A
Project Exhibition
on

Life Insurance Portal (JivanBima)

VIT BHOPAL
www.vitbhopal.ac.in

Submitted by

Arun Kumar Mudgal (21MCA10024)

Ajay Bhardwaj(21MCA10028)

Nitesh Kumar Verma(21MCA10060)

Arish Ali(21MCA10082)

Sandeep Rajput(21MCA10094)

Guided by

Dr. Devaraju S.
Senior Assistant Professor
(Internal Guide)

Submitted in partial fulfillment of the requirement for the degree of
"Master of Computer Applications"

Submitted to
School of Computing Science and Engineering
VIT Bhopal University
Bhopal (MP) – 466 114

May 2022

VIT BHOPAL UNIVERSITY, M P - 466114

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING

CANDIDATE'S DECLARATION

I hereby declare that the Dissertation entitled "Life Insurance Portal (JivanBima)" is my work conducted under the supervision of Guide/Co-Guide Dr.Devaraju S, Senior Assistance Professor, School of Computer Science of Engineering at VIT University, Bhopal.

I further declare that to the best of my knowledge this report does not contain any part of work that has been submitted for the award of any degree either in this university or in another university / Deemed University without proper citation.

Arun Kumar Mudgal (21MCA10024)

Ajay Bhardwaj (21MCA10028)

Nitesh Kumar Verma (21MCA10060)

Arish Ali (21MCA10082)

Sandeep Rajput (21MCA10094)

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Date: 19-May-2022

Dr. Devaraju S.

Senior Assistant Professor

VIT UNIVERSITY BHOPAL, M P – 466114

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the work embodied in this Capstone Project Report entitled **"Life Insurance Portal (JivanBima)"** has been satisfactorily completed by **Mr. Nitesh Kumar Verma** (21MCA10060), **Mr. Ajay Bhardwaj** (21MCA10028), **Mr. Arun Kumar Mudgal** (21MCA10024), **Mr. Sandeep Rajput** (21MCA10094), **Mr. Arish Ali** (21MCA10082), in the School of Computing Science and Engineering at VIT University, Bhopal. This work is a bonafide piece of work, carried out under my/our guidance in the School of Computing Science and Engineering for the partial fulfillment of the Master of Computer Applications.

Dr. Devaraju S.
Senior Assistant Professor

Forwarded by   Approved by

Dr. Kanchan Lata Kashyap   **Dr. S. Poonkuntran**
Program Chair   **Professor & Dean**

ACKNOWLEDGEMENT

I am ineffably indebted **to Dr. G. Viswanathan (Chancellor)**, Vellore Institute of Technology, Bhopal, and **Dr. U. Kamachi Mudali (Vice-Chancellor)**, Vellore Institute of Technology, Bhopal.
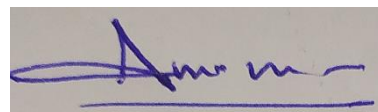
I would like to express my profound thanks to **Dr.S.Poonkuntran (Dean of SCSE)** for their support and encouragement towards me at every stage in the successful completion of my project and my degree.

I would like to extend my thanks and unbound sense for the timely help and assistance given to **Dr. Kanchan Lata Kashyap (Program Chair of Master of Computer Applications)** in completing the project.
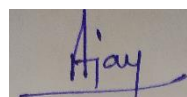
The satisfaction and euphoria that accompanies the successful completion of any task would be incomplete without the mention of the people who made it possible. If there is a driving force that kept used going on in doing the project, it is the constant support of my **internal guide Dr. Devaraju S.** I present my sincere and heartiest thanks to him, for giving me a patience hearing and clearing my doubts.

I take this opportunity to thank my parents and friends for their constant support and encouragement throughout this training.
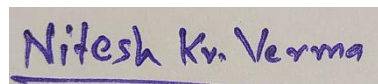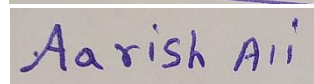
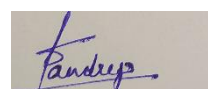**Arun Kumar Mudgal (21MCA10024)**

Ajay Bhardwaj (21MCA10028)

Nitesh Kumar Verma (21MCA10060)

Arish Ali (21MCA10082)

Sandeep Rajput (21MCA10094)

ABSTRACT

The purpose of the Life Insurance Portal is to automate the existing manual system with the help of computerized equipment and full-fledged computer software. The existing system is a paper process it takes a lot of time-consuming. So, we try to manage all processes with the help of the internet and computer to reduce the time. Because Everyone wants to save time. so that their valuable data/information can be stored for a longer period with easy access and manipulation of the same. The required software and hardware are easily available and easy to work with.

