Membership Drive And Balloting System

By

Muhammad Qasim

Institute Of Information Technology

Quaid-e-Azam University Islamabad

Spring 2014

Quaid-e-Azam Un versity, Islamabad

Project Title: Membership Drive and Balloting System

Developed By: Muhammad Qasim

Supervised By: Mam Bushra Alma's

Institute of Information Technology

STATEMENT OF SUBMISSION

This is to certify that ***Mr. Muhammad Qasim*** Registration No. ***01161211035*** has successfully completed the final project as ***"Membership Drive & Balloting System"*** Quaid-i-Azam University, Islamabad to fulfill the partial requirement of the degree **"Master of Sciences in Information Technology".**

External Examiner

Dr. Sharifullah Khan

NUST School of Electrical Engineering
& Computer Sciences (SEECS)
NUST SEECS Campus
Sector H-12, Islamabad.

Internal Examiner

Ms. Bushra Almas
Assistant Professor

Institute of Information Technology,
Quaid-i-Azam University, Islamabad.

A Report submitted to the
Institute of Information Technology,
Quaid-e-Azam University, Islamabad,
As a partial fulfillment of the requirements
For the award of the degree of
Master of Information Technology

Project in Brief

- **Project Title**
  Membership Drive and Balloting System

- **Undertaken by**
  Muhammad Qasim

- **Supervised by**
  Mam Bushra Alma's

- **Organization**
  Federal Government Employees Housing Foundation, Islamabad

- **Started**
  June 2013

- **Completed**
  January 2014

- **Software Tools**
  ASP.net
  C#
  JQuery
  SQL Server 2008

- **Operating System**
  Windows 7

Dedicated to

My Parents, Sisters, Brothers and my Teachers

For their endless support, affection, trust and encouragement

Abstract

The project is a web based application that automates the manual system of membership management, balloting of plots and installments submitted by the Federal Government Employees Housing Foundation. This project is developed to facilitate the working of Federal Employees Housing Foundation, Islamabad. The currently running system is manual which creates problems for the management of the organization. It is also time consuming and wastes a lot of resources and man power. Our proposed system handles the existing problems and provides the required reports related to the management of information. It also relives the admin of the system to relieve from the unauthorized access. It will save data, time. This Web-application helps the users to generate reports and search the record.

Acknowledgement

ALLAH has been the biggest source of strength for me and I am humbly thankful for His blessings. He has conferred upon me, my **parents, sisters** and **brother's** love, trust and their support has always played a major role in my achievements, I have been blessed throughout my life. I am obliged to pay my sincere and heartiest gratefulness to my supervisor or more accurately my counselor, **Mam.Bushra Almas**. Her timely guidance and useful suggestions not only helped in this software building but also in the completion of my thesis.

I am very grateful to **Mr. Asad Naeem Khan** (external supervisors) for their true guidance and motivation to fulfill requirements of this project. I would like to pay my sincere thanks to all my teachers for they taught me very informative and interesting courses that proved worthy to improve my skills.

Muhammad Qasim

Table of Contents

Chapter 1 ..... **1**

1.1. Introduction 2

1.2. Existing system 2

1.2.1. Drawbacks of Existing system 3

1.3. Project Scope 3

1.4. Proposed system 4

1.4.1. Benefits of proposed system 4

1.5. Main Module in proposed system 5

1.5.1. Membership Drive 5

1.5.2. Balloting 5

1.5.3. Payments of installments and Surcharge 5

1.5.4. User and Role Management 6

1.5.5. Reporting 6

1.6. Methodologies 6

1.6.1. Presentation tier 6

1.6.2. Application tier (business logic, logic tier, data access tier, or middle tier) 6

1.6.3. Data Tier 6

Chapter 2 ..... **8**

2.1. Introduction 9

2.2. Requirements Elicitation Techniques 9

2.3. Requirements Types 10

2.3.1. Functional requirements 10
2.3.2. Non-functional requirements 10
2.4. Functional Requirements 10
2.4.1. Use Case List 11
2.4.2. Use case Diagram 15
2.5 Non-Functional Requirements 17
2.5.1. Usability 17
2.5.2. Security 17
2.5.3. Speed 17
2.5.4. Efficiency 17
2.5.5. Reliability 17
2.5.6. Availability 18
2.6. External Requirements 18
2.6.1. User Interfaces 18
2.6.2. Operating System Requirements 18
2.6.3. Hardware Requirements 18
2.6.4. Software Development Requirements 18
2.7. Process Model 19
2.7.1. Incremental Process Model 19
Figure 2.3 Incremental Process Model. 20

Chapter 3 21

3.1. System Design 22
3.1.1. Data Flow Diagram 23

3.1.2. Class Diagram 23

3.1.3. Sequence Diagram 25

Figure 3.3 Login user 25

Figure 3.4 Register new user 26

Figure 3.5 Balloting 27

Figure 3.7 Pay Installment 28

3.1.4. ER Diagram 29

Figure 3.8 Entity Relationship Diagram 30

Table 1:Roles 31

Table 2:Users 32

Table 3:Function 32

4.1. Introduction 34

4.2. Pattern Selection 34

4.3. Programming Language Selection 35

4.3.1. ASP.NET 35

4.3.2. Why use ASP.NET 35

4.3.3. C# 36

4.3.4. HTML 37

4.3.5. JavaScript 38

4.4. Database Design 38

4.4.1. SQL 38

4.4.2. Why Choose SQL? 39

4.5. Development tools 39

4.6. Development methodologies 40

4.7. Deployment Requirements 40
4.7.1. Hardware Requirements 40
4.7.2. Software Requirements 40
Chapter 5 ..... **42**
5.1. Introduction 43
Types of Testing 43
5.2. Black box testing 43
5.3. Test Cases 44
5.3.1. Check whether user is login or not. 44
5.3.2. Check role is created 44
5.3.3. Check whether account is created or not. 45
5.3.4. Check whether users are updated to database 46
5.3.5. Check whether Balloting is done or not. 46
5.3.6. Check whether the installments are correctly paid. 47
5.3.7. Check whether Searching is done 47
5.3.8. Check whether Members are added 48
5.4. White box Testing 49
5.4.1. Unit Testing 49
5.4.2. Integration Testing 49
5.4.3. Security Testing 50
5.5. Test Plan 50
5.5.1. Testing Goals 50
5.5.2. Key Areas to be focused while testing 51

5.5.3. Functionality Testing 51
5.5.4. Performance Testing 51
5.5.5. Usability Testing 52
5.5.6. Security 52
Chapter 6 53
6.1. Login 54
6.2. Home Page 55
6.3. User Management 56
6.4. Search 57
6.5. Role Management 58
6.6. Pay Installments 59
6.7. Balloting 60
6.8. Add Payment 61
6.9. Payment Schedule of User 62
6.1. Payment Details of User 63
Chapter 7 64
7.1. Conclusion 65
7.2. Future Enhancements 65
Glossary 66
References 67

List of Tables

Table 1 Role 31
Table 2 User 32
Table 3 Function 32
Table 4 Role Function .....

Chapter 1
INTRODUCTION

1.1. Introduction

The main advantage of the internet is that it has made the information available in a quick and easy manner. Internet has converted the world into a global village and has allowed the communication around the world with no time. Web development is changing the application development scenario and helping the organization to reach their customer in cost effective manner.

Membership drive and Balloting System is a web application that is developed to automate the manual process of balloting done in FGEHF. Through this application the employees can easily check the records and see the balloting results they can find information about his installments (paid, remaining) and deadline for installments. . Administrator has access and updates all the functions of the system. He can manage the record of the Employees. Furthermore, he can manage the roles of the Employees so that only provide authority to those modules of system which are related to user.

This membership drive and balloting system can be used to keep records of different members and balloting result that are being used by the company or the department. The membership management registers the members who are interested in the balloting. After registration the data of membership is transferred to the balloting procedure which produces the balloting results randomly without any biasness. Once Balloting is done every members have to pay the installments of the Plots, For this purpose the keep track of installments of the every member. This system thus keeps record of every member automatically and administration is relived from manual work of managing membership & installments.

1.2. Existing system

The prevailing system of Federal Government Employee Housing Foundation for balloting of plots & management of memberships of Federal Employees is a manual system. Following are the drawbacks of the existing manual system that needs to be computerized.

1.2.1. Drawbacks of Existing system

- **Inaccurate**

  Keeping records of thousands of members & plots manually generally results in erroneous data entry in papers and results in inaccurate information.

- **Difficult to manage Members information**

  As the number of member's increases with time it become difficult for the employee of FGEHF to store and retrieve membership information on pages.

- **Difficult to manage balloting**

  Balloting is difficult to manage because the balloting of thousands of people can't be done using manual system.

- **Report generation is difficult**

  Generating reports of installments, previous trends of members is really difficult and time consuming due to manual record keeping.

- **Duplication**

  Duplication of data can't be checked or a lot of time is required to search for duplication.

- **Installments Calculation**

  Installments and surcharge calculation is difficult to calculate for every members

1.3. Project Scope

The scope of this project is information management related to membership and balloting, installments, user management and role management. This project will efficiently handle the membership management and balloting management and will saves the time. It will

provide security to member confidential data as it does not allow unauthorized access. It will provide the optimize solution to the problems and will help the users to search and generate reports quiet efficiently.

1.4. Proposed system

We are developing the web-based membership management and balloting system of Federal Government Employee Housing Foundation which handles the most of the problems of existing system and also provides the required reports related to the following information.

- Membership Details
- Balloting Details
- Installments Details
- Transfer Details
- Refund

1.4.1. Benefits of proposed system

As we are going to computerize the system, the new information system will provide many advantages over the existing manual system. Following are the main advantage of the proposed system.

- Minimum time consumption due to the computerize system.
- Provide security and fairness balloting system
- Provide maximum reliability.
- Provide accuracy in generating installment records.
- Manage employee information easily.
- Report generation easily. For example report of membership, reports for balloting etc.

1.5. Main Module in proposed system

- Membership drive
- Balloting
- Payment of Installments and surcharge
- User and Role Management
- Transferring Plot

1.5.1. Membership Drive

This module contains personal & official information about members of Federal Government Employees Housing Foundation which have applied for the balloting. Members submit their information through a form by logging into the system. The system verifies that a person submit his information only once; It further validates that the input data is in correct format/not.

1.5.2. Balloting

This module contains information about the balloting procedure and results. The data from membership drive is used for balloting. The balloting procedure randomly selects the members from database according to its category and quota. There are categories(1,2,3,4,5) and quota(Autonomous, FG services , Constitution , Retired/ Constitution, Widow, Retired / Autonomous, Retired). The system also check that the name of the person does not appear every time by selecting random person from database

1.5.3. Payments of installments and Surcharge

After installments every person has to submit its installments. Every member has to submit 12 installments excluding deposit with application and possession amount. Every Installment need to be submits after the period of three months. If the member can't submit its installments on the desire date then the FGEHF allow the one month of additional time for submission of the installment with original amount. But after the one month the member is surcharge with the

penalty as per days @ rate of 2% of the total amount. Complete information of installments and submission date is handled through this system.

1.5.4. User and Role Management

This module will manage the complete and correct user and role management. In this application admin has the rights to create the roles and users. Every user has assigned some roles for the use of application. Admin has the rights to select which user has assigned to which task or role. Thus security is implemented in this way to avoid unauthorized access to the application.

1.5.5. Reporting

This module will generate detailed reports that are frequently needed.

1.6. Methodologies

The three tiered architecture is a client/server architecture in which we logically separate presentation, application and data processes. This tiered architecture is most widely used among the multi-tiered architecture. The tiered architecture is a physical structuring of system infrastructure while layered architecture is logical structure of system infrastructure. Three-tier architecture has the following three tiers:

1.6.1. Presentation tier

- Presentation tier: what the users see, typically a web application.

