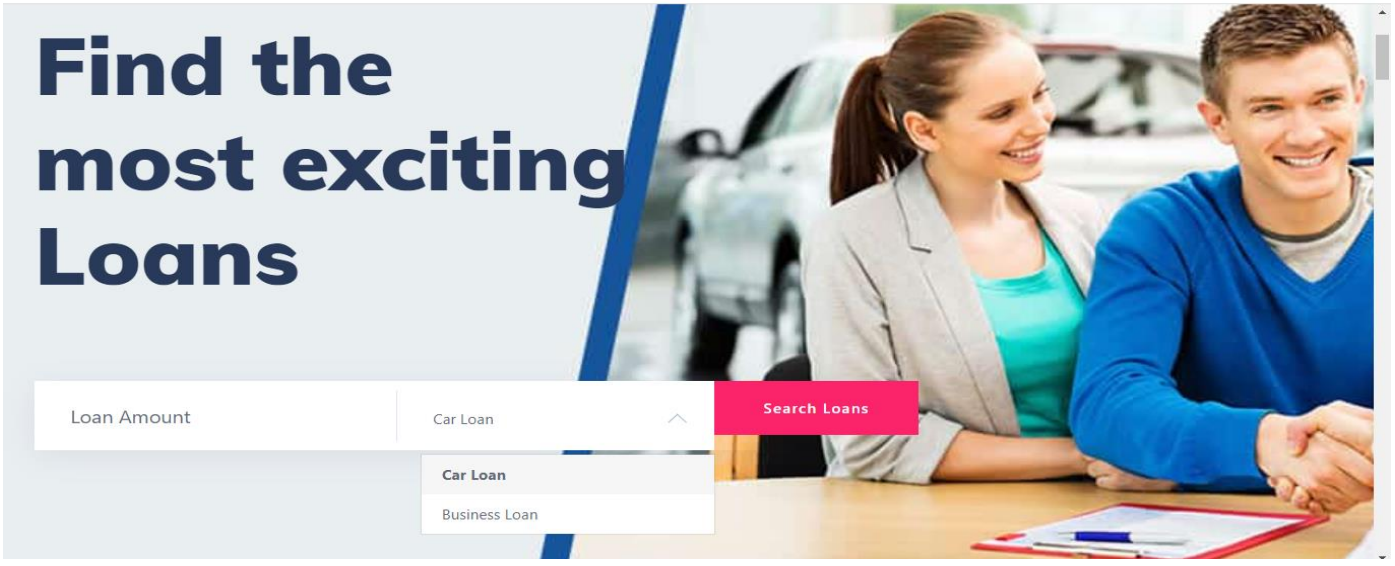


Figure 7.4: Search Loan Type interface

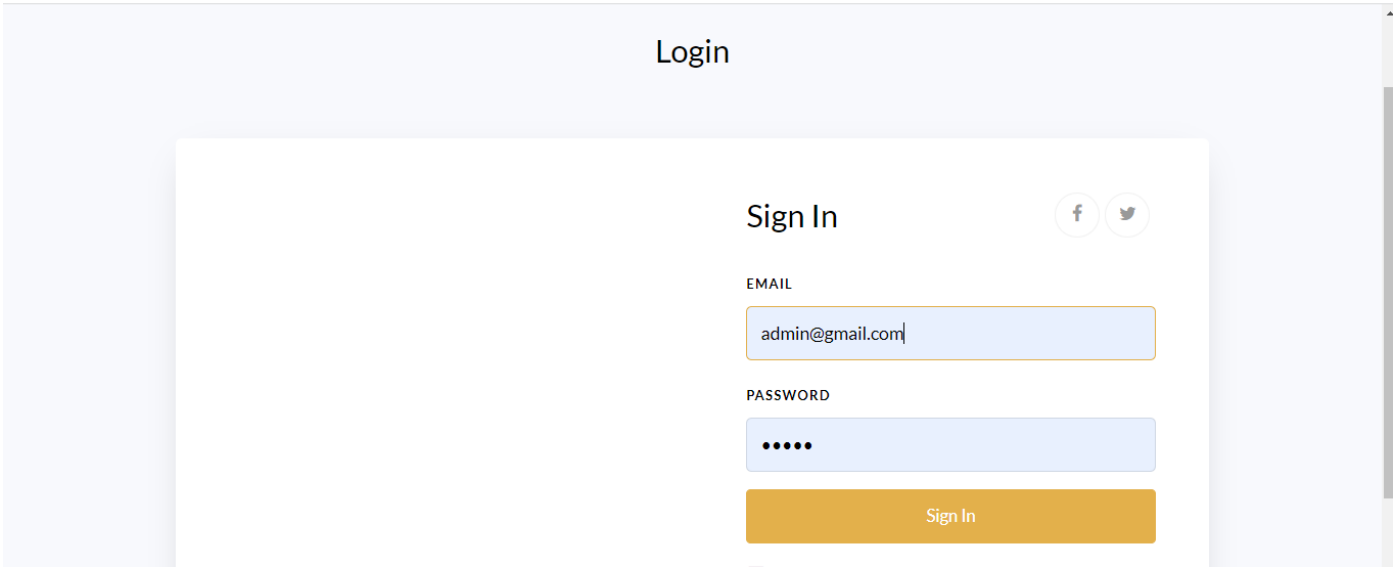


Figure 7.5: Admin login page

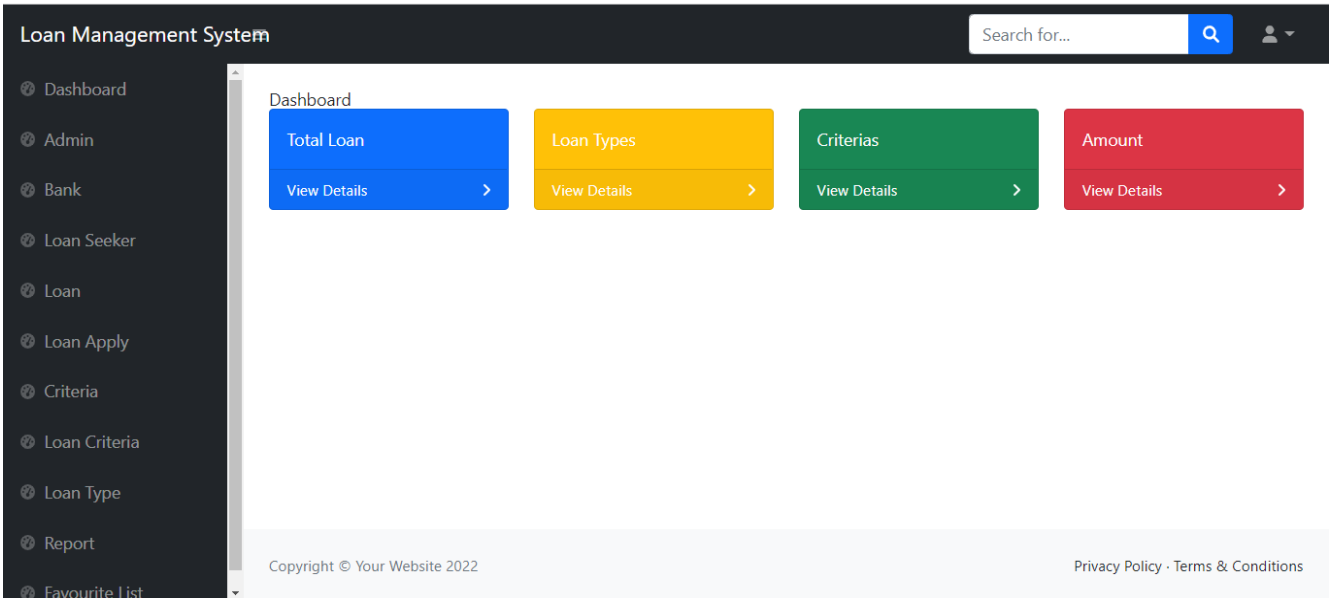


Figure 7.6: Interface design of Admin Dashboard

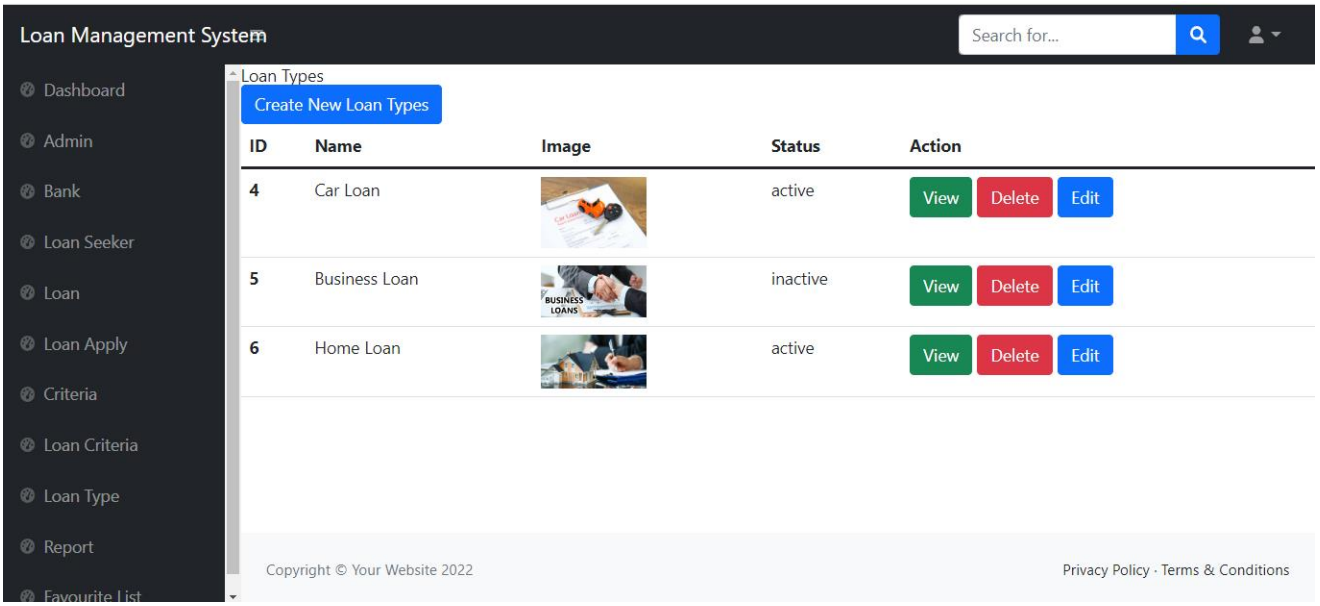


