PROJECT REPORT

ON

SOCIETY MANAGEMENT SYSTEM

Submitted By:

Faculty of Computer Science & Applications

Amrapali Institute of Applied Sciences, Haldwani

In the partial fulfillment for the award of degree of Bachelor of Computer Application

Project Guide: Submitted by

Mr. N.S Dasila Pandit Naveen Baidnath



Amrapali Group of Institutes, Haldwani 2020

DECLARATION

I declare that final semester report entitled "SOCIETY MANAGEMENT (SM)" is my own work conducted under the supervision of the external guide Mr. N.S Dasila from AITS.

I further declare that to the best of my knowledge the report from the BCA final semester does not contain part of the work which has been submitted for the award of BCA degree either in this or any other university without proper citation.

Candidate Signature: Pandit Naveen Baidnath

Branch: CS **Student Roll No.:** 170610330112

Submitted To:

Department of Science and Technology,

Amrapali Institute

Introduction and Abstract Modules System requirements Product Definition Problem Statement Function to be provided Processing Environment: H/W, S/W Solution strategy. Why is the particular topic choosen Acceptance Criteria. Feasibility Analysis Project Plan Project Analysis Tools and Technology Development Schedule		7 8 8-22
Design	Use-Case DiagramDetailed DFDs and Structure diagramE-R Diagram	26-43 43
Data dictionary		44-46
Test Plan	Functional, Performance, Stress tests etc.Test Cases	47-66 47-50 51-66
Result		67-75
Project legacy (Conclusion) • Future scope • Bibliography		76

ACKNOWLEDGEMENT

Constant Inspiration and encouragement given by all concerned was the driving force that enables me to submit this project. Guidance, direction, cooperation, love and care pave the way to accomplish the task which seemed almost impossible in the beginning.

I am extremely thankful to entire team of Amrapali Institute of Technology and Science for their throughout guidance, encouragement and immense support. This period of training provided me an opportunity to enhance my software development knowledge and helped me in inculcating and developing the spirit of professionalism, sincerity and dedication as required in the software industry.

I express my sincere thanks & deep gratitude to all faculty members of **AITS**& who had helped me to add new dimensions to my knowledge. I acknowledge the cooperation extended by my colleague for making moments to be cherished for a lifetime.

No words were enough to describe the overwhelming support & inspiration of my parents, who stood by me through thick & thin.

PREFACE

The project training in the 6th semester of the course has given us the exposure to the real world. It is my great pleasure to present the project on "SOCIETY MANAGEMENT(SM)" which we conceived during the schedule of BCA (6thSemester). We have been successful in developing the project in a given time limit and will give my best.

The aim of project training is to have practical experience of the real world knowledge in the IT industry. The project development is the acid test of the theoretical knowledge of student after completing the course. The aim of developing this project is to reduce the gap between theoretical and practical knowledge.

We have put on all my efforts for preparing this project as best as possible within a time limit. We have tried our best to satisfy all the requirements of the user. We hope that the concerned authority will kindly accept this project.

PROJECT TITLE: Society Management

INTRODUCTION

Generally, in Society all the work is decided in meetings and maintenance bills, contact no of members are recorded on the papers. There is no automated system for doing all the things that generally happens in society, so that members can come to know what is happening in society. The Society Management System allows members to login with their own account and get updated with society happenings. Society Management System is the website portal to reduce conflicts among society members. The system has automated functionality for calculating monthly maintenance bill and member can view their bill status on their account.

OBJECTIVE:

System for a society is based on our traditional way keeping records and details on paper and registers. Access of these details and papers are not granted to common member in absence of the authority. Proposed system has a facility for bill generate, complain post to admin, view notice of there society online which will provide anytime anywhere access. This software system generates bill automatically and manually. It creates bill of all members at single click. due day of bills can be assigned. This software concentrates mainly on generation of bills whereas our system integrates various other features of society which are exact replica of the real happenings in the society. Our system provides more efficient and accurate results and can be accessed anywhere anytime.

Society Management System is the website portal designed to reduce conflicts among society members. The system has automated functionality for calculating maintenance and member can view their bill status on their account. The main functionality of this project is that, there is a complaining and bill generating. Member can complain to admin by just push the post of complain. The Society Management System allows members to login with their own account and get updated with society happenings. There is a payment option only BHIM UPI for future there will be available for card payment. Now just have to pay your charges to admin directly or by BHIM UPI.

MODULES

ADMIN:

- **Admin Account:** Admin login. Admin can check and add various member details.
- Society Member data: Admin can add and update member data.
- **Flats**: In this section, admin can manage flats (Add/Update).
- **Allotment**: In this section, admin can manage the allotment (Add/Update).
- **Bills:** In this section, the admin can manage bills (Add/Update).
- **View Complain:** In this section, the admin can view the complaint of society members and response to the complain and change the status of complaint according to complain status.
- **NOTICE**: In this section, admin can broadcast upcoming event in society.

USER:

- **Home**: It is a welcome page for society members. And notice for user of there particular society.
- **View Bill**: In this section, user can view his/her own society charges which will be generated by admin per month.
- Complain: In this section, user can raise the complain and view the complain status which is provided by society admin.
- **My Profile**: In this section user can change his/her password, email and phone number.
- **Contact**: where user can contact detail of management.

SYSTEM REQUIREMENT

PRODUCT DEFINATION

To overcome the drawbacks of the existing system, Here we provide a smarter and efficient way to handle the critical issues by reducing efforts and advancements in a reliable communication. Different functions in the society like Complaints, Notices, bill generation, Contacts will available within a single sight so that user can observe it and make use of it whenever is necessary. We propose an application which will be useful for all the society members to get constantly update with society related information. This System allows to log in for staff at owner containing their name, password, address, phone number. Society members can know the society maintenance amount and get email notification for the same. Once payment is made, history of all last payments is stored in the database. It will be the very much reliable way of communication. Any at member can ask for plumber, carpenter, electrician, and society will arrange them for the flat members. Each technician is assigned an id details. The flat owners can report The payment of monthly bills done online. This application provides an easy solution for management and maintains the daily functions within society in more disciplined and easy way. It is the smarter efficient way of communication between society residents and higher authorities.

PROBLEM STATEMENT

The functions of a housing society are an inevitable part of our lives. There are chores here which unknowingly take up a considerable amount of our lives. Managing committee often gets tired of maintaining multiple email groups; excel sheets containing members' contact information of owners and at the same time addressing the grievances of the housing society residents. Worse is the case where technology remains unused. Also, residents get restless that issues are not getting resolved despite reminders and no one knows the status of the complaint raised. Such issues and many more are common in most housing societies. With no appropriate tools, managing a residential complex takes too much of time, effort and money with a lot of inefficiency. As times have changed, most of us have strived to combine technology with our daily chores irrespective of the field. Thus, changing the way of maintaining the society information will also prove to be beneficial, improve efficiency and save us time.

FOR FLAT

The problem faced by admin that when admin add new flat of same flat no and same wing then it doesn't show's any error. But it create duplicate flat which is impossible.

FOR ALLOTMENT OF USER

The problem faced by admin that when admin want to delete user then flat also deleted with user account.

FOR USER NOT GETTING MAIL

The problem faced by user that when complain or bill generated to user they can't get mail this problem where fixed in several hour.

FUNCTIONTO BE PROVIDED

The features this project possess and the facilities which are provided are as follows:

Different functions within society such as Complaints, Notice, Contacts will available within a single sight so that user can navigate easily among those.

LOGIN

User must login and fill their required details before use the services.

NOTICE BOARD

At Home page of site we provide notice board so society member to know about every event conducting on society.

COMPLAIN

Complaints by user are can be attended by admin as reached to him through mail thus authenticity of complaint is maintained and thus complaint posting can be uninterrupted.

BILL GENERATE

There is a function which create bill for user and that bill is generated in pdf file so user can view that bill.

Processing Environment: H/W, S/W

Hardware:

The development of this application is processed under:

- **i3 7th Gen** powered processor
- Memory 1GB
- Hard Drive 250GB

4 Software:

- Internet connection
- Sublime Text 3
- PHP 7.2
- WAMP Server
- MY SQL 5.6

Solution strategy

Objective:

The main focus of this application is to save time for the user through non physical queue system on the other hand it saves time and energy for maintain the registry records for users

Scope:

The area of this application is designed to serve under society field by building a best communication between the admin and their users.

Competitive advantage:

This project is on web base where bootstrap also uses there a lot benefit of using bootstrap such as:

- Multi-Browser Compatibility.
- Extraordinary Grid Framework.
- Standardised HTML Syntax.
- Excellent Pre-styled Components.
- Frequent Updates.
- Packaged JavaScript Modules.
- Simple Integration.
- Rich Documentation.