1.6.2. Application tier (business logic, logic tier, data access tier, or middle tier)

Business tier: where all the logic for your application is performed.

1.6.3. Data Tier

. Here information is stored and retrieved. This layer is independent from presentation layer and application layer. Data tier consists of database servers in which different table have relationship between them.

Figure 1.1 shows an example of the three-tiered architecture.

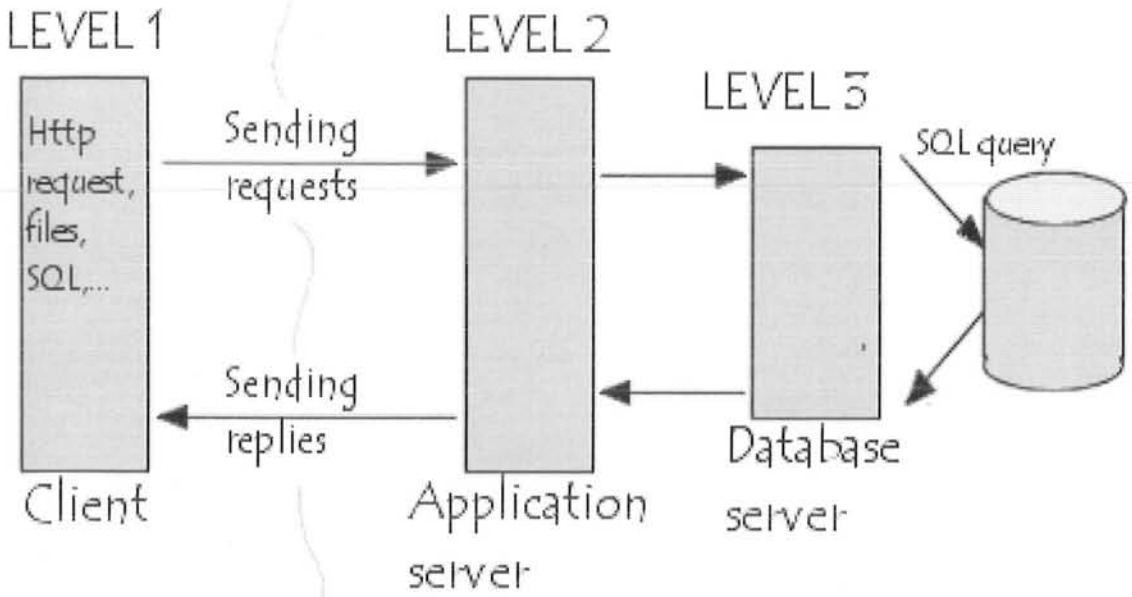


Figure 1.1: Three Tier Architecture

Chapter 2
Requirement Analysis

2.1. Introduction

In software development life cycle requirement analysis is first step. Requirements are specifications of the system or functionality required in the system by the user. Requirements analysis in software engineering, consist of those tasks that go into determining the needs or conditions to meet for a new or altered product, taking into account of the every possible conflicting requirements of the various stakeholders, analyzing, documenting, validating and managing software or system requirements.

Our Balloting system is designed for a organization that have a complete information about balloting and membership. The system is easy to use. Membership drive and balloting system may be used to keep records of members that are being used by the different department, all the information about the members and the balloting results and their installments.

2.2. Requirements Elicitation Techniques

There are so many techniques that are used for the requirement elicitation. Some of them are as under [1]:

- Interview
- Surveys
- Questioners
- Meetings
- Joint application development (JAD)

In software engineering these requirement techniques are so beneficial for the gathering of information from the stakeholders. These techniques are used according to the scope of project and availability of resources. I have used two techniques in gathering of requirements which are meeting and interviews. Throughout the project I have conducted meeting with organization personal and also conducted meeting with those who deal and managing membership and balloting system.

2.3. Requirements Types

There are two major types of software requirements:-

- Functional Requirements
- Non-Functional Requirements

2.3.1. Functional requirements

The requirements that totally relates to functionality of the system are functional requirements. The behavior of system is represented by these requirements [3].

2.3.2. Non-functional requirements

These requirements are not relates to the functionality of system but they are also so much important for the system. These requirements related to those elements that represent the system performance and security.

2.4. Functional Requirements

Our system has the following functional requirements:

- Login
- Role management
- User management
- Balloting
- Add members' details.
- Edit member's detail.
- Searching of members.
- Search balloting result.
- Pay installment.
- View installments information by user
- Change password.

- Transfer Plot
- Surcharge Calculation
- Refund

2.4.1. Use Case List

2.4.1.1. Log in use case

Name of use case	Log In
Description	For use of the system user have to logged in to the system
Precondition	The administrator must provide user name and password to use the system.
Post Condition	User successfully logged in to the system.
Normal Flow	User starts the application User enters the username and password and press login button. The system successfully logged in.
Alternate Flow	Prompt the wrong user name and password. Display wrong user name or password

2.4.1.2. User management use case

Name of use case	User Management
Description	The use case is used when the administrator start the application.
Precondition	The administrator must be logged in to update the system.
Post Condition	Administrator successfully manages users.
Normal Flow	Administrator successfully logged into the system. Mouse over the Admin, list will appear. Administrator click on the user management.

	Administrator enters the users. Click on save.
Alternate Flow	Prompt the wrong users. Or the user that are stored first in records.

2.4.1.3. Role Management use case

Name of use case	**Role Management**
Description	The use case is used when the administrator starts the application.
Precondition	The administrators have to log in and then can assign the roles to the specific user.
Post Condition	Administrator successfully assigns the roles.
Normal Flow	Administrator successfully logged into the system. Mouse over the Admin, list will appear. Administrator click on the role management. Administrator enters the role. Click on save.
Alternate Flow	The Administrator doesn't add roles and use the defined roles for creating users.

2.4.1.4. Balloting use case

Name of use case	**Balloting**
Description	The use case is used when the user start the application.
Precondition	User must log in and he has a right for balloting.
Post Condition	User has successfully ballot for the members.
Normal Flow	User logged in the system. Click over the ballot tab.

	Click ok
Alternate Flow	If the user has not assign role of balloting then he can't ballot.

2.4.1.5. Add Members use case

Name of use case	**Add Members**
Description	The use case is used when user want to apply for membership.
Precondition	The user must be logged in and can view the memebrship form.
Post Condition	Membership details successfully added.
Normal Flow	User starts the application (open the web page). User can do whatever is permitted by the administrator. Click on the membership tab. Click on add new members. User fills the information in specified fields. Members are successfully added.
Alternate Flow	The user does not have the user name and password to get into system.

2.4.1.6. Installments use case

Name of use case	**Instalments**
Description	The use case is used when those members who have the name in balloting can view the information about instalments
Precondition	The Members must be logged in.
Post Condition	Members successfully view the instalments.

Normal Flow	Members start the application (open the web page). Members can manage the instalments if they have permission by the administrator. The instalments successfully paid/view.
Alternate Flow	The Members has not access to the system and can't view instalments.

2.4.1.7. Searching use case

Name of use case	**Searching membership/balloting results**
Description	The use case is used when those member start the application which are permitted by the administrator.
Precondition	The members must be logged in and search the members details/ balloting results.
Post Condition	Members successfully search the information.
Normal Flow	Member starts the application (open the web page). Member can do searching related to membership or balloting management if they have permission by the administrator. The system successfully search the data.
Alternate Flow	If the details are not available in database then it will show not found error.

2.4.2. Use case Diagram

The interaction of user with the system can be represented as a use case diagram. A use-case diagram can help provide a higher-level view of the system. They provide the simplified and graphical representation of what the system must actually do. The main use case diagram of our system is shown in Figure 2.1 & 2.2.

Use Case Diagram

Figure 2.1 Use Case Diagrams (End User)

Use Case Diagram

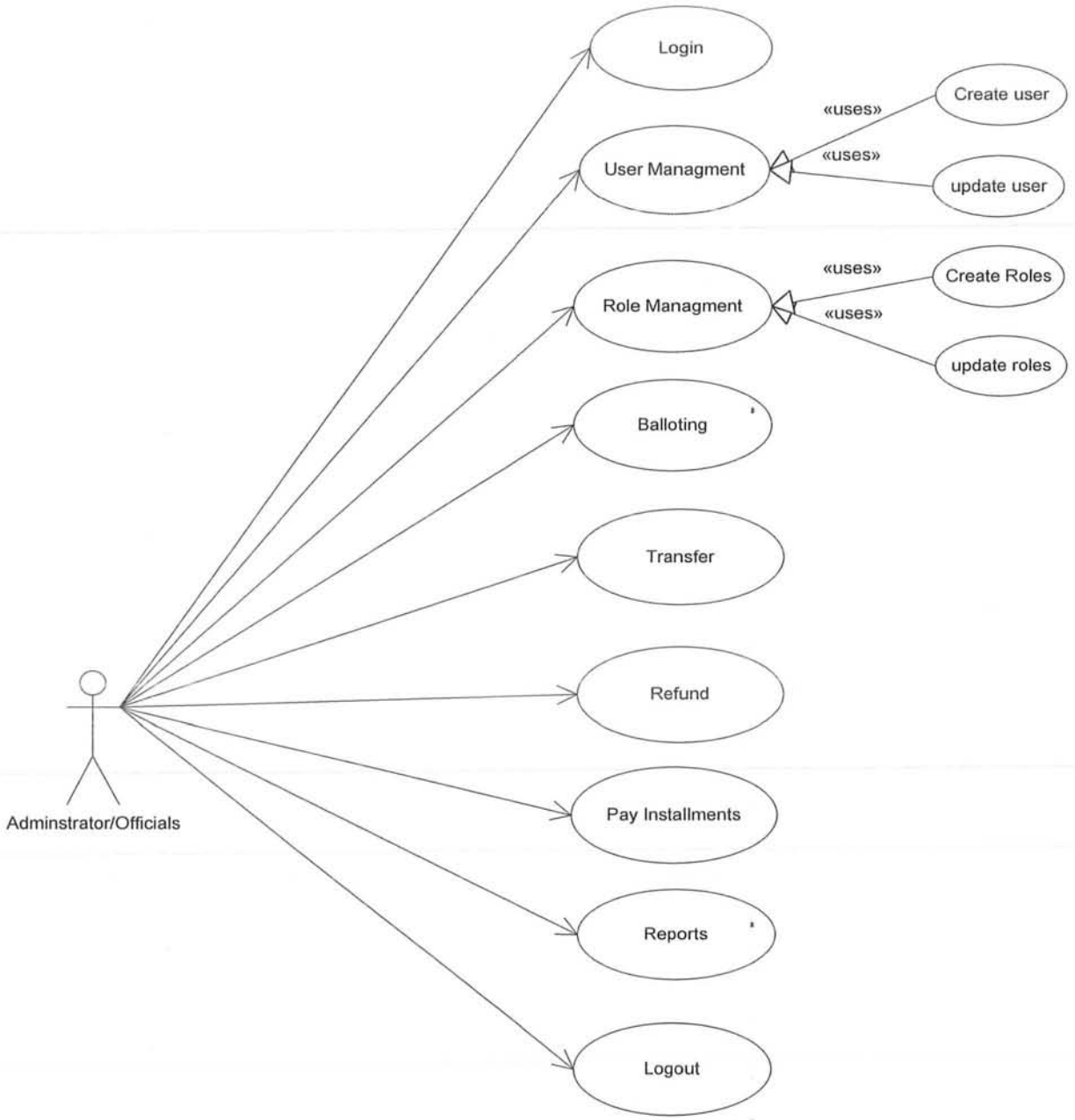


Figure 2.2 Use Case Diagrams (Administrator)

2.5 Non-Functional Requirements

As for as non-functional requirements are concerned it is also mandatory part of any software system. With the complex development of the software product we can't ignore the non-functional requirements. Some of the non-functional requirements are stated below:

2.5.1. Usability

Usability is the ease of use and learning of a human-made object. The object of use can be a software application, website etc. The system is usable in such a way that the interface is simplified so everyone can understand the system.