In this project, there are two-actor admin and client. Admin can manage everything such as adding client, deleting client, updating client data, adding policies, adding policy categories, updating policies, deleting policies, approving policies, rejecting policies, answering questions, managing the contacts list, etc.

Users can create an account in the portal after creating an account login in the portal after login users can see the policy and select policy as per choice and apply the policy. After applying the policy client wait some time policy will be approved by the admin after approval client can pay the 1st premium. Client can also ask questions regarding the policy.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

Everyone knows LIC is a big organization. It's provided policy to customers and very useful for us because it gives life insurance. But lots of the agents take commission through users. Users don't know how much the policy's price is. The client goes to the company and asks about policies through the agent. The agent explains all the policies and rules and regulations and the user purchases it but the user doesn't understand how much the agent takes a commission that's why we made this project whose name is "Life Insurance Portal (JivanBima)" that aims to assist people in different sectors to achieve security of their crucial data hence focusing on reducing agent commission. There are many policies and facilities are available you can easily search on LIC Portal all details are available which you want to search and no need to go anywhere. As our world is taking a thousand steps into digital life in a day, it is very important to look at the security part too. The Life Insurance Portal (JivanBima) is a web-based application. The objective of this project is to develop a system that automates the processes and activities of Insurance. The proposed system is highly automated and makes Insurance activities much easier and more flexible. The users can get the very right information at the very right time. This will increase the trust of the customer in the Insurance company as well. Life Insurance or Life Assurance is a contract between the policy owner and the insurer, where the insurer agrees to pay a designated beneficiary a sum of money upon the occurrence of an insured individual or event such as a critical illness. In return, the policy owner agrees to pay a stipulated amount.

1.1 Objective

The main objective of the project on the Life Insurance Portal is to remove the paper process and agent. It manages the activities related to insurance of the people. All the information and data catch up which providing insurance policy can be saved in to the database for which SQLite is used as a backend. Later on, the insurance holder for any product or insurance holder customers can view the details also using their login details. For rest of the users, the information of each other will keep confidential. The software is not limited up to providing insurance details of their customers but this project also works dynamically by providing various guideline tools for special insurance policies, schemes, terms and conditions, how to register and get extra benefits and all other details.

This software is designed keeping in mind to make it one of the best automation systems for keeping all the tracks of the details of their customers and for providing best services to them so that they can generate best revenues for themselves by putting up insurance which is best suited to them.

1.2 Proposed System

This software is aimed at:

- All the work will be digitalized and is done via computer and internet.

- Client can view policy and buy a policy as per convention.

- The client can claim the policy when he wants.

- The client can pay their premium when he wants within the deadline date.

- The payment environment is secure as compared to the existing system.

- Time consume will be reduced and users will get any easy way to access their insurance related information and new upcoming schemes.

- All the details regarding the insurance holder and schemes will be added via computer and the information data is being saved in servers.

- Backup should be there in case if by chance any of the information will be lost.

- Users just have to click on the button and just have to wait for some moments and they get an easy access to their information.

1.3 Scope

It may help collect perfect management in detail. In a very short time, the collection is obvious, simple, and sensible. It will help a client to know the policy. It also helps in current all works relative to Life Insurance Portal (JivanBima). It will also reduce the cost of collecting the management and the collection procedure will go on smoothly.

Our project aims at LIC process automation, we have tried to computerize various processes of the Life Insurance Portal (JivanBima).

- Have a good User Interface.
- Be easy to understand by the user.
- Easy to apply policy.
- Easy to claim policy.
- Easy to pay a premium.

CHAPTER 2

PROBLEM ANALYSIS

2.1 Product Definition

The purpose of this software is to provide facilities does a client wants. A general concept for this purpose. This software has multiple web pages and functionalities divided into multiple platforms. Clients have to know all perform to get a result.

2.2 Feasibility Analysis

A feasibility study is an analysis used in measuring the ability and likelihood to complete a project successfully including all relevant factors. It must account for factors that affect it such as economic, technological, legal, and scheduling factors. Project managers use feasibility studies to determine potential positive and negative outcomes of a project before investing a considerable amount of time and money into it.

A. Economic Feasibility

This is a very important aspect to consider while developing a project. We decided on the technology based on the minimum possible cost factor.

- All hardware and software cost has to be borne by the organization.
- Overall, we have estimated that the benefit organization is going to receive from the proposed system which surely overcome the initial cost and later running cost of the system.

B. Technical Feasibility

This included the study of function performance and constraints that may affect the ability to achieve an acceptable system. For this feasibility study, we studied the complete functionality to be provided in the system, as described in the System Requirement Specification (SRS), and checked if everything was possible using different types of frontend and backend platforms.