WHY IS THE PERTICULAR TOPIC CHOOSEN

• Simple and User-Friendly

Society management system needs to have a simple and user-friendly interface so that all stakeholders can easily access it. One of the most essential criteria to keep in mind while choosing housing management software is the user-friendliness – it should not require any specialized technical expertise and every person involved in society administration including house-owners should be comfortable with its functionality.

The login-based access controls should enable user's quick connectivity to the system, and adequate security measures should be integrated into the system to prevent fraud or misuse of sensitive information.

• Bill Management

The payment of accounts and bills is a very vital part of efficient society governance, and sophisticated housing management software systems handle the payment of water, electricity and other comprehensively.

People can also check the status of their maintenance payments from the comfort of their homes, and generate payout schedules to ensure timely payments

Acceptanace Criteria

Acceptance criteria are a range of conditions that a given product must satisfy before the end-user, consumer, or group of stakeholders accepts it. These set of statements specify whether a project passes or fails the functionalities and requirements set forth by users. In simple terms, they represent if a customer is satisfied with certain criteria or not.

Project Acceptance criteria are criteria that include performance requirements and essential conditions, which must be met before project deliverables.

They set out the specific circumstances under which the user will accept the final output of the project.

Acceptance Criteria must be expressed clearly, in simple language the customer would use and without any ambiguity. They must be testable: easily translated into one or more manual/automated test case.

FEASIBILITY ANALYSIS

Feasibility study is test of a system proposal according to its workability, impact on the organization, ability to meet user needs and effective use of resources. All the projects are feasible given unlimited resources and infinite time! Ergo, feasibility study means an evaluation of benefits versus costs incurred in developing project, where cost includes manpower, time, resources and money.

A purpose of feasibility study is to check out the possibility of a computerized solution to the organization's observed problem before very much money that has been spent on.

A feasibility study is carried out to select the best system that meets performance requirements. Only by spending the time to evaluate the feasibility do I reduce the chances for extreme embarrassment at later stage of the system project.

Each of the feasible area was then abstracted into proposed modules for "BI and DW REPORTING". For the complete feasibility study I need to concentrate on following area:

Economic Feasibility:-

Among the most important information contained in feasibility study is the cost-benefit analysis. That is, an assessment of economic justification for computer-based system. Cost-benefit analysis delineates cost for

development and weights them against tangible and intangible benefits in the system. A system can be developed technically and that will be used if installed must still be a good investment for the organization. In the economical feasibility, the development cost in creating the system is evaluated against the ultimate benefit derived from the new systems. Financial benefits must equal or exceed the costs.

The system is economically feasible. It does not require any addition hardware or software. Since the interface for this system is developed using the existing resources and technologies available at NIC, There is nominal expenditure and economical feasibility for certain.

Technical Feasibility:-

Technical analysis evaluates technical merits of the system at the same time collecting additional information about performance, reliability, maintainability and productivity. In some cases, this system analysis step also includes a limited amount of research and design.

- proposed equipments have the technical capacity to hold the data required to use the new system?
- Will the proposed system provide adequate response to inquiries, regardless of the number or location of users?
- Can the system be upgraded if developed?
- Are there technical guarantees of accuracy, reliability, ease of access and data security?

Earlier no system existed to cater to the needs of 'Secure Infrastructure Implementation System'. The current system developed is technically feasible. It is a web based user interface for audit workflow at NIC-CSD. Thus it provides an easy access to the users.

Operational Feasibility:

Operational feasibility measures how well the solution will work in the organization and how will end-user & management feels about the system. Proposed system is helpful for all the users associated with the organization.

It will allow the administrator to have up-to-date information regarding all the aspects of their users. The decision-making process will also become faster with the use of data integration, consolidation. So it is feasible to implement the system.

Proposed projects are beneficial only if they can be turned out into information system. That will meet the organization's operating requirements. Operational feasibility aspects of the project are to be taken as an important part of the project implementation. Some of the important issues raised are to test the operational feasibility of a project includes the following: -

- Is there sufficient support for the management from the users?
- Will the system be used and work properly if it is being developed and implemented?
- Will there be any resistance from the user that will undermine the possible application benefits?

This system is targeted to be in accordance with the above-mentioned issues. Before hand, the management issues and user requirements have been taken into consideration. So there is no question of resistance from the users that can undermine the possible application benefits.

PROJECT PLAN

Programming Languages and Development Tools:

The language used to develop the entire application is **PHP** which is considered under the best programming language for many decades.

Development tools used are:

- Sublime Text 3
- Chrome/I.E

PROJECT ANALYSIS

Analysis is an important part of any project. If Analysis is not done properly then the whole project moves in the wrong direction. It also provides a schedule for the proper project work.

Analysis task divided into 3 areas:

- Problem Recognition
- Requirement Analysis
- Feasibility Study

Problem Recognition:

It is the phase in which the Current need for the System is to be defined. This site of Computer Peripherals & Consumables has all the up to date information. Regarding to all Computer Peripherals & Consumables.

Feasibility Study:

Feasibility study of the system is a very important stage during system design.

Feasibility study is a test of a system proposal according to its workability impact on the organization, ability to meet user needs, and effective use of resources. Feasibility study decides whether the system is properly developed or not.

Requirement Analysis:

A software product always begins with the customers' needs. These needs initially are either in the mind of the customer. Sometimes it is present in the existing practice where the need is to automate a current manual process.

These software requirements which are there in the mind of the users are usually informal and not general.

This phase basically converts these informal needs from the user into a set of formal requirement.

This phase ends up with the SRS (System Requirements Specification).

The Requirement Phase has three stages:

Problem Analysis

The goal of problem analysis is to obtain a clear understanding of the requirements of the client and the users.

1. Requirement Specification (SRS)

The SRS makes an Agreement between the user(client) of the system and the developer on what the software product will do.

2. Requirement Validation

It validates whether the requirement specification document does not have any error in it. The common errors that may occur are incorrect fact, inconsistency and ambiguity.

TOOLS & TECHNOLOGY

Different phases of the project require different tools to be used. In most application domain oriented projects, it is not type and nature of the tools used, but the manner in which they are utilized and the overall process follow that is important.

The tools used are not only versatile in their ability to handle complex application but their ease of use lends the developer freedom for thinking about the nuances how to implement ideas as a whole.

Instead the developer can concentrate upon the core application domain knowledge gathering and the design of the system from users view point and rest assured that the language and other tools will provide enough features to implement the same.

FRONT-END: HTML, CSS, JAVASCRIPT, JQuery, BOOTSTRAP

FRONT-END:

HTML

Hypertext Markup Language is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as CSS and scripting languages such as JavaScript.

Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

CSS

CSS describes how HTML elements are to be displayed on screen, paper, or in other media. CSS saves a lot of work. It can control the layout of multiple web pages all at once. Basically it will be used to make the web page more attractive and eye catchy. It works on the colours, layout, and fonts.

JAVASCRIPT

In this I'll be creating functions and this page will be linked to html page so that it will be easy to pass the elements.

JavaScript ("JS" for short) is a full-fledged dynamic programming language that can add interactivity to a website. It was invented by Brendan Eich (co-founder of the Mozilla project), the Mozilla Foundation, and the Mozilla Corporation. JavaScript is versatile and beginner-friendly. With more experience, you'll be able to create games, animated 2D and 3D graphics, comprehensive database-driven apps, and much more!

JavaScript itself is relatively compact, yet very flexible. Developers have written a variety of tools on top of the core JavaScript language, unlocking a vast amount of functionality with minimum effort. These include:

- Browser Application Programming Interfaces (APIs) built into web browsers, providing functionality such as dynamically creating HTML and setting CSS styles; collecting and manipulating a video stream from a user's webcam, or generating 3D graphics and audio samples.
- Third-party APIs that allow developers to incorporate functionality in sites from other content providers, such as Twitter or Facebook.
- Third-party frameworks and libraries that you can apply to HTML to accelerate the work of building sites and applications.

BOOTSTRAP

Bootstrap is a sleek, intuitive, and powerful, mobile first front-end framework for faster and easier web development. It uses HTML, CSS and Javascript. Bootstrap has its own icon library it uses in project at navigator side.

Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains CSS- and (optionally) JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components.

Bootstrap is the sixth-most-starred project on GitHub, with more than 135,000 stars, behind freeCodeCamp (almost 307,000 stars) and marginally behind Vue.js framework. According to Alexa Rank, Bootstrap is in the top-2000 in the USA while vuejs.org is in the top-7000 in the USA

JQUERY

jQuery is a lightweight, "write less, do more", JavaScript library. The purpose of jQuery is to make it much easier to use JavaScript on your website. jQuery takes a lot of common tasks that require many lines of JavaScript code to accomplish, and wraps them into methods that you can call with a single line of code. From project jQuery have main role it perform on data tables, deletion of data pop up, filter of data in tables, search perform, no of data show selectors.jQuery is a fast, small, cross-platform and feature-rich JavaScript library. It is designed to simplify the client-side scripting of HTML. It makes things like HTML document traversal and manipulation, animation, event handling, and AJAX very simple with an easy-to-use API that works on a lot of different type of browsers.