2.5.2. Security

For security the system should not allow un-authorized access. To confirm the identity of the person the system authenticates a user. The software can authenticate users who can access the system. Security is the degree of protection against danger, damage, loss, and crime. Security is implemented in a way that every user has allotted a password and that password is encrypted in database so if a person can get into the database then in that case he cannot get the password.

2.5.3. Speed

The speed of the system is its processing speed. It can be measured by the throughput of the system.

2.5.4. Efficiency

The user data shall be stored that is still accessible even after the time period is complete or expired. The system should be much more efficient for checking the information about user. System should be efficient and self-explanatory. It means easy to understand and easy to use.

2.5.5. Reliability

Reliability is the ability of software work under; given environmental conditions, for a particular amount of time System handles exceptions. Proper data handling is made if any user want to insert wrong data then the system will show error to the user. If there is any

exception or error, system must handle this problem or show proper message and close the application.

2.5.6. Availability

Availability is defined as the probability that whether the system is available to user when it required. System will be available to all users 24 hours and 7 days.

2.6. External Requirements

2.6.1. User Interfaces

A user interface is the system by which people interact with software. The MainPage is the simple and fully designed page, which is easily accessible by all the users. Sitemap is the function in which all contents of the system are included and all contents of the website are accessible.

2.6.2. Operating System Requirements

Windows 7 operating system is used to fulfill software interface requirement.

2.6.3. Hardware Requirements

The system on which this application will be developed must have at least

- Core 2 Duo 2.4 GHz processor
- 2 GB Ram
- 120 GB Hard Disk

2.6.4. Software Development Requirements

- Visual studio 2008 will be used as development tool.
- C# is used as developing language.
- Crystal Reports.
- SQL server 2008.

- Object oriented approach.

2.7. Process Model

Process model describes the steps that are involved in the development of software project.

2.7.1. Incremental Process Model

The Incremental Process is combination of both iterative method and incremental build model for development of software product. The incremental model is suitable for the long term project. The relationship between iterations and increments is determined by the overall software development methodology and software development process. The exact number and nature of the particular incremental builds and what is iterated will be specific to each individual development effort.

Advantages

- The early versions of working software are quickly generated and implemented.
- More flexible and less costly to evolve the scope and requirements.
- Due to the smaller iteration testing and debugging is easier.
- Risk can be easily managed due to iteration.

Diagram:

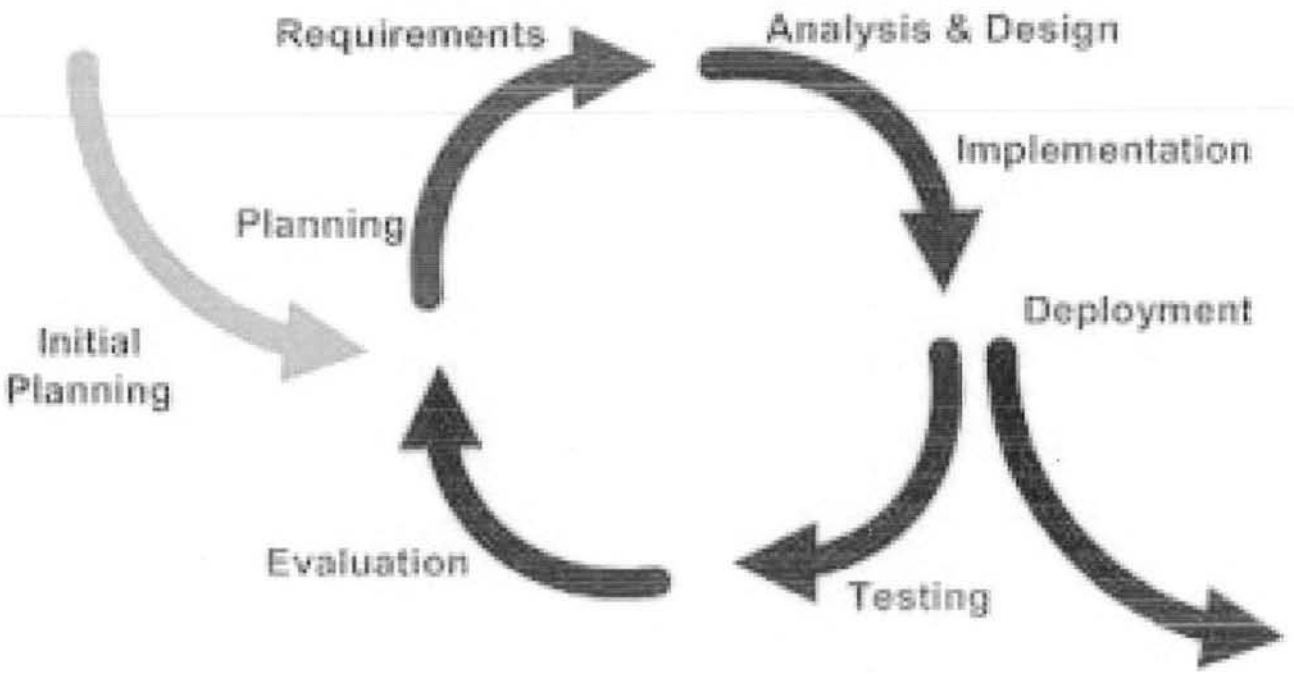


Figure 2.3 Incremental Process Model.

Chapter 3
System Design

3.1. System Design

The Software design that show interaction of the user to the system. It is the necessary part for system. After carefully analyzing the requirements and the functionality of the system, I had to analyze the system design. System design is related to how the system works. System design is used to determine the relationship between components and identify the component dependencies. It is the process of solving problems related to the system and planning for a software solution.

Some diagrams are included here to describe the flow of the system:

- Data Flow Diagram
- Class diagram
- Activity diagram
- Entity Relationship Diagram

3.1.1. Data Flow Diagram

Data Flow Diagram (DFD) is used to describe the flow of data. How the system is getting input from user and passing it from different modules and writing an output. Figure 3.1 shows DFD of our system.

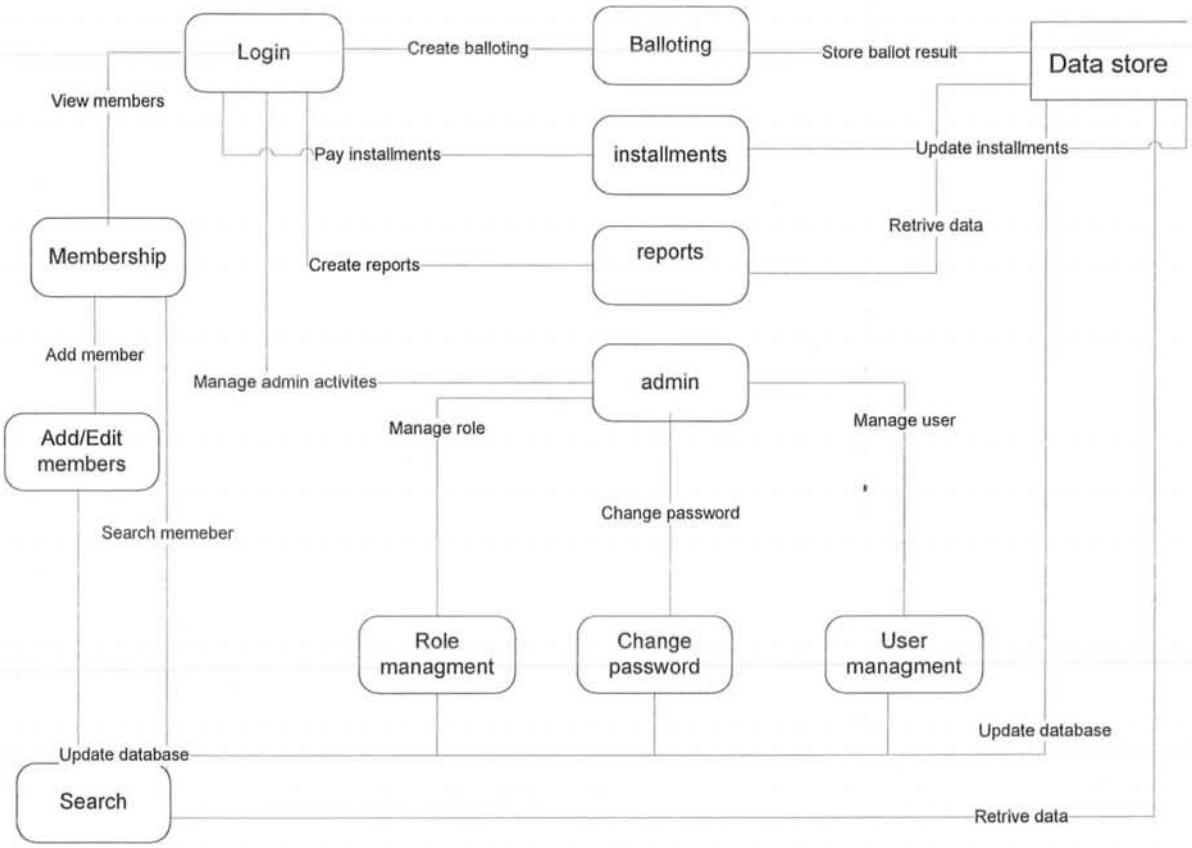


Figure 3.1 Data Flow Diagram

3.1.2. Class Diagram

Class diagram describes the structure of the system by showing the system classes. A class describes the properties and behavior of an object. A class diagram describes the static view in terms of classes and relationship among the classes.

The class diagram of our system with the relationship between the objects is shown in Figure 3.2.