C. Operational Feasibility

No doubt the proposed system is fully GUI based is very friendly and all inputs to be taken all self-explanatory even to a layman. Besides, proper training has been conducted to let know the essence of the system to the users so that they feel comfortable with the new system. As far as our study is concerned the clients are comfortable and happy as the system has down their loads and doing.

CHAPTER 3

SOFTWARE & HARDWARE REQUIREMENT ANALYSIS

3.1 Software Requirements

Name Of Component	Specification
Operating System	Windows X, Windows7, Linux
Language	HTML, CSS, Python and Django
Web Browser	Chrome, Mozilla, Opera, etc.
Web Server	Localhost:8000
IDE	Visual Studio Code
Scripting Language	JavaScript

3.2 Hardware Requirements

Name Of Component	Specification
Processor	Intel Pentium IV with 2GHz or later
RAM	1 GB
Hard Disk	240 GB

3.3 Technologies

HTML 5

HTML5 is a markup language used for structuring and presenting content on the World Wide Web. It is the fifth and final major HTML version that is a World Wide Web Consortium recommendation. The current specification is known as the HTML Living Standard. It is maintained by the Web Hypertext Application Technology Working Group (WHATWG), a consortium of the major browser vendors (Apple, Google, Mozilla, and Microsoft).

HTML5 includes detailed processing models to encourage more interoperable implementations; it extends, improves, and rationalizes the markup available for documents and introduces markup and application programming interfaces (APIs) for complex web applications.

CSS

CSS is the acronym for "Cascading Style Sheets".

CSS is a computer language for laying out structuring web pages (HTML or XML).

This language contains coding elements and is composed of these "cascading style sheets" which are equally called CSS files.

JavaScript

JavaScript, often abbreviated JS, is a programming language that is one of the core technologies of the World Wide Web, alongside HTML and CSS. Over 97% of websites use JavaScript on the client-side for web page behavior, often incorporating third-party libraries. All major web browsers have a decided JavaScript engine to execute the code on users' devices. JavaScript is a high-level, often just-in-time compiled language that conforms to theECMAScript stand. It has dynamic typing, prototype-based object orientation, and first-class functions. It is multi-paradigm, supporting event-driven, functional, and imperative programming styles. It is multi-paradigm, supporting event-driven, functional, and imperative programming styles. It has application programming interfaces (APIs) for working with text, dates, regular expressions, standard data structures, and the Document Object Model (DOM)

Django

Django is free and open-source web application framework written in Python Programming language. It is based on MVT (Model View Template) design Pattern. The Django is very demanding due to Rapid development, Secure, Scalable feature. By using Django, we can build web applications in very less time. Django is designed in such a manner that it handles much of configure things automatically, so we can focus on application development only.

In this project, we use there are many packages, modules and classes such as: -

❖ **Packages**

- ➢ asgiref==3.2.7

- ➢ django-widget-tweaks==1.4.8

- ➢ pytz==2020.1

- ➢ sqlparse==0.3.1

- ➢ pillow==9.1.0

❖ **Modules**
- ➢ Django.shortcuts
    - Render
    - Redirect
    - Response
- ➢ Django.html
    - HttpResponseRedirect
- ➢ Django.contrib
    - message
- ➢ Django.contrib.auth.decorators
    - login_required
- ➢ Django.contrib.auth.model
    - user
- ➢ Datetime
    - Datetime

SQLite

SQLite is a database, which is zero configured, which means like other database you do not need to configure it in your system. SQLite engine is not a standalone process like other database, you can link it statically or dynamically as per your requirement with your application. SQLite accesses its storage files directly. is an open-source relational database management system. A relational database organizes data into one or more data tables in which data types may be related to each other; these relations help structure the data. SQL is a language programmers use to create, modify and extract data from the relational database, as well as control user access to the database.

(1) DDL

- Create
- Drop
- Alter

(2) DML

- Insert
- Update
- Delete

(3) DQL

- Select

(4) DCL

- Grant
- Revoke

CHAPTER 4

PROJECT PLAN

Project Description

There has been a continuous effort to develop tools, which can ease the process of software development. But, with the evolving trend of different programming paradigms today's software developers are challenged to deal with the changing technology. Among other issues, software reengineering is being regarded as an important process in the software systems that are already developed and to transform them into a different software environment. Generally, this requires a lot of manual effort in going through a program that might have been developed by another programmer. This project makes a novel attempt to address the issue of program analysis and generation of diagrams, which can depict the structure of a program in a better way. Today, UML is being considered an industrialstandard for software engineering design processes. It is essential to provide severaldiagramming tools that can express different aspects/characteristics of the program such as **Use Case:** Elicit Requirement from users in meaningful chunks. Construction planning isbuilt around delivering some use cases and each interaction basis for system testing.