The main purpose of jQuery is to provide an easy way to use JavaScript on your website to make it more interactive and attractive. It is also used to add animation.

BACKEND: MySQL, PHP

BACKEND

MySQL

It is an open-source RDBMS. Its name is a combination of 'My', the name of cofounder Michael Widenius's daughter, and 'SQL'. It is used by many database-driven web applications, including Drupal, Joomla, phpBB, and WordPress.

It is also used by many popular websites, including Facebook, Flickr, MediaWiki, Twitter, and YouTube. Major features available in MySQL: Cross-platform support, Triggers, Cursors, Information schema, Updatable views, Full-text indexing and searching.

The PHP code consists of a core, with optional extensions to the core functionality. PHP's MySQL-related extensions, such as the MySQLi extension, and the MySQL extension, are implemented using the PHP extension framework. An extension typically exposes an API to the PHP developer, to allow its facilities to be used programmatically. However, some extensions which use the PHP extension framework do not expose an API to the PHP developer.

The PDO MySQL driver extension, for example, does not expose an API to the PHP developer, but provides an interface to the PDO layer above it.

MySQLi is an improved version of the older PHP MySQL driver, offering various benefits. [1]

The authors of the PHP scripting language recommend using MySQLi when dealing with MySQL server versions 4.1.3 and newer (takes advantage of new functionality)

PHP

Hypertext Preprocessor is a general purpose programming language originally designed for web development. PHP code may be executed with a CLI (Command Line Interface), embedded into HTML code, or used in combination with various web template systems, web content management systems, and web frameworks.

PHP connect to MySQL PHP 5 and later can work with MySQL database using

- MySQLi extension
- PDO

From project I used MySQLi where it is improved of MySQL.

Development Schedule

MILESTONE	DESCRIPTION	RELEASED DATE
M1	Front-end development	03/march/2020
M2	Back-end development	26/april/2020
M3	Integrating front-end and Back-end	07/may/2020
M4	Testing for initial release	17/may/2020
M5	Final release	05/june/2020

System Requirement Specifications

A SystemRequirements Specification (SRS) (also known as a Software Requirements Specification) is a document or set of documentation that describes the features and behavior of a system or software application. It includes a variety of elements (see below) that attempts to define the intended functionality required by the customer to satisfy their different users.

The production of the requirements stage of the software development process is SoftwareRequirements Specifications (SRS) (also called a requirements document). This report lays a foundation for software engineering activities and is constructing when entire requirements are elicited and analyzed. SRS is a formal report, which acts as a representation of software that enables the customers to review whether it (SRS) is according to their requirements. Also, it comprises user requirements for a system as well as detailed specifications of the system requirements.

The SRS is a specification for a specific software product, program, or set of applications that perform particular functions in a specific environment. It serves several goals depending on who is writing it. First, the SRS could be written by the client of a system. Second, the SRS could be written by a developer of the system. The two methods create entirely various situations and establish different purposes for the document altogether. The first case, SRS, is used to define the needs and expectation of the users. The second case, SRS, is written for various purposes and serves as a contract document between customer and developer.

Development Responsibility

I, Pandit naveen, would be developing the web-based project and I am responsible for the creation of the Database and all the other related stuffs.

PROJECT RISK

- ✓ Risk analysis and management are a series of steps that help a software team to understand and manage uncertainty.
- ✓ Many problems can infect a software project.
- ✓ A risk is a potential problem -- it might happen, it might not.
- ✓ But, regardless of the outcome, it's a really good idea to identify it, assess its probability of occurrence, and estimate its impact.
- ✓ Risk management aims at dealing with all kinds of risks that might affect a project.

• Type of risk

Risk identification is a systematic attempt to specify threats to the project plan.

By identifying known and predictable risks the project manager takes step toward avoiding them when possible and controlling them when necessary.

One method for identifying risks is to create a risk item checklist.

The checklist can be used for risks identification and focuses on some subset of known and predictable risks in the following generic subcategories:

Product size - risks associated with the overall size of the software to be built or modified.

Business impact - risks associated with constraints imposed by management or the marketplace.

Customer characteristics -risks associated with the constraints of the customer and the developer's ability to communicate with the customer in a timely manner.

Process definition -risks associated with the degree to which the software process has been define and is following by the development organization.

Development environment -risks associated with the availability and quality of the tools to be used to build the product.

Technology to be built -risks associated with the complexity of the system to be built and the "newness" of the technology that is packaged by the system.

Staff size and experience -risks associated with the overall technical and project experience of the software engineers who do the work.

The checklist can be used for risks identification and focuses on some subset of known and predictable risks in the following generic subcategories:

- Product size risks associated with the overall size of the software to be built or modified.
- Business impact risks associated with constraints imposed by management or the marketplace.
- Customer characteristics -risks associated with the constraints of the customer and the developer's ability to communicate with the customer in a timely manner.
- Process definition -risks associated with the degree to which the software process has been define and is following by the development organization.
- Development environment -risks associated with the availability and quality of the tools to be used to build the product.
- Technology to be built -risks associated with the complexity of the system to be built and the "newness" of the technology that is packaged by the system.
- Staff size and experience -risks associated with the overall technical and project experience of the software engineers who do the work.

DESIGN

Software design is the process of envisioning and defining software solutions to one or more sets of problems. One of the main components of software design is the software requirementsanalysis (SRA). SRA is a part of the software development process that lists specifications used in software engineering. If the software is "semi-automated" or user centered, software design may involve user experience design yielding a storyboard to help determine those specifications. If the software is completely automated (meaning no user or user interface), a software design may be as simple as a flow chart or text describing a planned sequence of events. There are also semi-standard methods like Unified Modeling Language and Fundamental modelingconcepts. In either case, some documentation of the plan is usually the product of the design. Furthermore, a software design may be platformindependent or platform-specific, depending upon the availability of the technology used for the design.

The main difference between software analysis and design is that the output of a software analysis consists of smaller problems to solve. Additionally, the analysis should not be designed very differently across different team members or groups. In contrast, the design focuses on capabilities, and thus multiple designs for the same problem can and will exist. Depending on the environment, the design often varies, whether it is created from reliable frameworks or implemented with suitable design patterns. Design examples include operation systems, webpages, mobile devices or even the new cloud computing paradigm.

Software design is both a process and a model. The design process is a sequence of steps that enables the designer to describe all aspects of the software for building. Creative skill, past experience, a sense of what makes "good" software, and an overall commitment to quality are examples of critical success factors for a competent design. It is important to note, however, that the design process is not always a straightforward procedure; the design model can be compared to an architect's plans for a house. It begins by representing the totality of the thing that is to be built (e.g., a three-dimensional rendering of the house); slowly, the thing is refined to provide guidance for constructing each detail (e.g., the plumbing lay). Similarly, the design model that is created for software provides a variety of different views of the computer software. Basic design principles enable the software engineer to navigate the design process

Use case diagram

The use case diagram are usually referred to as behavior diagram used to describe the actions of all user in a system. All user describe in use case are actors and the functionality asactionofsystem.

The Use case diagram is a collection of diagram and text together that make action on goal of a process.

In Society management system there is a two actor can do all the activities to run the system. Admin and User.

To model a system, the most important aspect is to capture the dynamic behavior. Dynamic behavior means the behavior of the system when it is running/operating.

Only static behavior is not sufficient to model a system rather dynamic behavior is more important than static behavior. In UML, there are five diagrams available to model the dynamic nature and use case diagram is one of them. Now as we have to discuss that the use case diagram is dynamic in nature, there should be some internal or external factors for making the interaction.