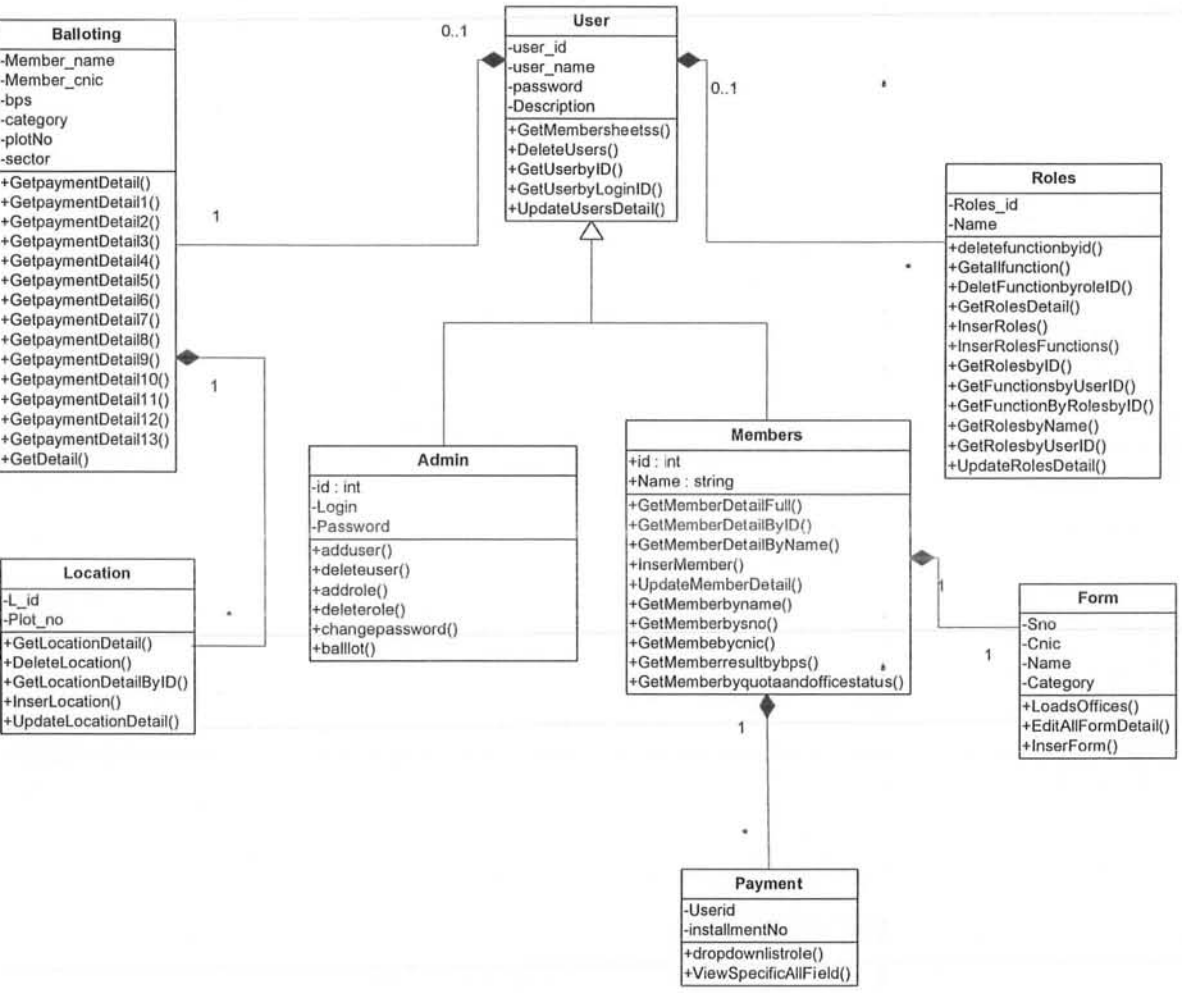


Figure 3.2 Class diagram

3.1.3. Sequence Diagram

Sequence Diagram is a kind of interaction diagram. It shows how processes interact with each other and in which sequence. It depicts the objects and classes involved and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario.