Class Diagrams: Shows static structure of concepts, types, and class. Concepts how users think about the world; type shows interfaces of software components; classes show the implementation of software components.

Activity Diagram: Shows behavior with a control structure. Can show many objects over many uses, many objects in a single-use case or implementation methods encourage parallel behavior, etc.

The end product of this project is a comprehensive tool that can parse any vb.net program and extract most of the object-oriented features inherent in the program such as polymorphism, inheritance, encapsulation, and abstraction.

CHAPTER 5

SYSTEM ANALYSIS

Defining a System

A collection of components that work together to realize some objective forms a system. There are three major components in every system, namely input, processing, and output.

In a system, the different components are connected and they are interdependent. For example, the Human body represents a completely natural system. We are also bound by many national systems such as the political system, economic system, educational system,and so forth. The objective of the system demand that some output is produced as a result of processing the suitable inputs.

System Life Cycle

The system life cycle is an organizational process of developing and maintaining systems. It helps in establishing a system project plan because it gives an overall list of processes and sub-processes required to develop a system. System development life cycle means a combination of various activities. In other words, we can say that various activities put together are referred to as the system development life cycle. In the System Analysis and Design terminology, the system development life cycle means software development life cycle.

Following are the different phases of the software development cycle:

- System Analysis
- System Design
- Coding
- Testing
- Implementation
- Maintenance

System Analysis

Assuming that a new system is to be developed, the next phase is system analysis. The analysis involved a detailed study of the current system, leading to specifications of a new system. The analysis is a detailed study of various operations performed by a system and theirrelationships within and outside the system. During analysis, data are collected on the available files, decision points, and transactions handled by the present system. Interviews, on-site observations, and questionnaires are the tools used for system analysis. Using the following steps, it becomes easy to draw the exact boundary of the new system underconsideration:

- Keeping in view the problems and new requirements.
- Workout the pros and cons including new areas of the system.

All procedures and requirements must be analyzed and documented in the form of detailed data flow diagrams (DFDs), data dictionaries, logical data structures, and miniature specifications. System Analysis also includes subdividing a complex process involving the entire system, identification of data store, and manual processes.

The main points to be discussed in system analysis are:

- Specifications of what the new system is to accomplish based on the other requirements.
- Functional hierarchy showing the functions to be performed by the new system and their relationship with each other.
- Functional networks are similar to function hierarchy but they highlight those functions which are common to more than one procedure.
- List of attributes of the entities- there are the data items that need to be held about each entity (a record).

CHAPTER 6

SYSTEM DESIGN

System Design

Based on the user requirements and the detailed analysis of a new system, the new system must be designed. This is the phase of system designing. It is the most crucial phase in the development of a system. Normally, the design proceeds in two stages:

- Preliminary or General Design
- Structure or Detailed Design

Preliminary or general design: In the preliminary or general design, the features of the new system are specified. The costs of implementing these features and the benefits to be derived are estimated. If the project is still considered to be feasible, we move to the detailed design stage.

Structure or Detailed design: In the detailed design stage, computer-oriented work begins in earnest. At this stage, the design of the system becomes more structured. The structure design is a blueprint of a computer system solution to a given problem having the same components and inter-relationship among the same components as the original problem.Input, output, and processing specifications are drawn up in detail. In the design stage, the programming language and the platform in which the new system will run are also decided.

There are several tools and techniques used for designing. These tools and techniques are:

- Flow Chart
- DFD
- E-R Diagram
- Data Dictionary
- Decision Table
- Decision Tree

Data Flow Diagram

A Data Flow Diagram (DFD) is a diagram that describes the flow of data and the processes that change or transform data throughout a system. The Data Flow Diagram reviews the current physical system, prepares input and output specifications, specifies the implementation plan, etc.

Four basic symbols are used to construct data flow diagrams. They are symbols that represent data sources, data flows, data transformations, and data storage. The points at which data are transformed are represented by enclosed figures, usually circles, which are called nodes.
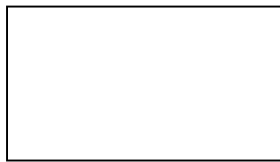
a) **Source of Destination of data**
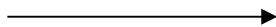
b) **Flow of Direction**
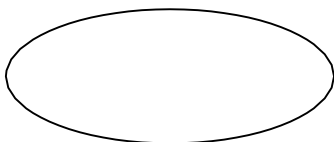
c) **Process**
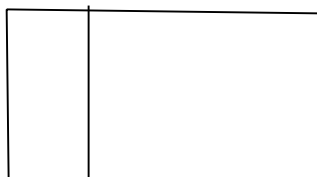
Process