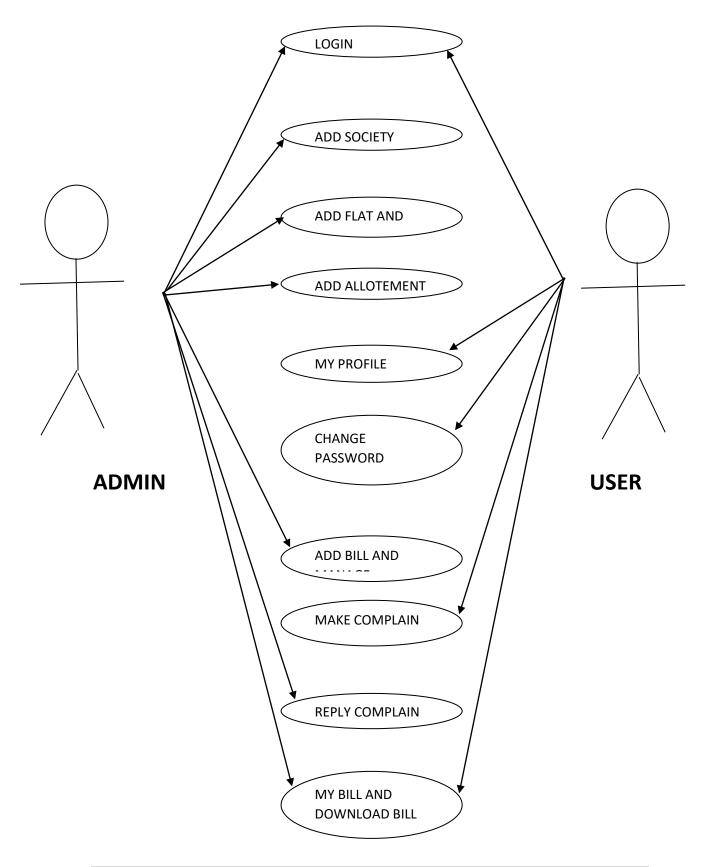
These internal and external agents are known as actors. Use case diagrams consists of actors, use cases and their relationships. The diagram is used to model the system/subsystem of an application. A single use case diagram captures a particular functionality of a system.

Use case diagram elements:

The use case diagram consist of six graphics elements that represent whole system:

- Systems
- Actors
- Use cases
- Association
- Dependencies
- Generalization

Usecase Diagram



ACTIVITY DIAGRAM

The activity diagram used to describe flow of activity through a series of actions. Activity diagram is a important diagram to describe the system. An activity diagram shows the overall flow of control.

Activity Diagrams describe how activities are coordinated to provide a service which can be at different levels of abstraction. Typically, an event needs to be achieved by some operations, particularly where the operation is intended to achieve a number of different things that require coordination, or how the events in a single use case relate to one another, in particular, use cases where activities may overlap and require coordination. It is also suitable for modeling how a collection of use cases coordinate to represent business workflows

- 1. Identify candidate use cases, through the examination of business workflows
- 2. Identify pre- and post-conditions (the context) for use cases
- 3. Model workflows between/within use cases
- 4. Model complex workflows in operations on objects
- 5. Model in detail complex activities in a high level activity Diagram

Activities

It is a behavior that is divided into one or more actions. Activities are a network of nodes connected by edges. There can be action nodes, control nodes, or object nodes. Action nodes represent some action. Control nodes represent the control flow of an activity. Object nodes are used to describe objects used inside an activity. Edges are used to show a path or a flow of execution. Activities start at an initial node and terminate at a final node.

Activity partition/swimlane

An activity partition or a swimlane is a high-level grouping of a set of related actions. A single partition can refer to many things, such as classes, use cases, components, or interfaces.

If a partition cannot be shown clearly, then the name of a partition is written on top of the name of an activity.

Fork and Join nodes

Using a fork and join nodes, concurrent flows within an activity can be generated. A fork node has one incoming edge and numerous outgoing edges. It is similar to one too many decision parameters. When data arrives at an incoming edge, it is duplicated and split across numerous outgoing edges simultaneously. A single incoming flow is divided into multiple parallel flows.

A join node is opposite of a fork node as It has many incoming edges and a single outgoing edge. It performs logical AND operation on all the incoming edges. This helps you to synchronize the input flow across a single output edge.

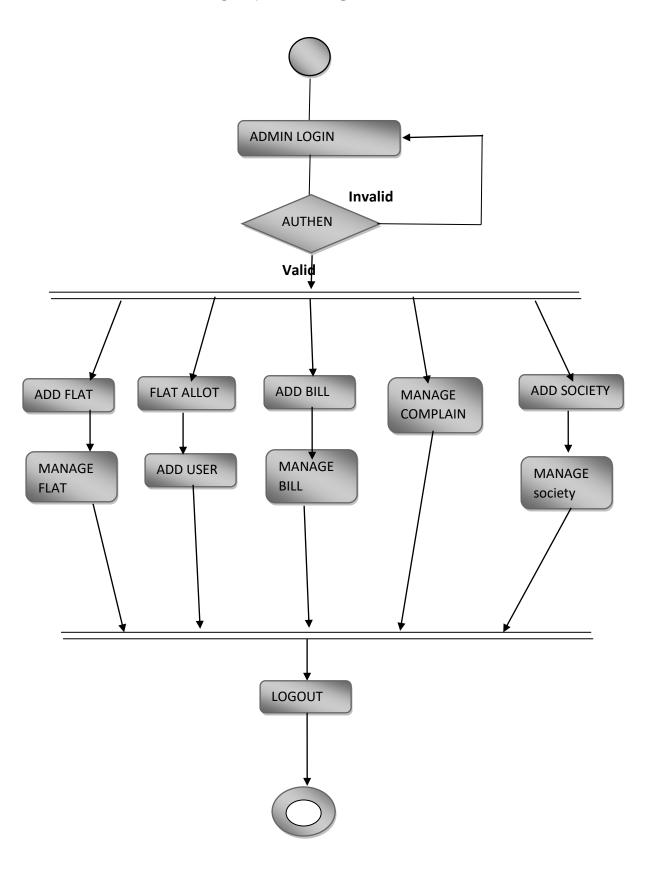
Pins

An activity diagram that has a lot of flows gets very complicated and messy.

Pins are used to clearing up the things. It provides a way to manage the execution flow of activity by sorting all the flows and cleaning up messy thins. It is an object node that represents one input to or an output from an action.

Both input and output pins have precisely one edge.

ACTIVITY DIAGRAM



DFD {Data Flow Diagram}

 Data flow design is concerned with designing a sequence of functional transformation that converts system imports into the required outputs. The design is represented as data flow diagram illustrated how data flows through a system and how the output is derived from the input through a sequence of functional transformations.

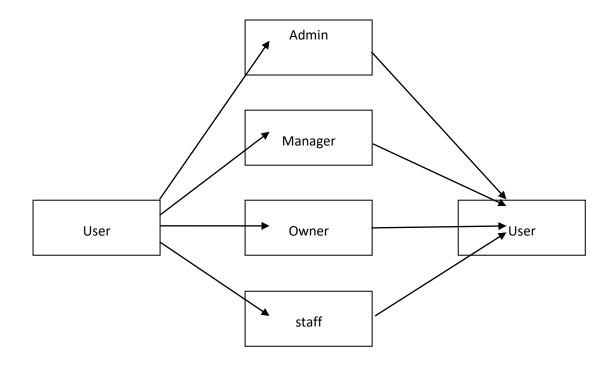
Data flow diagrams are useful and intuitive way to describing a system. They are normally understandable without special training, especially if control information is excluded.

They show end-to-end processing. That is the flow of processing from when data enters the system to where it leaves the application can be traced.