Login User

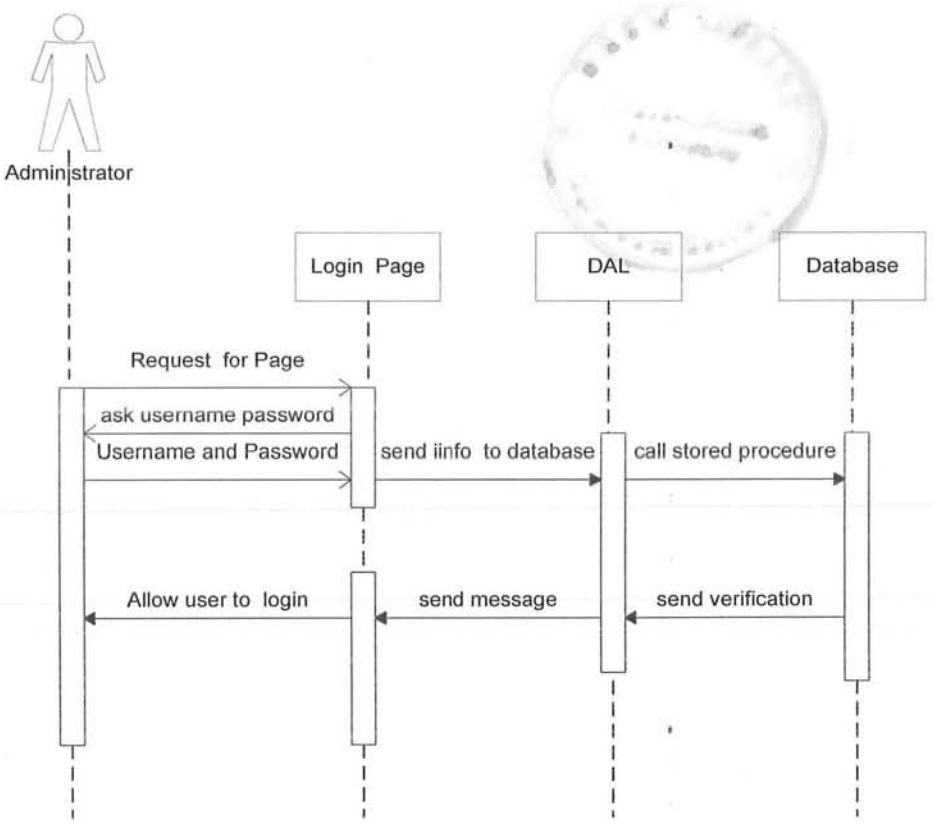


Figure 3.3 Login user

Register New User

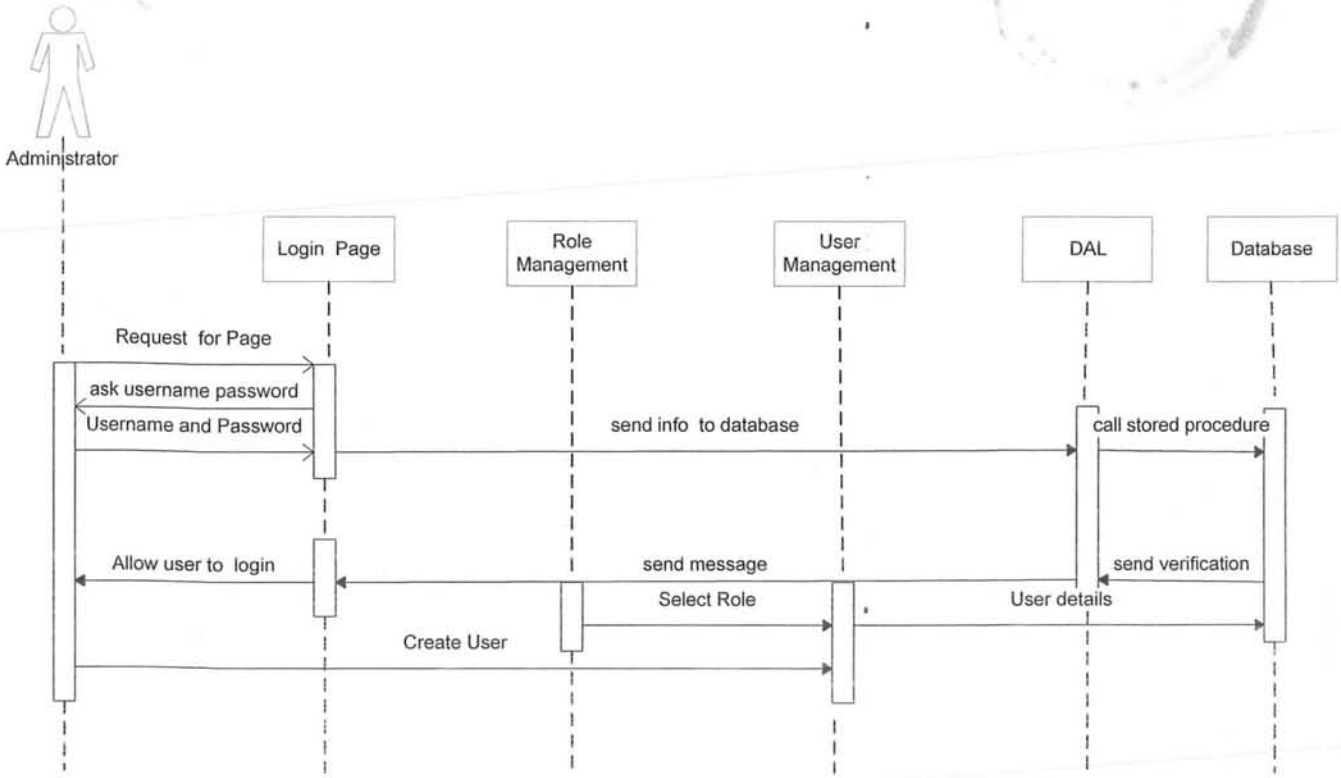


Figure 3.4 Register new user

Balloting

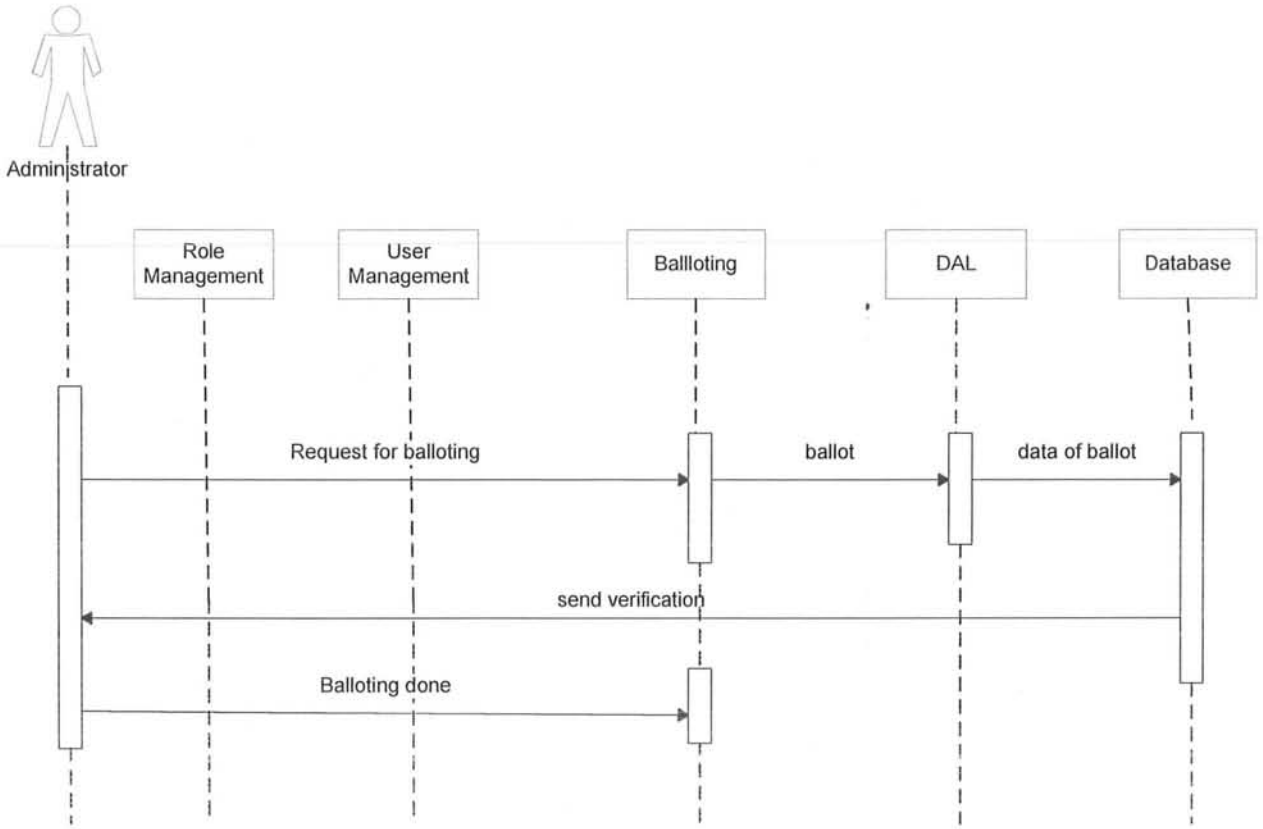


Figure 3.5 Balloting

Pay Installments

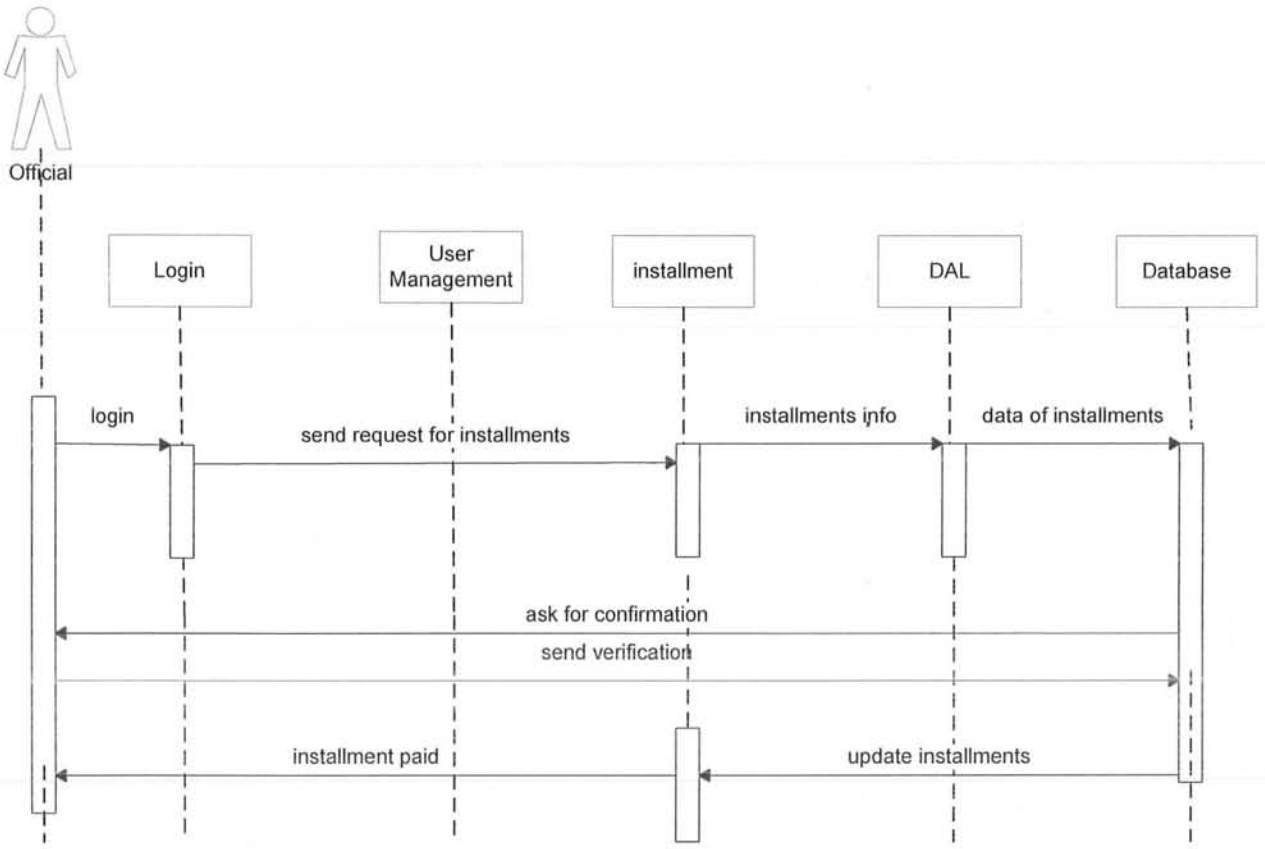
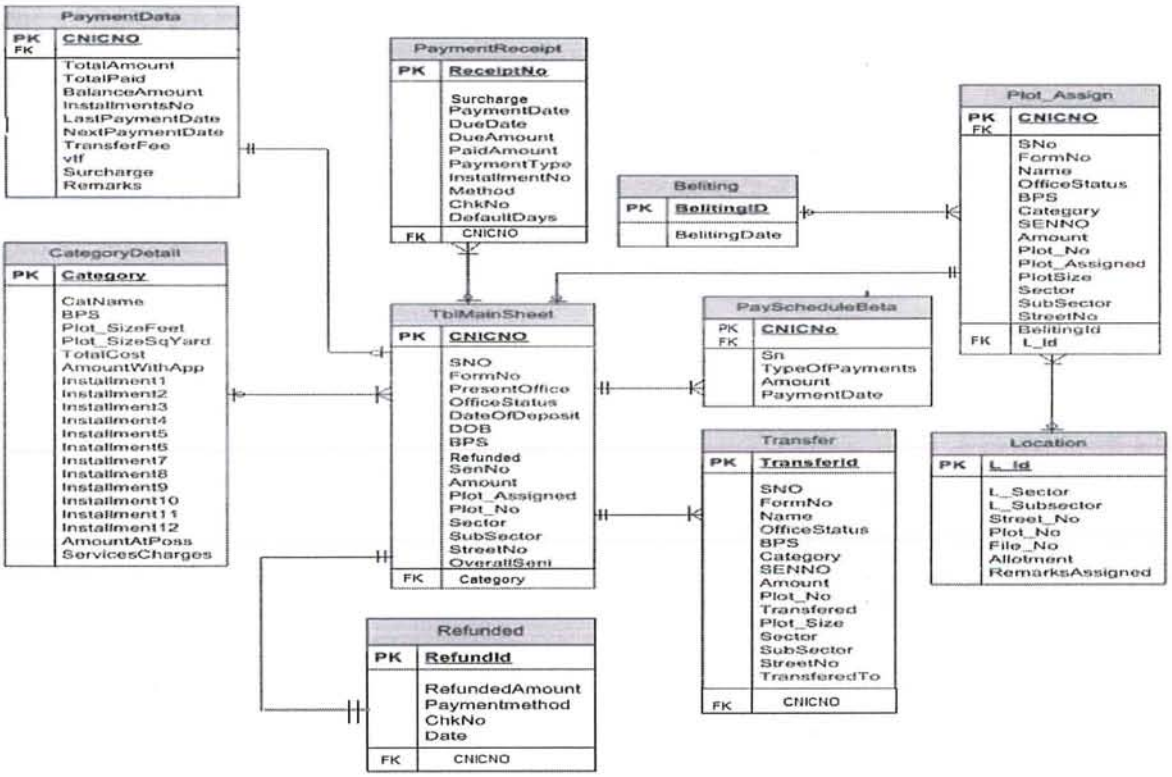


Figure 3.7 Pay Installment

3.1.4. ER Diagram

The Entity Relationship Model (ER Model) is a detailed and the logical representation of the data. The entity relationship model is described in terms of the entities, the relationship among these entities and the attributes of the entities. To express the ER model, normally we use the Entity Relationship Diagram (ERD), which is the graphical representation of the ER model.



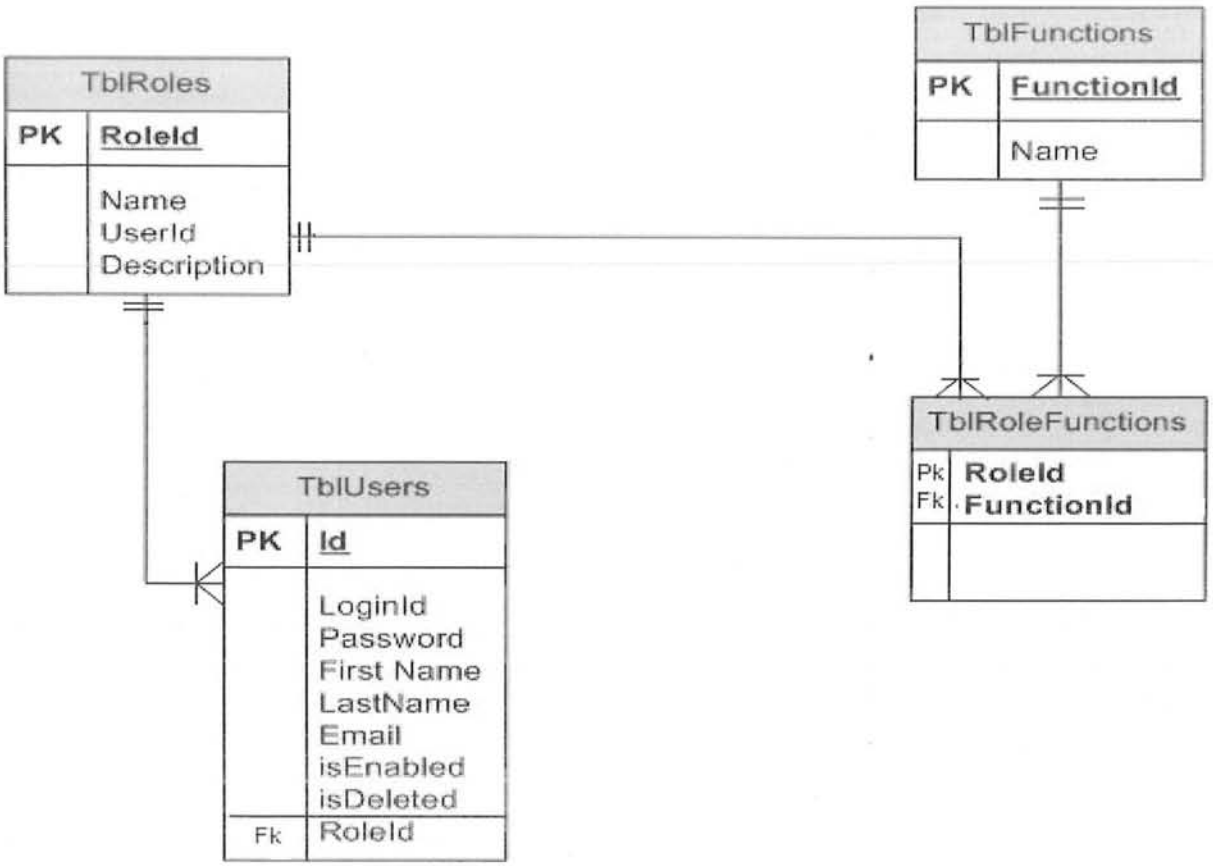


Figure 3.8 Entity Relationship Diagram

Explanation: In Membership Drive and Balloting System every user is assigned some roles and function to perform. The table role contains information about the roles. The user performs those roles which are assigned to it in tblrole. The user can fill form and the data of user is saved in Tblmainsheet table. Balloting is done by using the data from the Tblmainsheet and the location table for picking up the plotno. After the balloting the results are inserted into the table Plotassign

3.1.4.1. Rational Schema

Relation Name: **Roles**

Primary Key **role_id**

Purpose: **Stores the roles assign to each user**

Field Name	Type	Width	Constraint	Description
role_id	Int	-	Not Null	role identity
Name	Varchar	50		Name of the category
User_id	Int			User idenity
Description	Varchar			Description about user

Table 1:Roles

Relation Name: **User**

Primary Key **user_id**

Purpose: **Stores the information about each user**

Field Name	Type	Width	Constraint	Description
User_id	int	-	Not Null	Identity of user
loginId	Varchar	-	-	Id for the user
password	Varchar	-	-	Password for user
FirstName	Varchar	50	-	Name of user
LatName	Varchar	50	-	

Email	Varchar	50	-	
Role_id	int	-	Not Null	Foreign key for relationship from role table

Table 2:Users

Relation Name: **Function**

Primary Key **F_id**

Purpose: **Stores the data about each function**

Field Name	Type	Width	Constraint	Description
F_id	Int	-	Not Null	Identify for function
Name	Varchar	50		Stores the name of function

Table 3:Function

Relation Name: **RoleFunction**

Primary Key **Role_id**

Purpose: **Store the information of roles and functions**

Field Name	Type	Width	Constraint	Description
Role_id	Int	-	Not Null	Store the identifier for Role
F_id	Int	-	Not Null	Stores the Identifier of function

Chapter 4

System Implementation

4.1. Introduction

After the design phase, the development phase starts. In Implementation phase the system design is converted into an executable form. Execution of a plan, realization of an application, idea, model, design, specification, standard, algorithm, or policy is Implementation. It describes the various functionalities step by step under each module with their outputs. This includes the performance of hardware devices, software's utilities or tools that aid in development and the problems faced during their installation. The goal of the implementation is to implement a system correctly, efficiently, and quickly using particular tools and programming languages. Implementation activities are primarily environmental. They deal with realities of particular machines, systems, languages, tools, developers and the clients to translate a design into working code. This chapter explains all the steps that are necessary for the system development.