d) **Process**

e) **Storage**

DFD Level 0
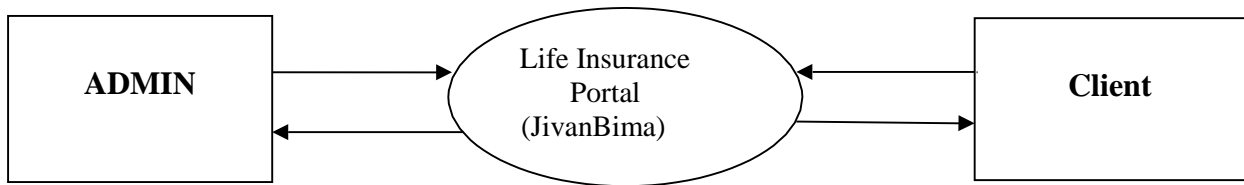
DFD Level
(Client Page)
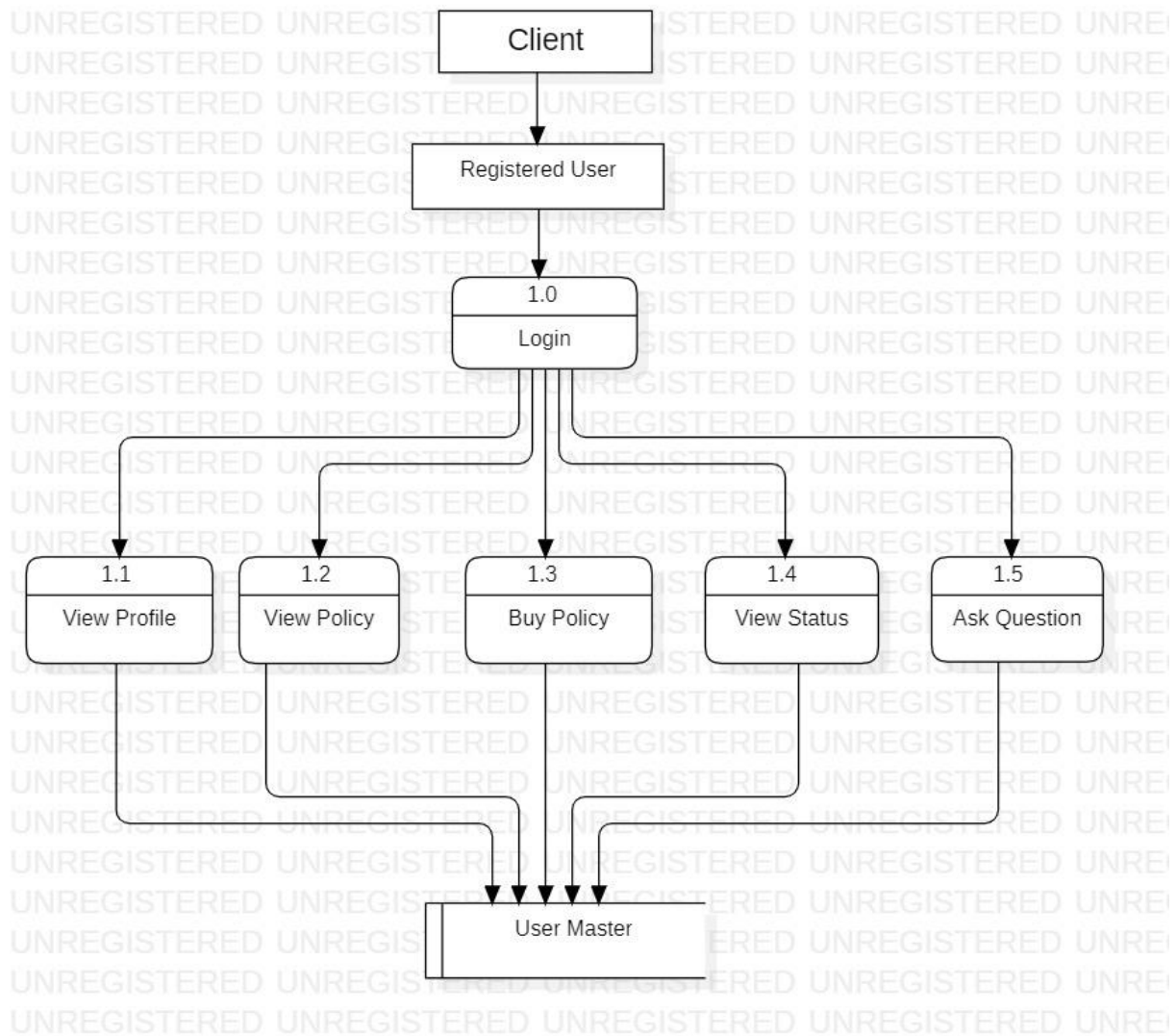
DFD Level 1
(Admin Panel)
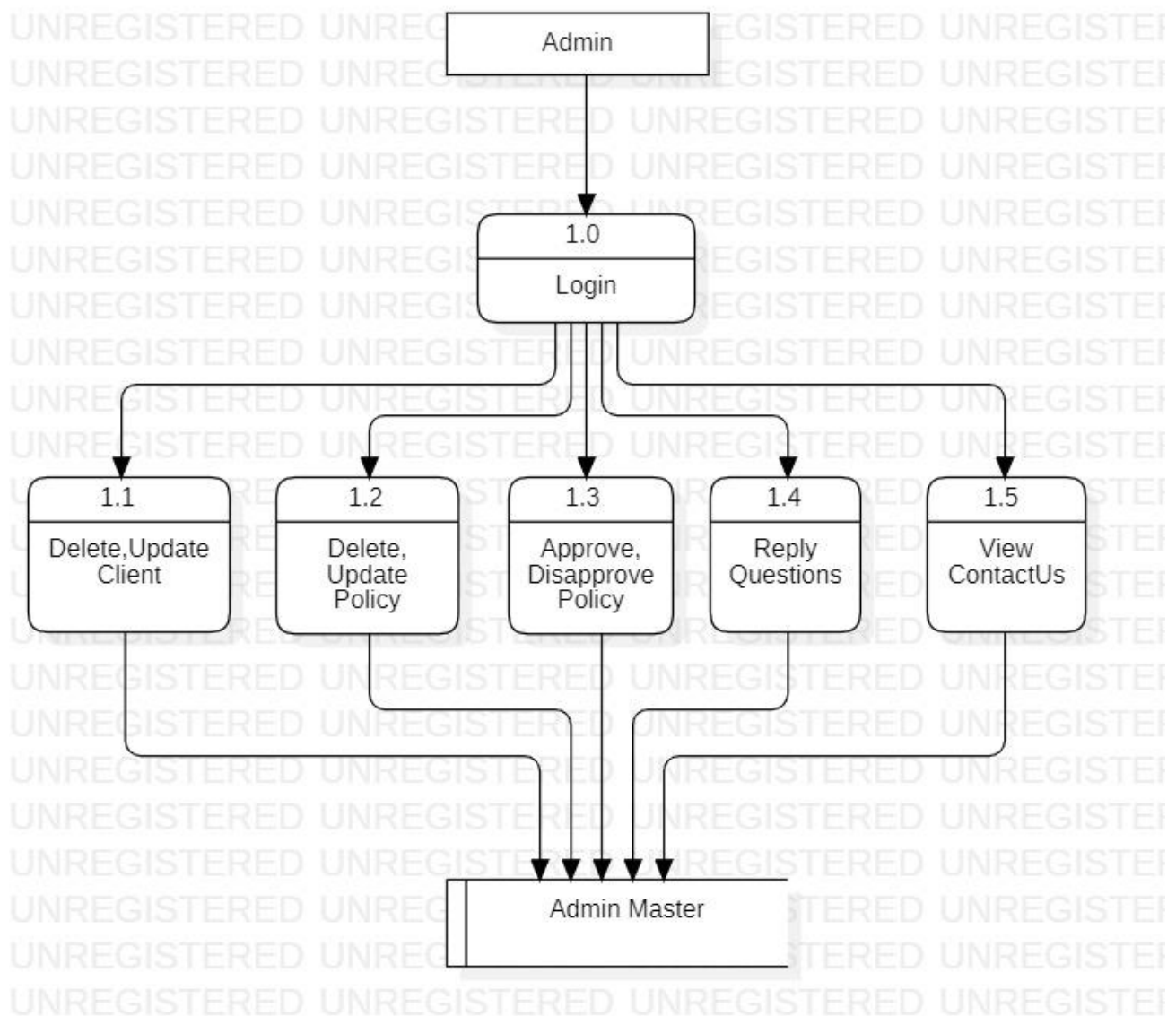