Figure 7.7: Interface design of Loan Types page

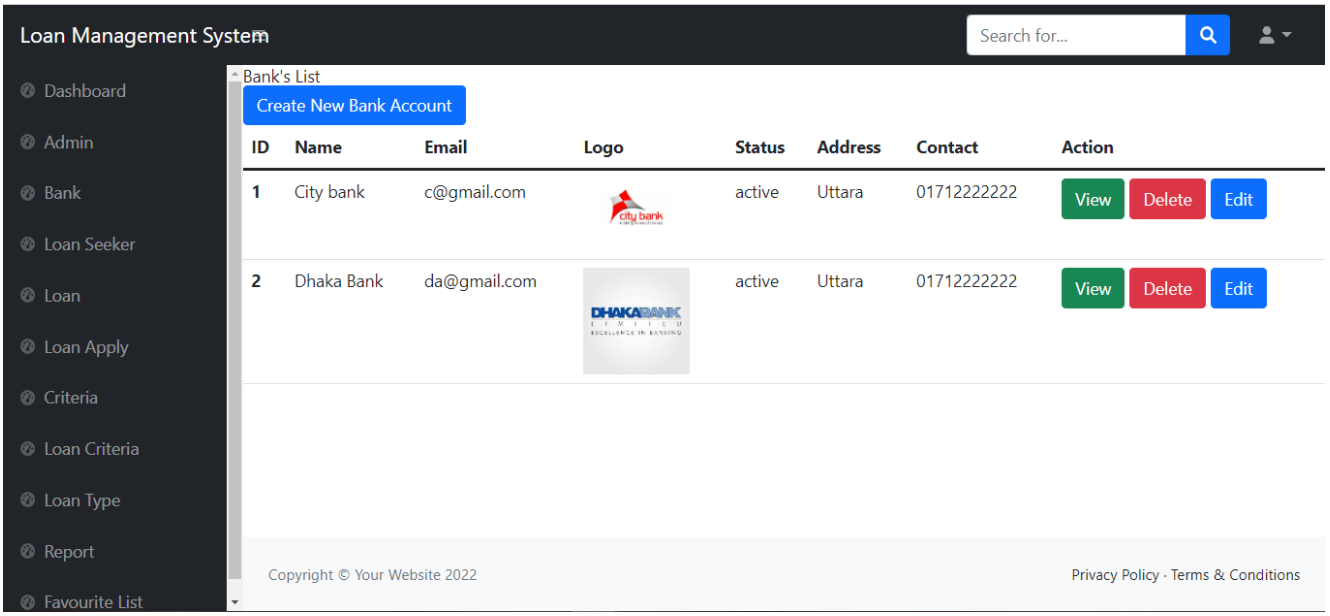


Figure 7.8: Interface design of Bank List page

7.2 Data Flow Diagram

A Data Flow Diagram (DFD) is a classic visual depiction of a system's information flows. A DFD diagram shows the information flow for any process or system. It shows data inputs, outputs, storage sites, and the pathways between each destination using predetermined symbols such as rectangles, circles, and arrows, as well as brief text labels. It demonstrates how data enters and exits the system, what alters the data, and where it is kept.