4.2. Pattern Selection

This system uses the Model/View/Controller (MVC) pattern. Model–View–Controller (MVC) is a software design for interactive system that separates the representation of information from the user's interaction with it. The *Model* consists of application data and business rules. The Controller controls the input, and converting it to commands for the model or view. A view gives the output of data such as a chart or a diagram. Multiple views of the same data are possible such as a pie chart. In addition to dividing the application into three kinds of component, the MVC design defines the interactions between them.

- A controller can send commands to its associated view to change the view's presentation of the model. It can send commands to the model to update the model's state.
- When there has been a change in states, a model notifies its associated views and controllers. Due to this notification, views produce updated output and allow the controllers to change the available set of commands. A passive implementation of MVC omits these notifications, because the application does not require them or the software platform does not support them.
- View asks for information that is required as output.

4.3. Programming Language Selection

4.3.1. ASP.NET

It is a Microsoft technology and is the next version of Active Server Pages (ASP) [6]. It is the server side scripting technology that enables scripts to be executed by an internet server. ASP.NET allow the program to run in ISS (Internet Information Services). ASP.NET was developed to solve the problem that was being faces by the user with classic ASP. While the ASP.NET is largely syntax compatible with ASP, it also provides the new programming model and infrastructure for more scalable and stable applications that help provide greater protection.

HTML editors and other programming tools, including the Microsoft Visual Studio ASP.NET is designed. JAVA Scripting can also use. It is not only beneficial for the development of web application, but it also provides all benefits that these tools have to offer, including the GUI that developers can use to drop server controls onto a Web page and fully integrated debugging support. ASP.NET is a technology for building powerful, dynamic Web applications and is part of the .NET Framework. .NET is the language independent, which means you can use any .NET-supported language to make .NET applications.

One of the main differences between ASP.NET and Classic ASP/PHP is the fact that ASP.NET can be compiled, while Classic ASP is always interpreted. The PHP can be compiled by using the commercial products, but usually it is interpreted as well. In ASP.NET Web Applications are built using Web Forms. ASP.NET comes with built-in Web Forms controls, which are responsible for generating user interface. They mirror HTML widgets like text boxes or buttons. If these controls do not fit your needs, you are free to create your own user controls.

4.3.2. Why use ASP.NET

There are certain reasons due to which that I have used the ASP.NET as development environment. ASP.NET is not limited to the scripting languages; it allows us to make use of .NET languages like C# etc. It is built on the common language runtime that can be used on any

windows server to host powerful ASP.NET websites and technologies. It provides the very easy to use code. Following are the advantages of using ASP.NET.

- ASP.NET pages are easy to write and maintain because the source code and HTML are together.
- There is no need to write the large code for making the tags. We just have to drag and drop on the page.
- ASP.NET provides better performance by taking advantage of early binding, just-in-time compilation, native optimization, and caching services right out of the box.
- The source code is executed on the server. This provides a lot of power and flexibility to the Web pages.
- It easily works with the ADO.NET using data binding and page formatting features. It is an application, which runs faster and counters large volumes of users without having performance problems.
- It offers rich controls that are easy to use.

4.3.3. C#

C# is used in the ASP.NET for server side scripting. All the code that we want to run over the server is written in C#. The C# is the programming language that is designed to work with Microsoft's .NET platform. It is an object-oriented programming language that enables the developers to build a variety of secure and fast applications that are run on the .NET Framework. We can use the C# to create traditional windows client applications, client-server applications, database application and much more. C# is intended to a simple, modern, general-purpose, object-oriented programming language.

It is an international standard programming language used to create instructions that direct the computer about what to do, when to do it, and how to do. The C# is designed to be a platform-independent language.

C# is designed to take the advantage of design of .NET. Microsoft C# is the new programming language that is designed to build a wide range of enterprise application that run on the .NET Framework. By using it, C# developers can develop Web programs and Web services easily than other Microsoft tools such as Visual C++ and Visual Basic.

The design goals of C# programming language are given below:

- C# can make our code more stable and productive.
- C# is a simple, general-purpose and object-oriented programming language.
- It is intended for the use in developing software components suitable for deployment in distributed environments.
- It provides the support for software engineering principles such as strong type checking, array bounds checking, detection of attempts to use uninitialized variables, and automatic garbage collection.
- Portability of C# is very important, especially for those programmers who are already familiar with C and C++.
- It uses unified type system and a simplified way in which value and reference types are used by the language.

4.3.4. HTML

HTML is markup language. It is not a programming language. In addition, the markup language is the set of markup tags. It is time to take Web-designing skills to the next level with Cascading Style Sheets (CSS). The use of HTML in this project is to represent the UI (User Interface). They are the way to control the look and feel of your HTML documents in an organized and efficient manner. The CSS defines the color and sizes in "styles". If we change a certain style it will change the look of the entire site. The advantage of CSS is that it offers much more detailed attributes than plain HTML for defining the look and feel of the website. By using the CSS, we can make our system more attractive and user friendly.

- CSS adds the new looks to HTML it is used to add the styles to the pages.

- We can completely re-style a Web site with only a few changes to CSS code.

4.3.5. JavaScript

JavaScript is the scripting language and was designed to add interactivity to HTML pages. JavaScript is usually embedded into the HTML pages directly. A JavaScript is the lightweight programming language. JavaScript gives the HTML designers programming tool but the JavaScript is a scripting language with a very simple syntax. A JavaScript can be set to execute when something happens, like when a page has finished the loading or when the user clicks on the HTML element. A JavaScript can read or change the content of the HTML element.

A JavaScript can be used to store and retrieve information on the visitor's computer. In the JavaScript, there are no classes, so we have no template for the object creation. JavaScript objects are entirely different from the C#. Java and the JavaScript are the completely different languages in both the concept and design. Java is the powerful and much more complex programming language in the same category as C and C++.

4.4. Database Design

4.4.1. SQL

SQL is the reliable and high performance relational database management system. SQL uses the structured Query Language (SQL). SQL is the most popular language for adding, accessing and managing content in a database. SQL is very fast, reliable and flexible Database Management System. SQL database has become the world's most popular database, because it is easy to use and flexible for creating the database. SQL is used for the Internet applications as it provides good speed and is very secure. SQL can be used for the variety of applications but it is mostly used for Web applications on the internet.

A database is the structured collection of data. To add, access and process data stored in a computer database, you need a database management system such as SQL Server. Since

computers are very good at handling large amount of data, database management systems play a central role in computing as the standalone utilities, or as the parts of other application.

4.4.2. Why Choose SQL?

SQL is the reliable and high performance relational database management system. We can use it to store many GB's of data into database. SQL has the many capabilities to handle most corporate database application. SQL supports more than twenty different platforms including major Linux distribution, UNIX or Microsoft Windows etc.

4.5. Development tools

Developments tools used to implement our system are described here.

1. **Microsoft Visual Studio 2010**

   I haves used the visual studio [7] for the development of the Website using Web pages. The Web pages are designed by using the intrinsic control of the ASP.net. I have used the Microsoft crystal reporting tool for report generation.

2. **Microsoft SQL server 2008 Management studio**

   The database that I have created is in SQL server Management Studio 2008 because it is easy to use and add records to database and easy to connect it with the ASP.net.

3. **Microsoft word 2010**

   The Microsoft Word is used in my project for the documentation of the whole project.

4. **Microsoft Visio**

   I used the Microsoft Visio for creating different types of the diagrams for example class diagram, dataflow diagramed and Sequence diagram.

5. **Crystal report**

   It is used for creating reports of data using databases.

6. **Platform**

   I have used the Microsoft .Net framework and Windows 7 platform for the system development.

4.6. Development methodologies

For the development of this system, incremental model is used which develops whole project in steps. Firstly I developed the database and the login mechanism. After creating this, I have developed the role management, user management and inventory subsystems steps by step, so my project has reached to destination in a step by step process.

4.7. Deployment Requirements

Hardware and software required to deploy this system are described here.

4.7.1. Hardware Requirements

The hardware includes the following things:

- Computer having the internet facility.
- Printer for creating reports as hard copy.
- RAM 2GB
- Hard Disk 120GB

4.7.2. Software Requirements

Following software are required:

- Internet browser
- Office/PDF reader for report reader
- Windows 7

- Dot Net Platform
- Visual Studio
- SQL Server Database
- C# Language
- JQuery

Chapter 5
System Testing

5.1. Introduction

Software testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test. Software testing can also provide an objective, independent view of the software. Test techniques include, but are not limited to the process of executing a program or application with the intent of finding software bugs (errors or other defects).

1) Find errors and bugs
2) Validate that the goal of system is achieved or not
3) Verify the system is working as user requirement

The validation determines that wether the system is completed according to user needs and specification verifies that the system is correctly working or not. Both validation and specification are conducted to show that requirement of user satisfied or not.

Types of Testing

- Black Box testing
- White Box testing

5.2. Black box testing

Black box testing involves the checking of functions of the system without the internal working or procedure. Tester doesn't even examine code. He/she just insert different valid/invalid inputs to show how the system response on specific input/outputs

➢ **Advantages of black box testing**

1) Testing is balanced and unprejudiced because tester and developer are independent of each other.
2) Test is done according to the user/tester requirement
3) Testing helps to identify vagueness and contradictions in functional specifications.

4) Test can be conducted by non-technical person or in other words test is also conducted by the people who have no knowledge of programming.

5.3. Test Cases

5.3.1. Check whether user is login or not.

Test Case ID	**TC-001**
Tester	Muhammad Qasim
Test Type	Manual-Black Box
Test Case Name	Login
Description	Check whether user is login or not.
Procedure	Open login page. Insert the username or password. Click the submit button.
Expected Result	Login successful
Actual Result	Login successful
Status	Success

5.3.2. Check role is created.

Test Case ID	**TC-002**
Tester	Muhammad Qasim
Test Type	Manual-Black Box
Test Case Name	Create role

Description	Check whether role is created or not.
Procedure	Click on admin tab. Click on Role Management. Enter details for role creation. Select role. Click on save.
Expected Result	Role created successfully.
Actual Result	Role created successfully.
Status	Success

5.3.3. Check whether account is created or not.

Test Case ID	**Tc-003**
Tester	Muhammad Qasim
Test Type	Manual Black Box
Test Case Name	Account creation
Description	User wants to create account for using website.
Procedure	Only admin can create new user. Login default page admin tab is available. Click on admin tab, two tabs will appear user management and role management. Click on user management tab, after that click on link "Add new user". Submit username and password.
Expected Result	The account will be created.