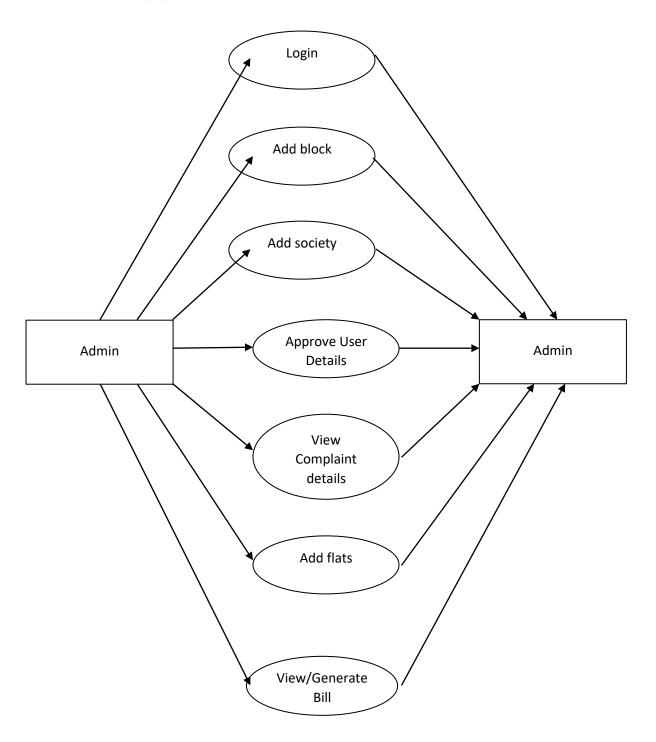
0 Level DFD



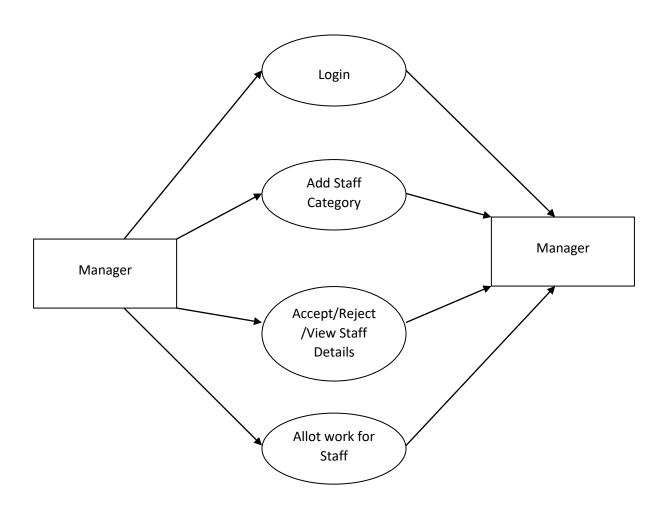
1 Level DFD



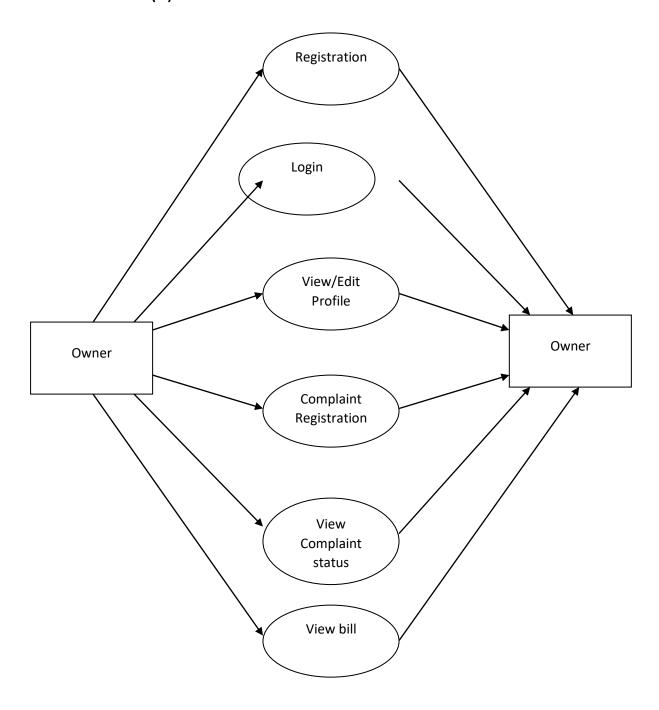
2 Level DFD (a)



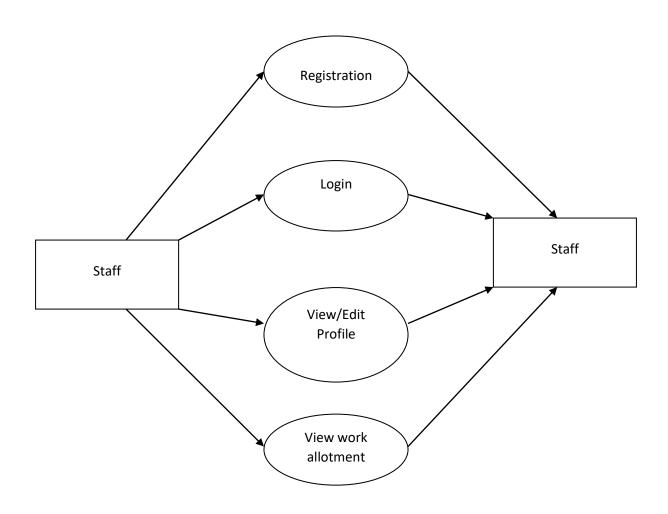
2 Level DFD (b)



2 Level DFD (c)



2 Level DFD (d)



Flowcharts

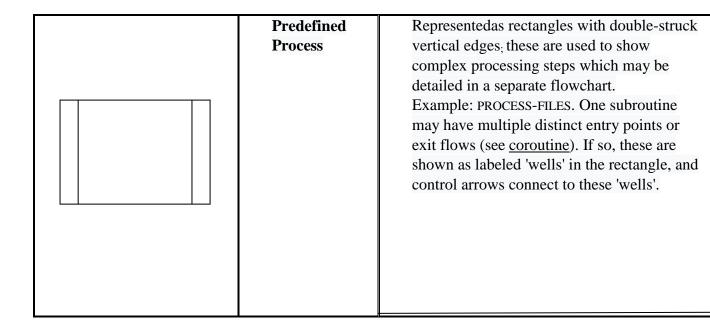
A flowchart is a type of diagram that represents an algorithm, workflow or process, showing the steps as boxes of various kinds, and their order by connecting them with arrows. This diagrammatic representation illustrates a solution model to a given problem. Flowcharts are used in analyzing, designing, documenting or managing a process or program in various fields. Flowcharts are used in designing and documenting simple processes or programs. Like other types of diagrams, they help visualize what is going on and thereby help understand a process, and perhaps also find flaws, bottlenecks, and other less-obvious features within it.

There are many different types of flowcharts, and each type has its own repertoire of boxes and notational conventions. The two most common types of boxes in a flowchart area processing step, usually called activity, and denoted as a rectangular box a decision, usually denoted as a diamond. A flowchart is described as "cross-functional" when the page is divided into different swim lanes describing the control of different organizational units. A symbol appearing in a particular "lane" is within the Control of that organizational unit. This technique allows the author to locate the responsibility for performing an action or making a decision correctly, showing the responsibility of each organizational unit for different parts of a single process. Flowcharts depict certain aspects of processes and they are usually complemented by other types of diagram. For instance, Kaoru Ishikawa defined the flowchart as one of the seven basic tools of quality control, next to the histogram, Pareto chart, check sheet, control chart, cause-and-effect diagram, and the scatter diagram. Similarly, in UML, a standard concept-modeling notation used in software development, the activity diagram, which is a type of flowchart, is just one of many different diagram types. Nassi-Shneiderman diagrams and Dracon-charts are an alternative notation for process flow.

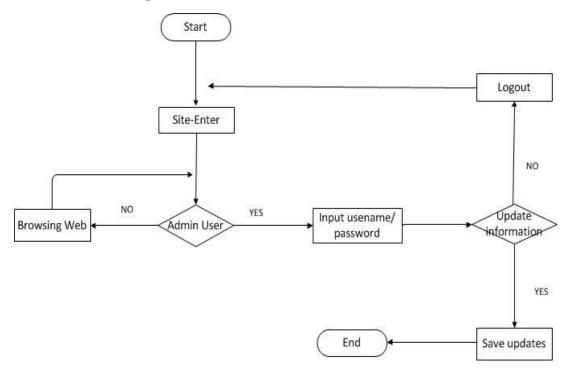
Common alternative names include: flow chart, process flowchart, functional flowchart, process map, process chart, functional process chart, business process model, process model, process flow diagram, work flow diagram, business flow diagram. The terms "flowchart" and "flow chart" are used interchangeably.

Shape	Name	Description
	flow line	An arrow coming from one symbol and ending at another symbol represents that control passes to the symbol the arrow points to. The line for the arrow can be solid or dashed. The meaning of the arrow with dashed line may differ from one flowchart to another and can be defined in the legend.
	Terminals	Represented as circles, ovals, stadiums or rounded (fillet) rectangles. They usually contain the word "Start" or "End", or another phrase signaling the start or end of a process, such as "submit inquiry" or "receive product"
	Process	Represented as rectangles. This shape is used to show that something is performed. Examples: "Add 1 to X", "replace identified part", "save changes", etc

Decision	Represented as a diamond (rhombus) showing where a decision is necessary, commonly a Yes/No question or True/False test. The conditional symbol is peculiar in that it has two arrows coming out of it, usually from the bottom point and right point, one corresponding to Yes or True, and one corresponding to No or False.
Input/output	Represented as a parallelogram. Involves receiving data and displaying processed data. Can only move from input to output and not vice versa. Examples: Get X from the user; display X.
Preparation	Represented as a hexagon. May also be called initialization. Shows operations which have no effect other than preparing a value for a subsequent conditional or decision step. Alternatively, this shape is used to replace the Decision Shape in the case of conditional looping.