Context Level Diagram (0 Level)

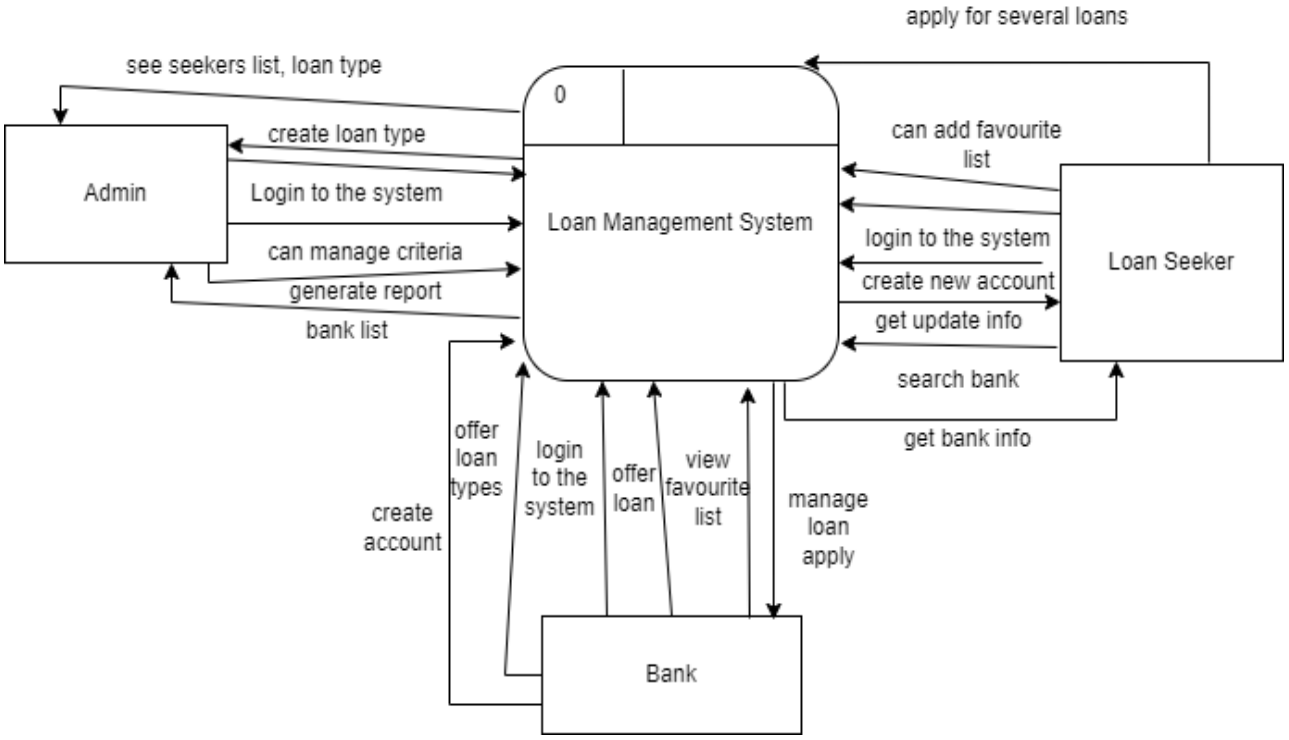


Figure 7.9: Data Flow Diagram for Context Level (level 0)