Actual Result	Account is created.
Status	Successful

5.3.4. Check whether users are updated to database.

Test Case ID	**TC-004**	
Tester	Muhammad Qasim	
Test Type	Manual-Black Box	
Test Case Name	Add\Delete user	
Description	Existing users can be deleted or not	
Procedure	Login. Click on admin tab. Select user management. Only admin can add new user and can delete existing users. For adding click on Add new user, for deleting click on delete user link.	
Test Data	Scenario 1: Select Add New user:	Scenario 2: Delete User.
Expected Result	The user is added\deleted in the database.	
Actual Result	User is added\delete.	
Status	Success	

5.3.5. Check whether Balloting is done or not.

Test Case ID	**TC-005**
Tester	Muhammad Qasim
Test Type	Manual-Black Box
Test Case Name	Select balloting

Description	Purpose of this test is whether the balloting is correctly running or not
Procedure	Click the balloting tab
Expected Result	Balloting will be performed
Actual Result	Balloting done successfully.
Status	Success

5.3.6. Check whether the installments are correctly paid.

Test Case ID	**TC-006**
Tester	Muhammad Qasim
Test Type	Manual-Black Box
Test Case Name	Pay instalment
Description	Check whether instalments are correctly paid or not.
Procedure	After user login, enter the payment tab Enter allotter payment Select or search the name for instalment payment. Click pay link Click the pay button
Expected Result	The pending instalment will paid.
Actual Result	Instalment paid
Status	Success

5.3.7. Check whether Searching is done.

Test Case ID	**TC-007**

Tester	Muhammad Qasim
Test Type	Manual-Black Box
Test Case Name	Searching
Description	Purpose is to check whether searching is done correctly
Procedure	Select Membership tab and then select searching tab. After that select on which basis you want to search (name, cnic, category &quota etc). After that enter the input. Press search button
Expected Result	Searching will performed if data is available else no data will available
Actual Result	Searching done successfully.
Status	Success

5.3.8. Check whether Members are added.

Test Case ID	**TC-008**
Tester	Muhammad Qasim
Test Type	Manual-Black Box
Test Case Name	Member creation
Description	Purpose of this test is to test whether new member are added.
Procedure	Select membership Select add new member Insert required information in the form Click save
Test Data	Add member
Expected Result	Member will be added

Actual Result	Member added.
Status	Success

5.4. White box Testing

White-box testing (also known as clear box testing, glass box testing, transparent box testing, and structural testing) is a method of testing software that tests internal structures or workings of an application, as opposed to its functionality (i.e. black-box testing).

Following are the testing levels

5.4.1. Unit Testing

Unit testing is a method by which individual units of source code, sets of one or more computer program procedure together with control data and user procedures, are tested to determine if they are correctly working or not. In Membership Drive and Balloting system I have perform different unit testing.

1) Roles are managed in role management it is tested that it is properly adding, editing and deleting.

2) User management unit is also tested and it is found that it is properly adding, deleting user.

3) Balloting unit is properly tested and found that it is completed working as per requirement.

5.4.2. Integration Testing

Integration testing is the phase in software testing in which different software modules are combined and tested as a group. The purose of this testing is to show wether the different modules on combining can work correctly or not.

Role management and user management are dependent modules. It is tested that the after adding roles the new user can be added. If the roles are edited then the user functionality also changed.

5.4.3. Security Testing

In the security testing, it is tested that the balloting system is well secured from the unauthorized access or not. We also assure that our confidential data will be safe and not used for any wrong activity using password encryption.

5.5. Test Plan

Test plan describe the contents that are linked to the testing of the system. It describes the strategy for the software testing. While testing the application, following cases are being considered:

- Testing Goals
- The key areas to be focused during the testing
- Functionality Testing
- Performance Testing
- Usability Testing
- Security

5.5.1. Testing Goals

Testing is done to find the errors, correct them, and make the program compliance with the user's requirements. Testing goals that meets the needs of user requirements are:

- Correctness
- Reliability
- Availability

- Maintainability
- Efficiency/Performance
- Validation
- Verification

5.5.2. Key Areas to be focused while testing

The focus is on the functionality and usability of the application. The focus is also on the acceptance testing. The planning process outlines the user acceptance testing strategy. Rather the focus on all types of testing to meet the user needs in successful way. The requirements are changed during the development of software. It is to be focused on the requirements that are changed from time to time. The software is developed as the requirements are changed for making it successful.

5.5.3. Functionality Testing

It is to be tested to show that all functions of our system is in working condition or not. And after updating the system whether the system is still perform its all requirement or not. In the testing of functionality, the following are also tested:

- Forms
- Links
- Database Connectivity
- Cookies

5.5.4. Performance Testing

Performance testing is in general testing performed to determine how a system performs in terms of responsiveness and stability under a particular workload.

5.5.5. Usability Testing

It is the process by which human-computer interaction characteristics of the system are measured and the weaknesses are identified for the correction.

5.5.6. Security

The security testing determines that the application protects data and maintains functionality as intended. It is tested that our system is secure from the vulnerabilities. In security of the application the confidentiality, integrity, authentication, authorization and the availability of data is tested. Security is verified through the proper encryption of the password through the encrypting algorithm. Thus the unauthorized person can't get access to the secure password

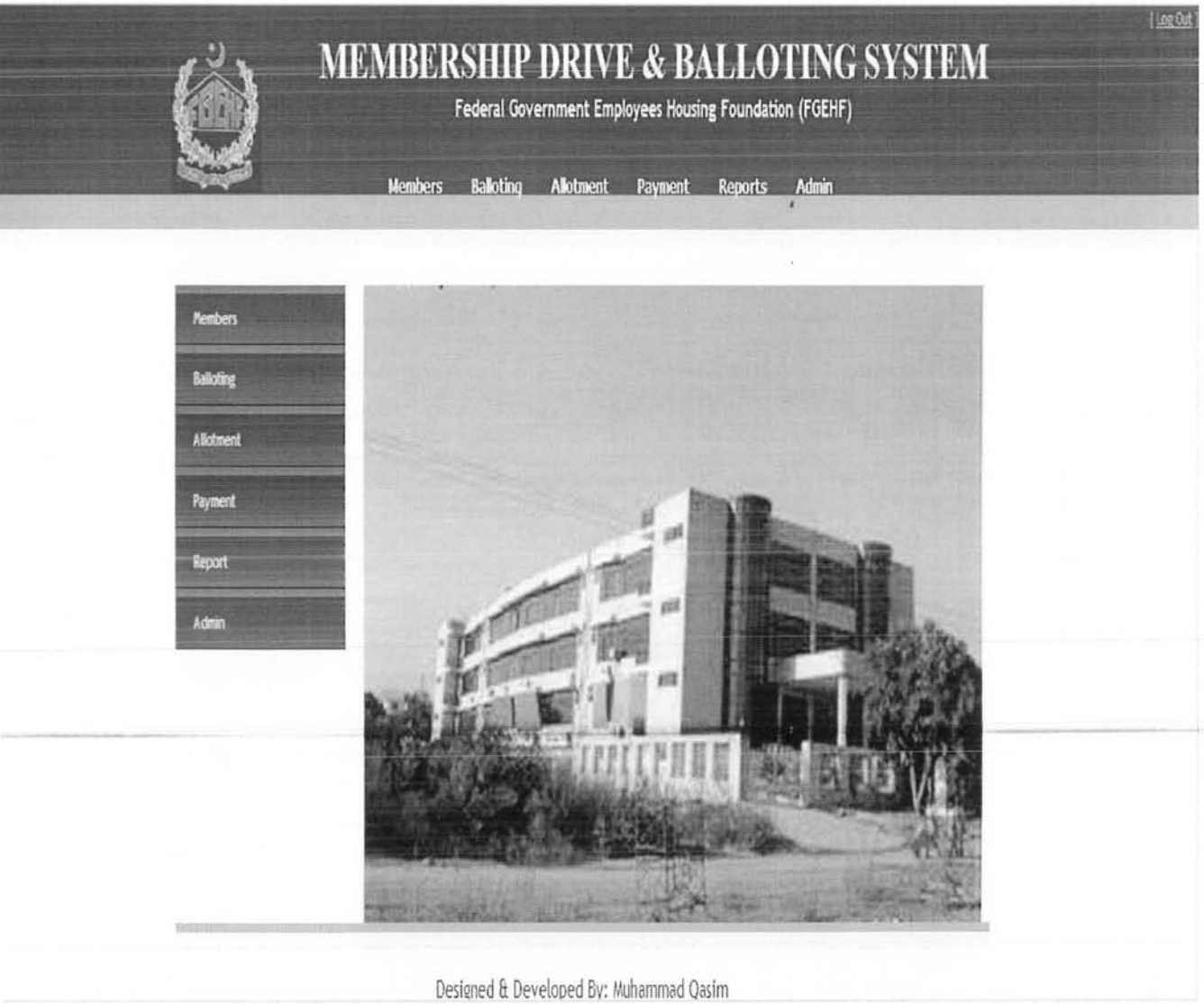
Chapter 6
User Interface Design

6.1. Login

In login page only user can login. Only that user can login who is registered and his user name and password exist in the database.

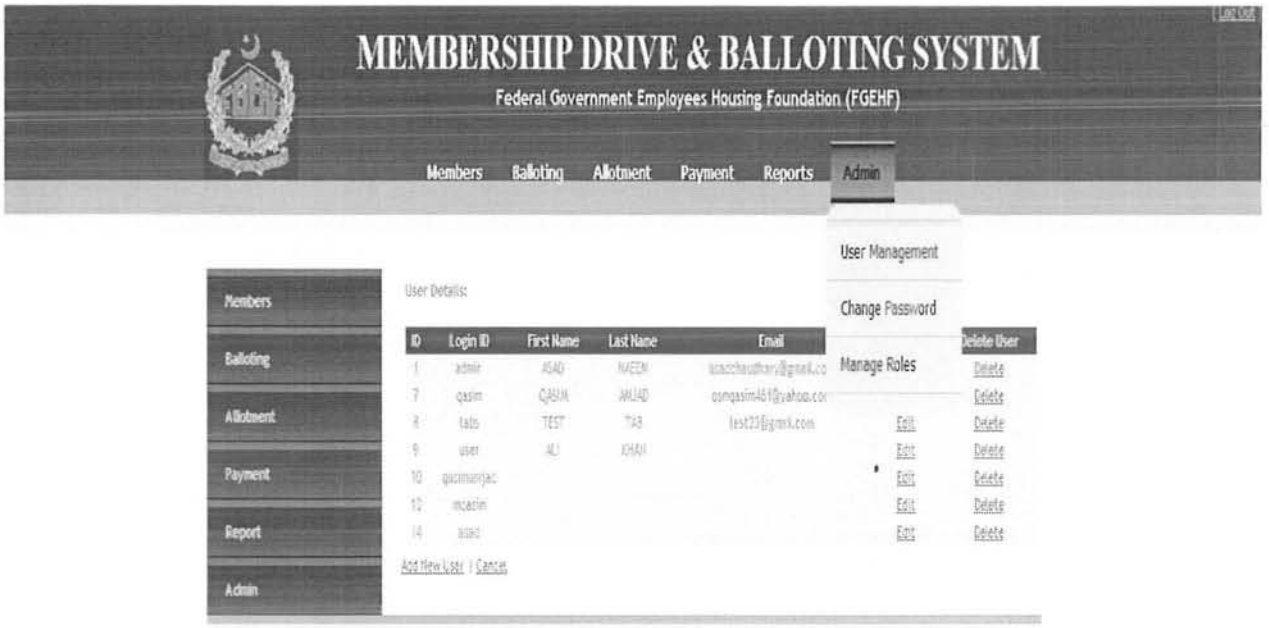
6.2. Home Page

In home page, user can manage membership, balloting, report, allotment, reports and payment, administration.



6.3. User Management

In user management, we manage users. Assign them user name and password. Only that user can login.