• Admin Login

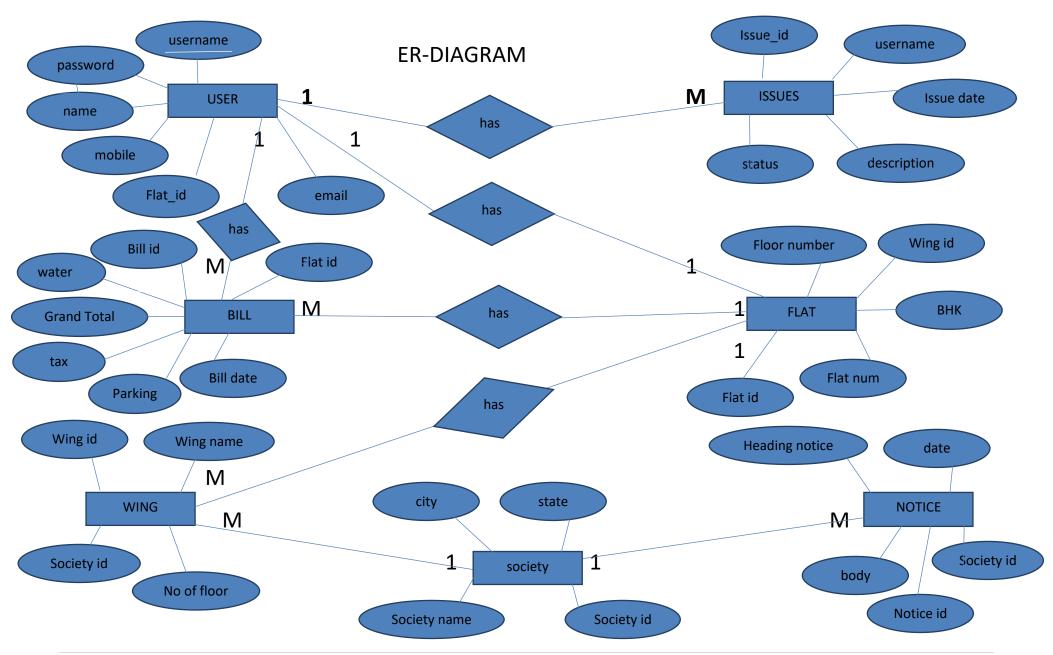


ER-DIAGRAM

An entity—relationship model (or ER model) describes interrelated things of interest in a specific domain of knowledge. A basic ER model is composed of entity types (which classify the things of interest) and specifies relationships that can exist between entities (instances of those entity types).

In software engineering, an ER model is commonly formed to represent things a business needs to remember in order to perform business processes. Consequently, the ER model becomes an abstract data model that defines a data or information structure which can be implemented in a database, typically a relational database.

Entity—relationship modeling was developed for database and design by Peter Chen and published in a 1976 paper, with variants of the idea existing previously. Some ER models show super and subtype entities connected by generalization-specialization relationships, and an ER model can be used also in the specification of domain-specific ontology's.



Data Dictionary

• ACCOUNT TABLE

S.no	NAME	Constraint	Datatype	
1	username	Primary	varchar	
2	Flat_id	foreign key	Int	
3	Password		Varchar	
4	Fullname		Varchar	
5	Email		Varchar	
6	Mobile_no		Int	
7	Past address		Varchar	
8	Pending dues		Decimal	
9	Photo		Varchar	

• ADMIN TABLE

S.no	NAME	Constraint	Datatype
1	Admin id	Primary	int
2	username		varchar
3	Password		Varchar
4	Fullname		Varchar
5	Email		Varchar
6	Mobile_no		Int
7	Past address		Varchar

• Alert table

S.no	NAME	Constraint	Datatype	
1	Alert_id	primary	int	
2	Flat_id	foreign key	int	
3	message		Varchar	
4	status		int	

• BILL TABLE

S.no	NAME	Constraint	Datatype	
1	Bill_num	primary	Int	
2	Flat_id	foreign key	int	
3	Bill_date		Date	
4	Watercharge		Decimal	
5	Parking charge		Decimal	
6	Tax charge		Decimal	
7	Due date		Date	
8	Grand total		Decimal	

• FLAT TABLE

S.no	NAME	Constraint	Datatype	
1	Flat_id	primary	int	
2	Wing_id	foreign	Int	
3	Flat num		Int	
4	Bhk		int	
5	Floor no		Int	
6	Status		Int	

• ISSUE TABLE

S.no	NAME	Constraint	Datatype	
1	issue_id	primary	int	
2	Username	foreign	Varchar	
3	Issue date		Date	
4	Issue desc		Varchar	
5	Status		Int	
6	Mailed		varchar	

• NOTICE TABLE

S.no	NAME	CONSTRAINT	DATATYPE	
1	notice_id	primary	int	
2	Society_id	foreign	Int	
3	Notice header		Varchar	
4	Notice content		Varchar	
5	Notice date		Date	

SOCIETY TABLE

S.no	NAME	CONSTRAINT	DATATYPE	
1	society_id	primary	int	
2	Society_name		Varchar	
3	City		Varchar	
4	State		Varchar	

WING TABLE

S.no	NAME	CONSTRAINT	DATATYPE	
1	wing_id	primary	int	
2	Society_id	foreign	Int	
3	Wing name		Varchar	
4	No of floor		int	

TESTING

INTRODUCTION-

Testing is vital to the success of any system. Testing is done at different stages within the development phase. System testing makes a logical assumption that all parts of the system is correct and the goals will be achieved successfully.

Inadequate testing or no testing leads to errors that may come up after a long time when correction would be extremely difficult.

Another objective of testing is its utility as a user-oriented vehicle before implementation.

The testing of the system was done on both test and user data. The following tests are performed.

Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design and coding. In fact, testing is the one step in the software engineering process that could be viewed as destructive rather than constructive.

A strategy for software testing integrates software test case design methods into a well-planned series of steps that result in the successful construction of software.

Testing is the set of activities that can be planned in advance and conducted systematically. The underlying motivation of program testing is to affirm software quality with methods that can economically and effectively apply to both strategic to both large and small-scale systems.

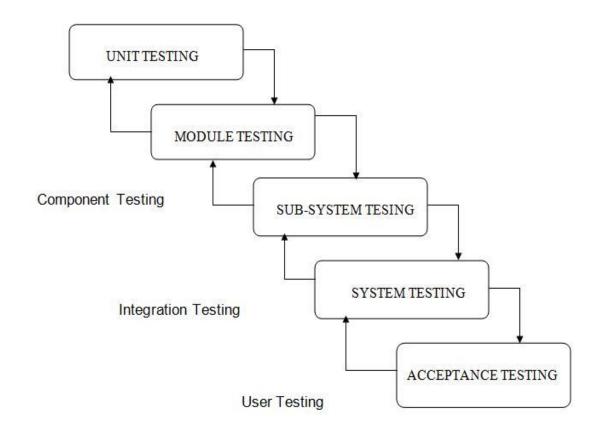
STRATEGIC APPROACH TO SOFTWARE TESTING-

The software engineering process can be viewed as a spiral. Initially system engineering defines the role of software and leads to software requirement analysis where the information domain, functions, behavior, performance, constraints and validation criteria for software are established. Moving inward along the spiral, we come to design and finally to coding. To develop computer software we spiral in along streamlines that decrease the level of abstraction on each turn.

A strategy for software testing may also be viewed in the context of the spiral. Unit testing begins at the vertex of the spiral and concentrates on each unit of the software as implemented in source code.

Testing progress by moving outward along the spiral to

Integration testing, where the focus is on the design and the construction of the software architecture. Talking another turn on outward on the spiral we encounter validation testing where requirements established as part of software requirements analysis are validated against the software that has been constructed. Finally we arrive at system testing, where the software and other system elements are tested as a whole.



UNIT TESTING

In unit testing the focuses is on the verification of the smallest unit of the project that is a module or a function. In unit testing we work according to white box testing that is providing the input set and checking the output is in accordance with the expected output or not.

1. White Box Testing

This type of testing ensures that

- All the independent modules and function should be executed at least once in the testing phase.
- All the inputs of must include the boundary values & middle values.
- All the logical decisions must be have output as true or false.

To follow the concept of white box testing we have tested each of the above mentioned forms. Tests are done to ensure correct flow of data in the system. All conditions of the system are exercised to check their accuracy

2. Conditional Testing

In Conditional testing, each condition is tested to both true and false aspects. And all the resulting paths of true and false output are tested. So that each path that may be generate on particular condition is traced to uncover any possible errors in the system.

All the conditions like selecting the option button for pdf and word document, not selecting the option button, providing the file path, not providing the file path etc.

3. Data Flow Testing

Data Flow Testing selects the path of the program according to the location of definition and use of variables. This kind of testing is used only when some local variable were declared and their scope in the program is to be tested. The *definition-use chain* method is used in this type of testing. It is used in situations like selecting the word document option button and then checking file is converted to .pdf format or not.

TEST INTRODUCTION

The test gives a brief idea of the correct expected output and the incorrect output. This test confirms proper behavior of the GUI during system initialization and startup. The unit under test should initialize without any errors.

Test Cases

The test cases should be written to get the basic idea of the input provided to the system and the expected output to be received from the system.