Level 1 Data Flow Diagram

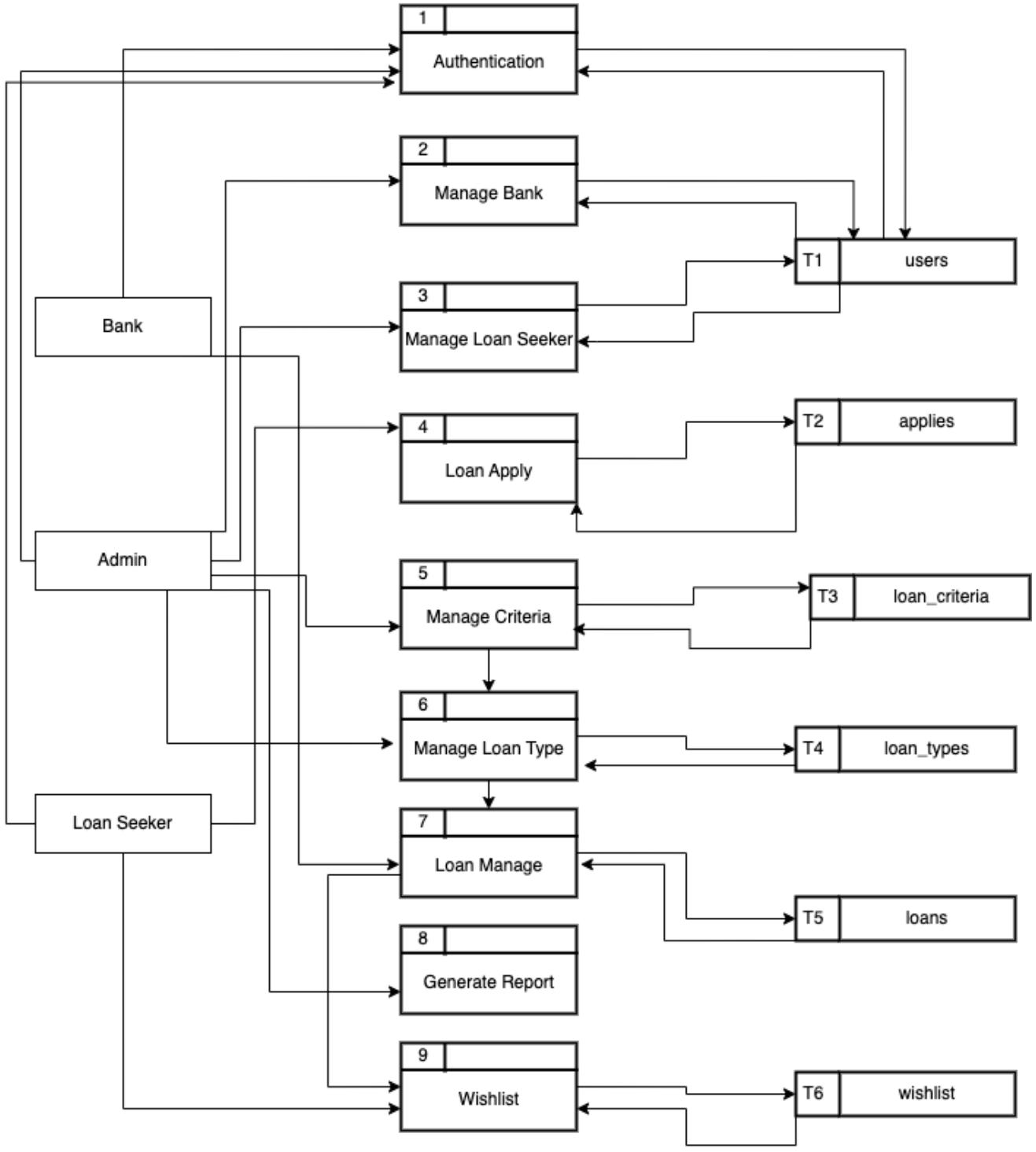


Figure 7.10: Data Flow Diagram for Level 1

7.3 Database Design

A database is an organized collection of data. A relational database, more restrictively, is a collection of schemas, tables, queries, reports, views, and other elements. Database designers typically organize the data to model aspects of reality in a way that supports processes requiring information, such as (for example) modeling the availability of rooms in hotels in a way that supports finding a hotel with vacancies.

The process of creating a detailed data model for a database is known as database design. This data model includes all of the logical and physical design options, as well as physical storage parameters, that are required to construct a design in a data definition language, which can then be used to establish a database. Each entity in a properly attributed data model has extensive attributes.
Designing a database is a two-step process:

1. Information-level design: In this step, user requirements are gathered together and a database is designed which will meet these requirements as clearly as possible. This step is called Information Level Design. Information level design is completed independently of any particular DBMS.
Information-Level Design Method:
For each user view:

- Represent the user view as a collection of tables
- Normalize these tables
- Identify all keys in these tables

2. Physical-level design: Information level design is transferred into a design for the specific DBMS that will be used to implement the system in question. This step is called Physical Level Design, concerned with the characteristics of the specific DBMS.

- Information-level design adapted for the specific DBMS that will be used
- Must consider characteristics of the particular DBMS
- Undertaken after information-level design completion
- Most DBMSs support primary, candidate, secondary, and foreign keys

In the field of relational database design, normalization is a systematic way of ensuring that a database structure is suitable for general-purpose querying and free of certain undesirable characteristics—insertion,

update, and deletion anomalies that could lead to loss of data integrity.