6.4. Search


MEMBERSHIP DRIVE & BALLOTING SYSTEM
Federal Government Employees Housing Foundation (FGEHF)
Members
Balloting
Allotment
Payment
Reports
Admin
Members
Balloting
Allotment
Payment
Report
Admin
Select category
1
Select Quota
AUTONOMOUS
Search
Print
Designed & Developed By: Muhammad Qasim

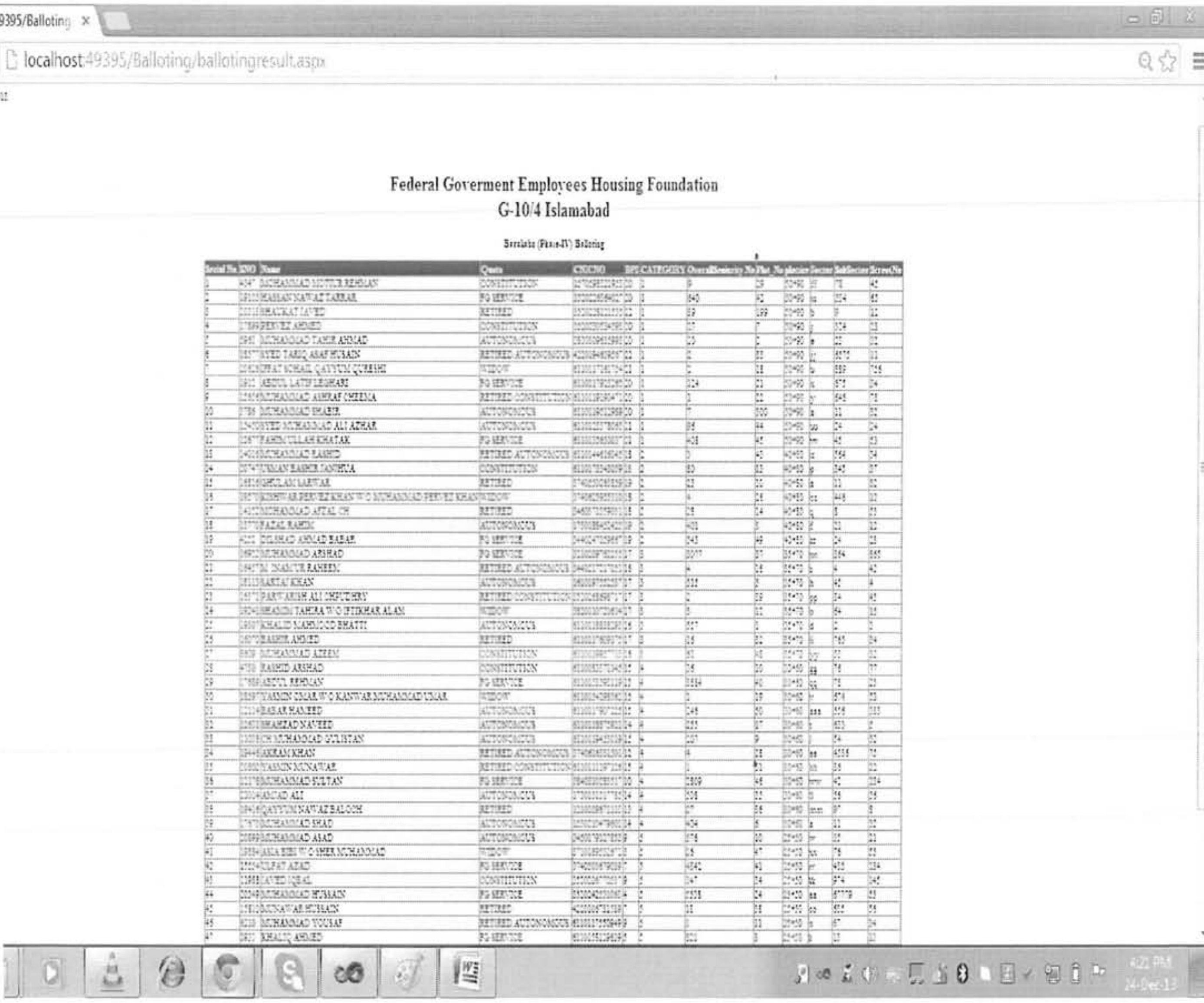
6.5. Role Management

Designed & Developed By: Muhammad Qasim

6.6. Pay Installments

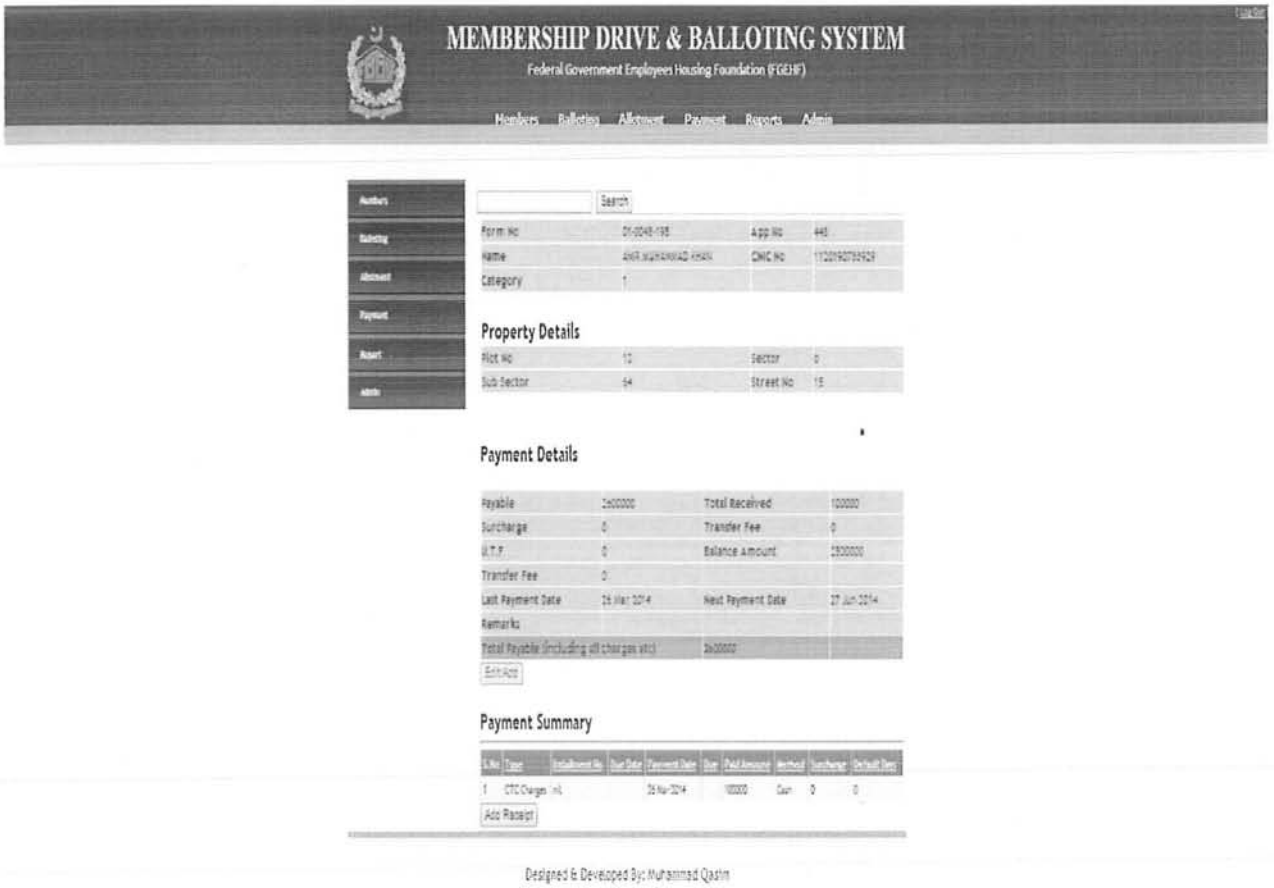
Designed & Developed By: Muhammad Qasim

6.7. Balloting

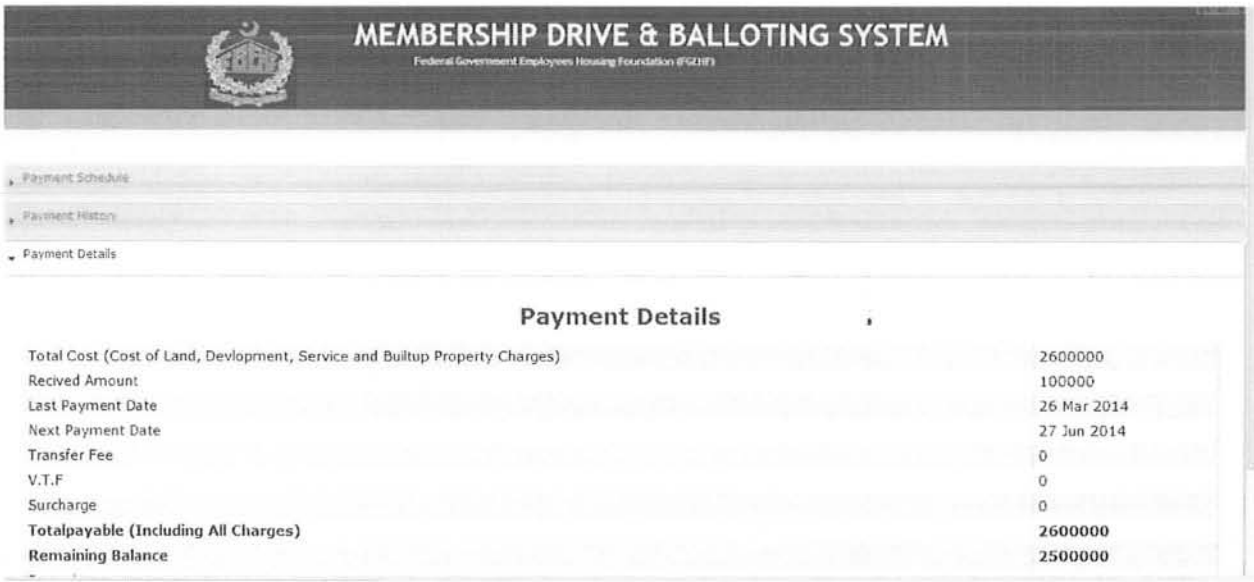


6.8. Add Payment

In it, give all the details of payments of members



6.1. Payment Details of User



Chapter 7
Conclusion and Future Enhancements

7.1. Conclusion

After the completion of this software, it will save time, reduce the working hours spent in office. The organization's prevailing system is a file system, which is very time consuming. With the help of this software you can easily search, edit, and delete membership record. It is secure because only admin have the authority to add and delete users. Admin can assign roles to users such as which functionality is accessible for user. Only the registered users can login. Membership drive and balloting system is developed as a generic system. Balloting System randomly selects data without the involvement of biasness thus the chance of fraud is kept to minimum level. Thus the manual system of balloting is time consuming and there will be a chance of fraud in it. The security is accomplished by encrypting the password. Thus in case an unauthorized user get access to database he can't find out the password for the particular user because he doesn't know which encrypting algorithm is applied on the password.

7.2. Future Enhancements

The membership drive and balloting system is successful software but it may need some enhancement. This project is for Federal Government Employees Housing Foundation (FGEHF) but by making a few changes in it we can use it for other organizations as well. In this system the organization staff is responsible for the payment of the installments of members but in near future it enhanced to that level so that users by sitting at home can deposit their installments through their debit/credit cards.

Glossary

Term	Description
GUI	Graphical User Interface
XAML	Extensible Application Mark-up Language
TC	Test Cases
UC	User Cases
HTML	Hypertext Mark-up Language
XHTML	Extensible Hypertext Mark-up Language
ASP	Active Server Pages
DSL	*Digital Subscriber Line*
GB's	Giga Byte's
RDBMS	Relational Database Management System
SQL	Structured Query Language
CSS	Cascading Style Sheet
ADO	*ActiveX Data Object*
IIS	Internet Information Services

References

1]. Pressman, Roger S. (2001), Software Engineering: a practitioner's approach. 5th edition.

2]. Danuza Neiva, Fernando Cesar de Almeida, Eduardo Santana de Almeida, Silvio Lemos Meira. (2010), A Requirements Engineering Process for Software Product Lines.

3]. Dhirendra Pandey, U. Suman, A. K. Ramani. (2010), an Effective Requirement Engineering Process Model for Software Development and Requirements Management.

4]. www.asp.net.com, last accessed 2013

5]. www.msdn.com, last accessed 2013