Flow Chart
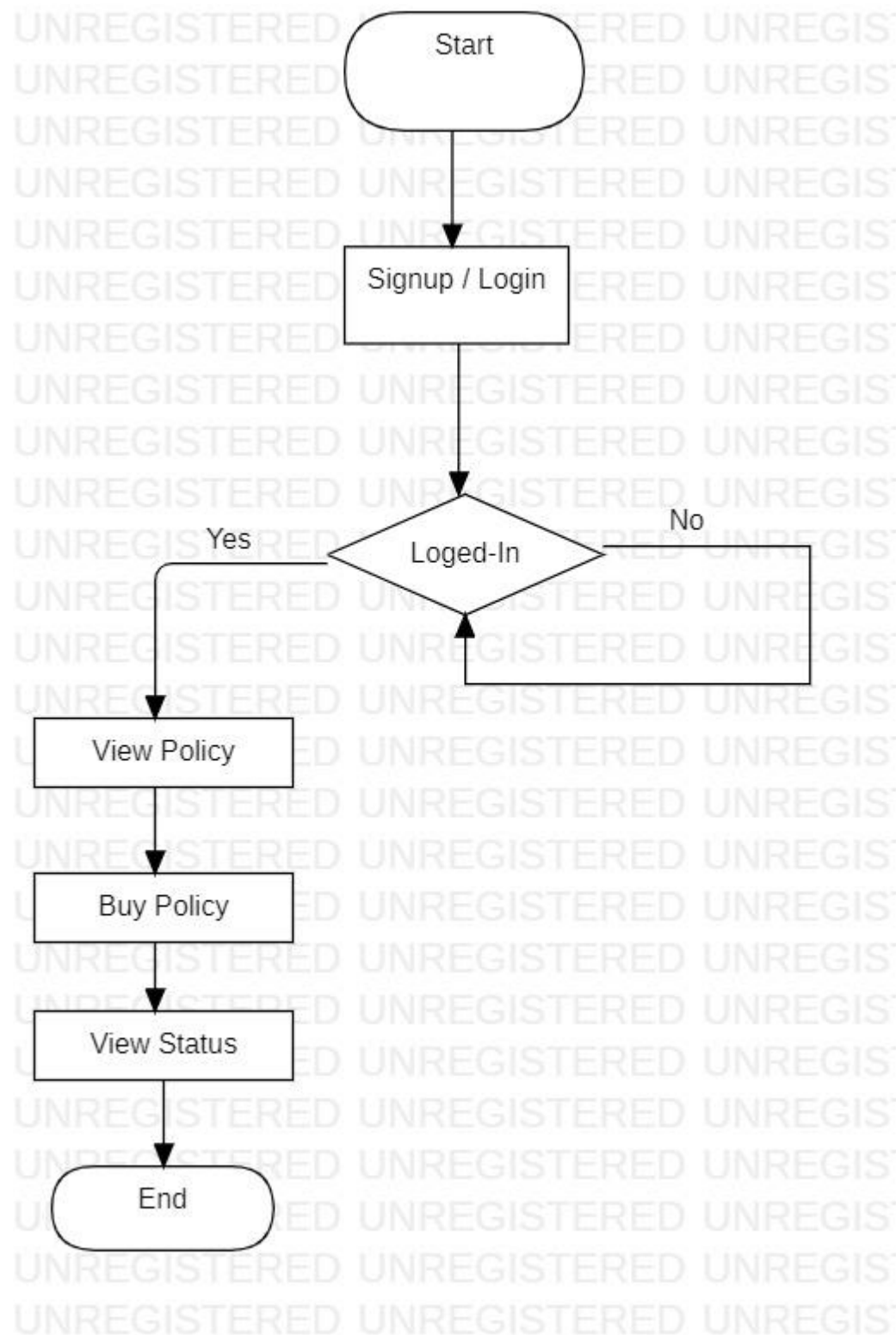
ER Diagram
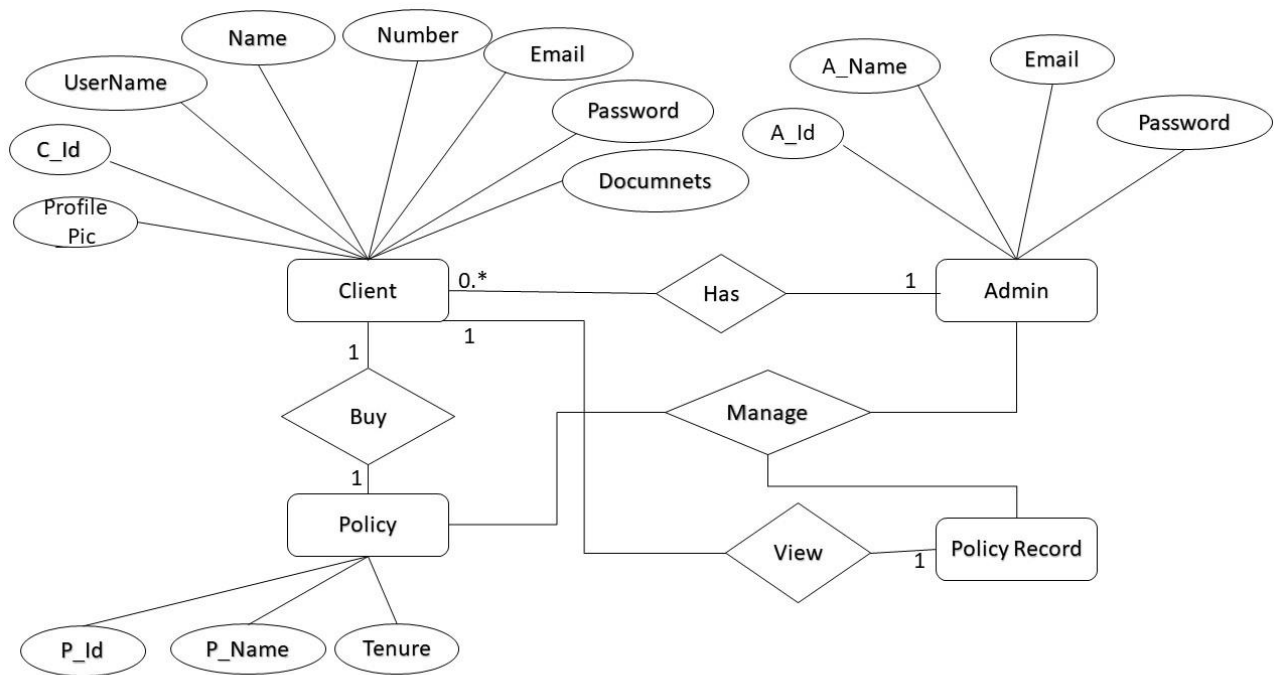
CHAPTER 7