7.4 Entity Relationship Model

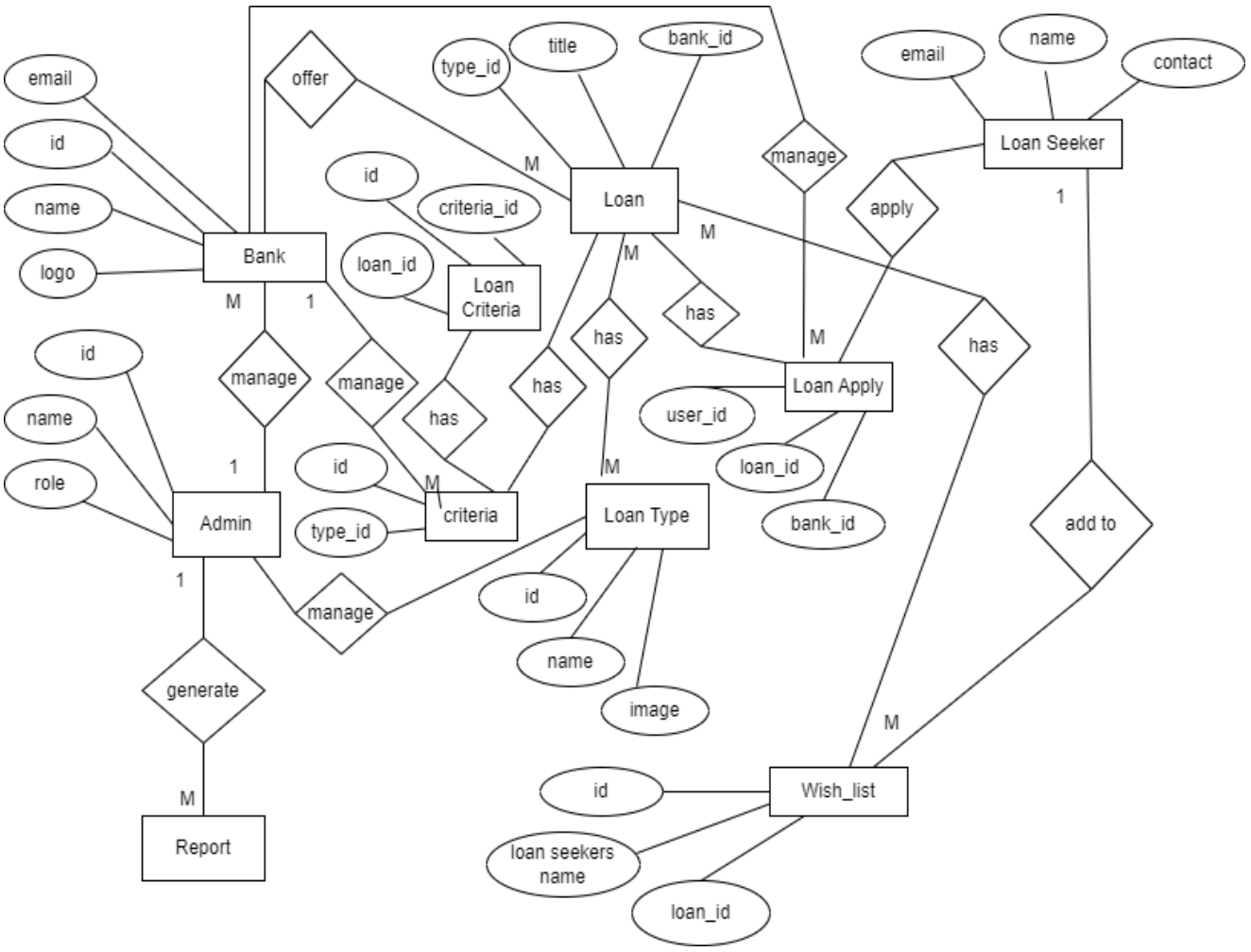


Figure 7.11: Entity Relationship Diagram

7.5 Database Table Structure

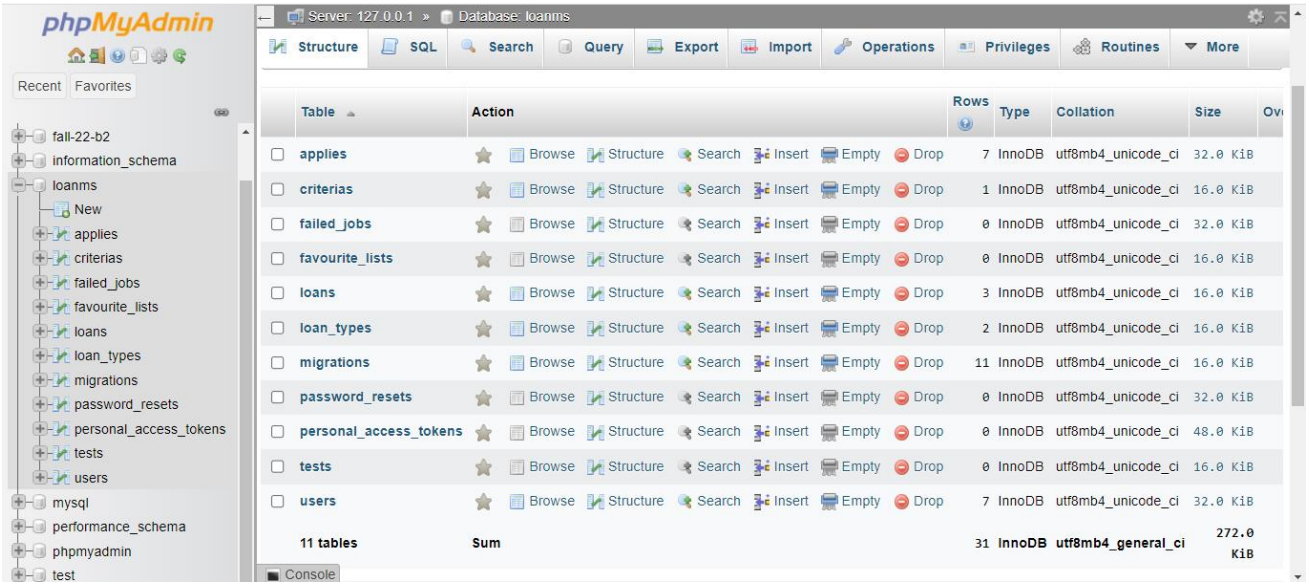


Figure 7.12: All Database Table Structure

Chapter 8: Quality Assurance and Testing

8.1 System Quality Management

A regulated or ISQ-compliant organization must have an automated quality management software system that integrates all departments. A QMS (quality management system) or TQM (total quality management) system may link each phase of a product's development lifecycle to every department in a corporation. There are two kinds of quality. n.d. (TechTarget)

There are mainly two types of quality, they are given below:

- Internal Quality
- External Quality

Internal quality:

- Test coverage
- Testability
- Portability
- Thread-safeness
- Conciseness
- Maintainability
- Documentation
- Legibility
- Scalability

External quality:

- Features
- Speed
- Space
- Network usage
- Stability
- Robustness
- Ease-of-use
- Determinism

- Back-compatibility

8.1.1 Software Quality Management Process

- The purpose of Software Quality Management (SQM) is to monitor the quality of software and development, as well as the development process.
- A quality product is one that meets its specifications and pleases the user.
- A quality culture is an organizational climate in which everyone is responsible for quality.

8.1.2 Quality Assurance Matrix

The quality assurance matrix of the project "Online Tutor Finder System" entails implementing particular quality processes and ensuring that these attributes are followed. The level of quality in this software's system