• Admin Login

Test Case id	Test scenario	Test Steps	Test Data	Expected Results	Actual results	Pass/fai
1	Check user login with valid data	Step 1- Run the login page, click on user login, Login page will open, and ask to enter username and password. Step 2- Enter username Step 3- Enter Password Step 4- Click Login Button	Username- xyz Password: 123	User should login into applicatio n	As Expected	Pass

2	Check	Step 1- Run	Username:	User	As	Pass
	user	the home	xyz	should	Expected	
	login	page, click on	Password:	not be		
	with in-	user login,	123456	able to		
	valid	Login page		login into		
	data	will open, and		applicatio		
		ask to enter		n		
		username and				
		password.				
		Step 2- Enter				
		username				
		Step 3- Enter				
		Password				
		Step 4- Click				
		Login Button				

• User Login

Test Case id	Test scenario	Test Steps	Test Data	Expected Results	Actual results	Pass/fai 1
1	Check user login with valid data	Step 1- Run the login page, click on user login, Login page will open, and ask to enter username and password. Step 2- Enter username Step 3- Enter Password Step 4- Click Login Button	Username- xyz Password: 123	User should login into applicatio n	As Expected	Pass

2	Check	Step 1- Run	Username:	User	As	Pass
	user	the login	xyz	should	Expected	
	login	page, click on	Password:	not be	_	
	with in-	user login,	123456	able to		
	valid	Login page		login into		
	data	will open, and		applicatio		
		ask to enter		n		
		username and				
		password.				
		Step 2- Enter				
		username				
		Step 3- Enter				
		Password				
		Step 4- Click				
		Login Button				

ADD BILL

Test Case id	Test scenario	Test Steps	Test Data	Expected Results	Actual results	Pass/fai 1
1	Check admin enter valide data	Step 1- Run the mantenance page, click on ADD BILL, Form will open, and ask to enter Bill Date ,water charge, parking charge and due date. Step 2- Enter bill date Step 3- Enter water charge Step 4- Enter parking charge Step 4- Click submit Button	Bill date- 2020- 06-16 Water charge- 500 Parking chage- 400 Due date- 2020-06-19	Query should fire to database And alert box open with message "successfu lly bill added".	As Expected	Pass

2	Check	Step 1- Run	Bill date-	Query	As	Pass
	Check	the	2020aug02	should not	Expected	
	user	mantenance		fire to	-	
	enter in	page, click on	Water charge-	database		
	valide	ADD BILL,	500	And alert		
	data	Form will		box open		
		open, and ask	Parking chage-	with		
		to enter Bill	400	message		
		Date ,water		"unsucces		
		charge,	Due date-	sfull bill		
		parking	asdfghjk	added".		
		charge and				
		due date.				
		Step 2- Enter				
		bill date				
		Step 3- Enter				
		water charge				
		Step 4- Enter				
		parking				
		charge				
		Step 4- Click				
		submit Button				

ADD FLAT

Test Case id	Test scenario	Test Steps	Test Data	Expected	Actual results	Pass/fai
1	Check there is no duplicat e flat	Step 1- Run the flat page, click on ADD flat, Form will open, and ask to enter flat id optional ,wing, flat number, bhk and floor. Step 2- Select wing Step 3- Select flat number Step 4- Select Bhk Step 4- Click submit Button	wing- 200-A flat number- 01 Bhk- 4	Results Query should fire to database And alert box open with message "successfu lly Flat added".	As Expected	Pass
2	Check there is duplicat e flat	Step 1- Run the flat page, click on ADD flat, Form will open, and ask to enter flat id optional ,wing, flat number, bhk and floor. Step 2- Select wing Step 3- Select flat number Step 4- Select Bhk Step 4- Click submit Button	Flat id – 1 wing- 200-A flat number- 01 Bhk- 4	Query should not fire to database And alert box open with message "Flat already exist can not add Duplicate"	As Expected	Pass

Allottment

Test Case id	Test scenario	Test Steps	Test Data	Expected Results	Actual results	Pass/fai 1
1	Check there is no duplicat e usernam e	Step 1- Run the allottment page, click on ADD allot, Form will open, and ask to enter username ,passward, select flatid, email, phone number and balance pending. Step 2- enter username Step 3- Select flat id Step 4- enter password Step 5 enter email Step 6 enter phone number Step 7- Click submit Button	username- xyz flat id- 19 Password- 123 Email- xyz@gmail.com	Query should fire to database And alert box open with message "successfu lly user added".	As Expected	Pass

2	Check	Step 1- Run	username-	Query	As	Pass
	there is	the allottment	xyz	should not	Expected	
	no	page, click on	-	fire to	-	
	duplicat	ADD allot,	flat id-	database		
	e	Form will	19	And alert		
	usernam	open, and ask		box open		
	e	to enter	Password-	with		
		username	123	message		
		,passward,		"unsucces		
		select flatid,	Email-	sfull to		
		email,	xyz@gmail.com	add user".		
		phone number				
		and balance				
		pending.				
		Step 2- enter				
		username				
		Step 3- Select				
		flat id				
		Step 4- enter				
		password				
		Step 5 enter				
		email				
		Step 6 enter				
		phone number				
		Step 7- Click				
		submit Button				

Change password

Test Case id	Test scenario	Test Steps	Test Data	Expected Results	Actual results	Pass/fai
1	Check is new passwor d match with confirm passwor d	Step 1- Run the profile page, click on edit, Form will open, and ask to enter new passward, email, phone number and confirm password Step 2- enter new password Step 3- enter confirm password Step 7- Click submit Button	New password- 12345 Confirm password- 12345	On click submit button alert box message "successfu lly password updated	As Expected	Pass
2	Check is new passwor d mismatc h with confirm passwor d	Step 1- Run the profile page, click on edit, Form will open, and ask to enter new passward, email, phone number and confirm password Step 2- enter new password Step 3- enter confirm password Step 7- Click submit Button	New password- 12345 Confirm password- 123iuytr	On click submit button alert box popup and messege "password do not match".	As Expected	Pass

Page Session

Test Case id	Test scenario	Test Steps	Test Data	Expected Results	Actual results	Pass/fai
1	Check session time	Step 1- Run the any page till one hour		After one hour page session get destroy and move to again login page	As Expected	Pass

Upload photo

Test Case id	Test scenario	Test Steps	Test Data	Expected Results	Actual results	Pass/fai
1	Check for uploade d image path	Step 1 go to allotment page step 2 update user image or add user image. Step 3 click on submit	Choose image "xyz.jpeg" and Click on submit	Path "{wamp install dir}/www /society/u pload" The file present with name xyz.jpeg	As Expected	Pass
2	Check if the user is able to view the uploade d image	Step 1- Run the profile page.		User can view image to his profile	As Expected	Pass

3	Default	Step 1-goto	username-	At view	As	pass
	image	allottment	xyz	panal of	Expected	
	appear	page		allottmet		
	if image	Step 2 add	flat id-	that user		
	is not	user	19	default		
	uploade	Where form		pic added		
	d	appear	Password-			
		Step 3 ask to	123			
		enter user				
		detail and	Email-			
		image	xyz@gmail.co			
			<u>m</u>			
			image-			

Email Validation

Test Case id	Test scenario	Test Steps	Test Data	Expected Results	Actual results	Pass/fai
1	Check email is invalid	Step 1 go to allotment page Step 2 add allot or update allot Step 3enter detail of user Step 4 click on submit button	username- xyz flat id- 19 Password- 123 Email- Xyz.com	On click of submit button below the text bar of email message print "enter valid email".	As Expected	Pass

2	Check	Step 1 go to	username-	Successfu	As	Pass
	email is	allotment	xyz	l user	Expected	
	valid	page		added		
		Step 2 add	flat id-			
		allot or	19			
		update allot				
		Step 3enter	Password-			
		detail of user	123			
		Step 4 click				
		on submit	Email-			
		button	Xyz.com			

Phone Validation

Test Case id	Test scenario	Test Steps	Test Data	Expected Results	Actual results	Pass/fai
1	Check Phone is invalid	Step 1 go to allotment page Step 2 add allot or update allot Step 3enter detail of user Step 4 click on submit button	username- xyz flat id- 19 Password- 123 Phone- Xudlli758	On click of submit button below the text bar of phone message print "invalid".	As Expected	Pass

1	Check	Step 1 go to	username-	On click	As	
	Phone is	allotment	xyz	of submit	Expected	Pass
	valid	page		button		
		Step 2 add	flat id-	below the		
		allot or	19	text bar of		
		update allot		phone		
		Step 3enter	Password-	message		
		detail of user	123	print		
		Step 4 click		"valid".		
		on submit	Phone-			
		button	9999222292			