8.2 System Testing

System testing is the examination of a fully integrated software product. In most cases, software is merely one component of a broader computer-based system. Finally, software is linked to other software/hardware systems. System testing is a collection of tests designed to put the entire computer-based system through its paces.

Two Category of Software Testing

- Black Box Testing
- White Box Testing

8.2.1 Black-box Testing

Black-box testing, also known as behavioral testing, focuses on the software's functional requirements. It allows the software engineer to create sets of input circumstances that completely exercise all functional requirements for a program. The black-box testing approach will be used to test the loan recommendation and application system modules.

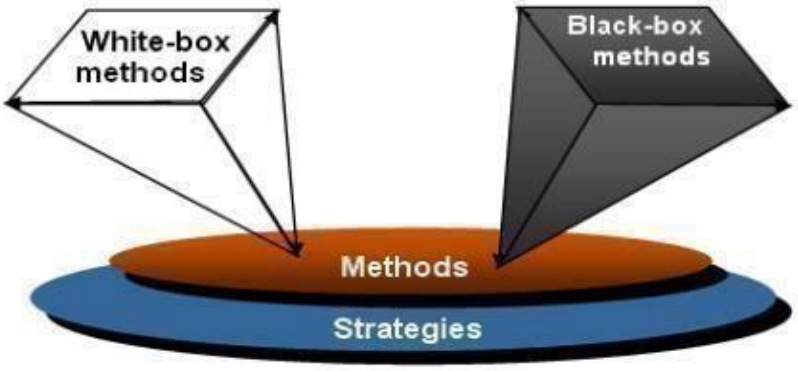


Figure 8.1: Black box and White box Testing Source

8.2.2 White-box Testing

White-box testing, commonly described as glass-box testing, is a test case design strategy that extracts test cases from the control structure of the process or system. Software developers can generate test cases using white-box testing techniques that,

1. Software developers can generate test cases using white-box testing techniques.
2. Go through all logical conclusions and check their true and false sides.
3. Execute all loops within their operational constraints and borders.
4. Test internal data structures to guarantee their accuracy.

White-box testing will be used for modules that involve sophisticated computations or decision-making code, such as checking the availability of a library item.