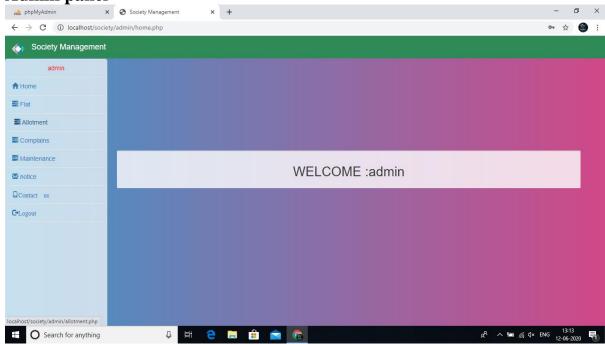
Username validation

Test Case id	Test scenario	Test Steps	Test Data	Expected Results	Actual results	Pass/fai
1	Check usernam e is duplicat e	Step 1 go to allotment page Step 2 add allot Step 3enter detail of user Step 4 click on submit button	username- xyz flat id- 19 Password- 123 Phone- Xudlli758	Pop up message "usernam e already existed"	As Expected	Pass

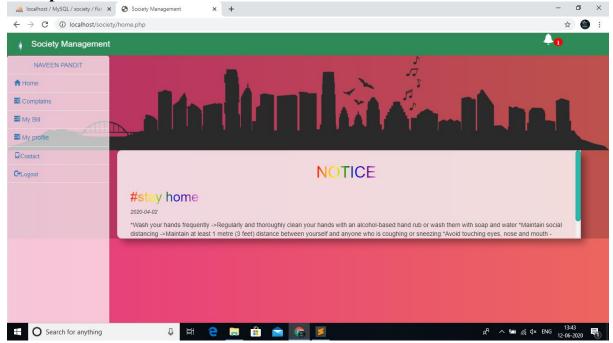
2	Check	Step 1 go to	username-	Successfu	As	
	usernam	allotment	xyz123	l allotted	Expected	Pass
	e is not	page		to user		
	duplicat	Step 2 add	flat id-	And open		
	e	allot	19	allotment		
		Step 3enter		page		
		detail of user	Password-			
		Step 4 click on submit	123			
		button	Phone- Xudlli758			

RESULT

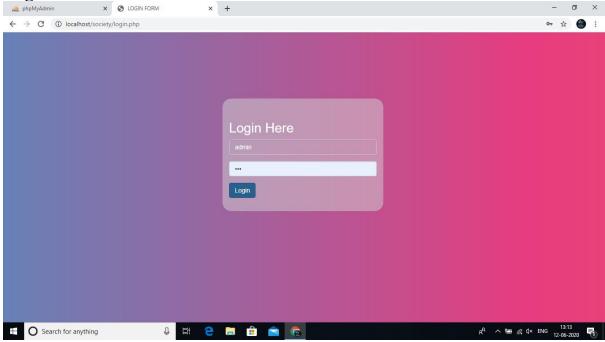
Admin panel



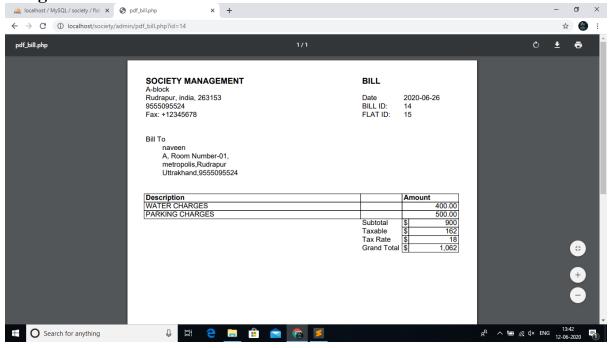
User panel



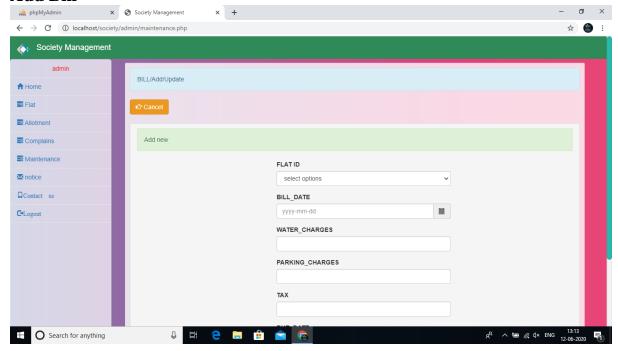
Login



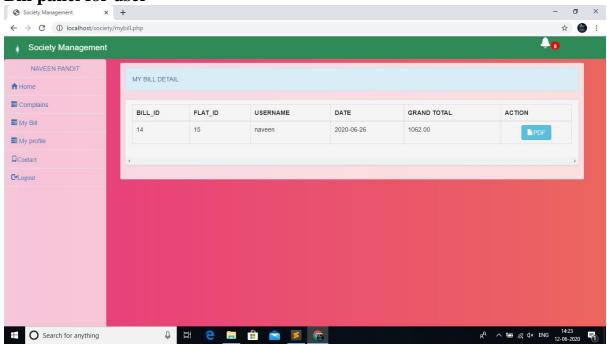




Add Bill

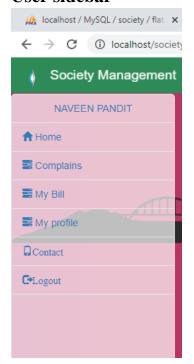


Bill panel for user



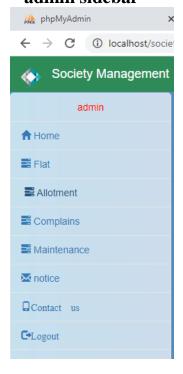
Sidebar menu

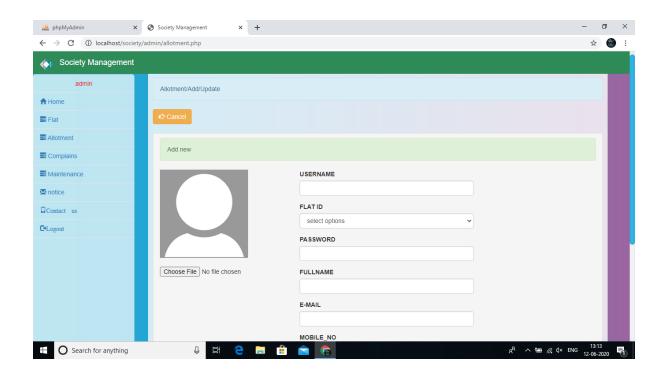
User sidebar



User registration

admin sidebar

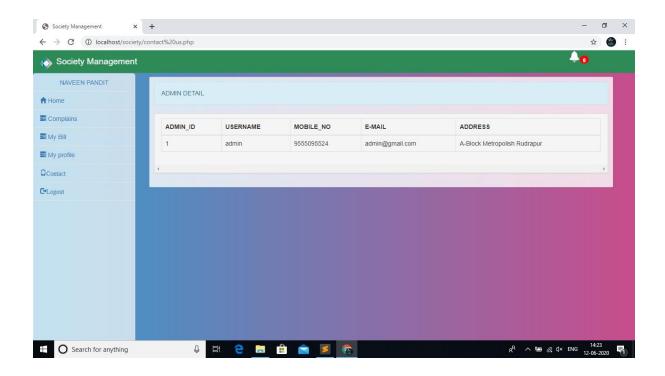




Notice



Contact us page



Small screen (mobile)



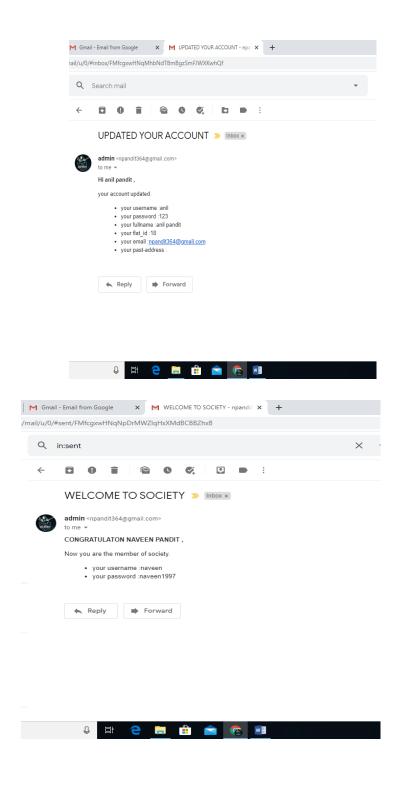


Alert box





Receive mail



Conclusion:

Society management system puts forth the actual working of a society. Administration, working, voting, etc. similar to a society are the key features of our project. User can access feeds and news from the society anywhere and anytime for their own comfort.

FUTURE SCOPE

This project can be enhanced further by adding online payment facility for the members to reduce the extra work of the admin. The software is flexible enough to be modified and implemented as per future requirements. We have tried our best to pre- sent this free and user–friendly website to Society members. Mobile number alerts for various happenings in the society can be added to the system so that users do not miss the updates and happenings of the society

Bibliography

- www.google.com
- www.stackoverflow.com
- http://www.w3.org/TR/html5
- www.wikipedia.com