8.3 System Testing Design

Table:8.1 Test Case 1

Testing Scenario No: 1	
Scenario	User Login testing scenario of my system
Input's	E-mail, password of User for Login
Desired Output's	When enter E-mail, password then get access level define.
Actual Output's	For login my system work correctly
Verdict	Getting result from Desired Output's and Actual Output's decided this system is successful for login.

Table:8.2 Test Case 2

Testing Scenario No: 2	
Scenario	Admin can manage banks, loan seeker, loan types
Input's	Insist to manage banks, loan seekers, loan types
Desired Output's	Flash the data to the admin
Actual Output's	Since disporting showing all records my framework works accurately.
Verdict	The interaction is operated accurately also effectively.

Table:8.3 Test Case 3

Testing Scenario No: 3	
Scenario	Admin can manage loan appliers application
Input's	Insist to manage loan appliers application
Desired Output's	Flash the data to the bank
Actual Output's	Since disporting showing all records my framework works accurately.
Verdict	Getting result from the output's decided this system is successfully showing loan appliers application.

Table:8.4 Test Case 4

Testing Scenario No: 4	
Scenario	Loan Seeker can view loan type, criteria, loan
Input's	Click on loan, loan type, criteria
Desired Output's	When enter all fundamental data accurately, all information show
Actual Output's	Reports will be show by day to day
Verdict	Come by result from wanted result's this framework is showing the informations

Chapter 9: Conclusion

9.1 Preface

The age of contemporary science and communication is crucial to the creation of more effective operational and managerial processes. to give better and more consistent service to employees in order to maintain service at all times. I was fortunate and privileged to be given the opportunity to work with some of these efficient, hard-working, and kind engineers. From the bottom of my heart, my heartfelt appreciation, gratitude, and salutations to these lovely folks.

9.2 Practicum and Its Value

The primary purpose of this report is to present a method that allows a project to be completed more efficiently and promptly with correct direction from both the banks and the loan Seeker. Using this method, any loan seeker can easily get their preferable loan according to the criteria. As a result, the website will be more user-friendly and efficient.

Undergraduate engineering courses last four years and provide students with theoretical and practical skills. The practicum program clarifies those topics' concerns to another degree by using that knowledge and watching an actual operating system. Given this, it brings us great pleasure to report that our practicum was a huge success.

There are no additional options for practical job experience. Before starting a career, students should obtain real-life work experience in their primary subject of study. They place a significant value on an applicant's professional experience. Students with more work experience are given more job options.

Working at Longbitz Software Solutions provides me with an opportunity to work in a professional setting. Throughout the internship, I did my best to make my system as efficient as possible. I adhered to the lessons, methods, tools, and techniques that I had learned during my time at IUBAT. Successful software development is a combination of best practices, theoretical understanding, and the developer's creativity.

9.3 Future Plan

This project is still in its early stages. So, we will add additional features to this system, and we will maintain and support it based on the needs of the users. Some other characteristics are listed

below.

- The system will send an email to both the bank and the loan seeker.
- Google/Facebook sign-up will be added.

9.4 Conclusion

Working at Longbitz Software Solutions has given me the most expertise in terms of developing and implementing software. Our most exciting moment revolved around the design issue; I gained a lot of new things that were completely unseen to us. The purpose of the planning was to construct a framework that allows the loan seekers to apply the bank for loan after fulfill all the criteria and they can apply for the loan after fulfill their preferable criteria within a certain time period at the start of the software project and should be updated on a regular basis as the study progresses. I worked hard to make it completely functional in this limited amount of time. As my understanding of the code expands, I will strive to improve it in every manner imaginable. I hope that this software project benefits its donors and motivates programmers to implement more advanced automated Loan Management Systems and associated problem resolution. I thank our distinguished academics for their assistance in achieving the project's objectives by providing helpful advice and guidelines. I feel I will be able to apply this experience to our professional prospects.

References

Meyer and Bertrand. (1997), OOSC2: The Use Case Principle.
Available at: http://www.elj.com/elj/v1/n2/bm/usecases.
(Accessed on 25 October, 2018)

Anon., 2015. Database Management System. [Online]
Available at: https://www.tutorialspoint.com/dbms/dbms_tutorial.pdf
[Ac cessed on 30 June 2018].

Longstreet, David. (2005). Fundamentals of Function Point Analysis.
Available at:: http://www.softwfaremetrics.com/fpafund.html
[Accessed on 20 November, 2018]

S.Kushal, 2007. *Calculating Function Points.* [Online]
Available at: https://www.codeproject.com/Articles/18024/Calculating-Function-Points
[Accessed 20 November 2018].