IMPLEMENTATION

7.1 PROCESSING MODEL USED: Agile Model

The agile SDLC model is a combination of iterative and incremental process models with a focus on process adaptability and customer satisfaction by rapid delivery of working software products. Agile Methods break the product into small incremental builds. These builds are provided in iterations. Each iteration typically lasts from about one to three weeks. Every iteration involves cross-functional teams working simultaneously on various areas like-

- Planning
- Requirement Analysis
- Design
- Coding
- Unit Testing
- Acceptance Testing

The agile model believes that every project needs to be handled differently and the existing methods need to be tailored to best suit the project requirements. In Agile, the tasks are divided into time boxes (small time frames) to deliver specific features for a release.

An iterative approach is taken and a working software build is delivered after each iteration. Each build is incremental in terms of features; the final build holds all the features required by the customer.

The Agile thought process started early in software development and started becomingpopular with time due to its flexibility and adaptability.

CHAPTER 8

LIST OF TABLES

- **Customer Table**

FIELD	TYPE	NULL	KEY	EXTRA
Cid	int (30)	**NO**	PRI	**auto_increment**
Uname	varchar(20)	**NO**		
Name	varchar(25)	**NO**		
Phone	varchar(15)	**NO**		
Email	varchar(30)	**NO**		
Password	varchar(16)	**NO**		
Address	varchar(50)	**NO**		
Profile	ImageField	**NO**		
Documnets	FileField	NO		

- **Policy Category**

FIELD	TYPE	NULL	KEY	EXTRA
Cid	int (30)	**NO**	PRI	**auto_increment**
Cname	varchar(20)	**NO**		

- **Policy Table**

FIELD	TYPE	NULL	KEY	EXTRA
Pid	int (30)	**NO**	PRI	**auto_increment**
Pname	varchar(20)	**NO**		
Pcategory	varchar(30)	**NO**		
Sum**A**ssuracne	int(10)	**NO**		
Premium	int(5)	**NO**		
Tenure	Int(5)	**NO**		

- **Policy Record**

FIELD	TYPE	NULL	KEY	EXTRA
Rid	int (30)	**NO**	PRI	**auto_increment**
Cid	int(5)	**NO**		
Pid	int(5)	**NO**		
Status	varchar(30)	**NO**		

- **Contact Table**

FIELD	TYPE	NULL	KEY	EXTRA
Cid	int (30)	**NO**	PRI	**auto_increment**
Name	varchar(25)	**NO**		
Phone	varchar(15)	**NO**		
Email	varchar(30)	**NO**		
Message	Varchar(200)	**NO**		

DEMONSTRATION/SCREENSHOT

8.1 HOME PAGE

This page is the home page and it contains many pages like- (ContactUs, Aboutus, Login, and Signup). The user can access all pages, and if they want to Buy a policy, they must log in to the website.
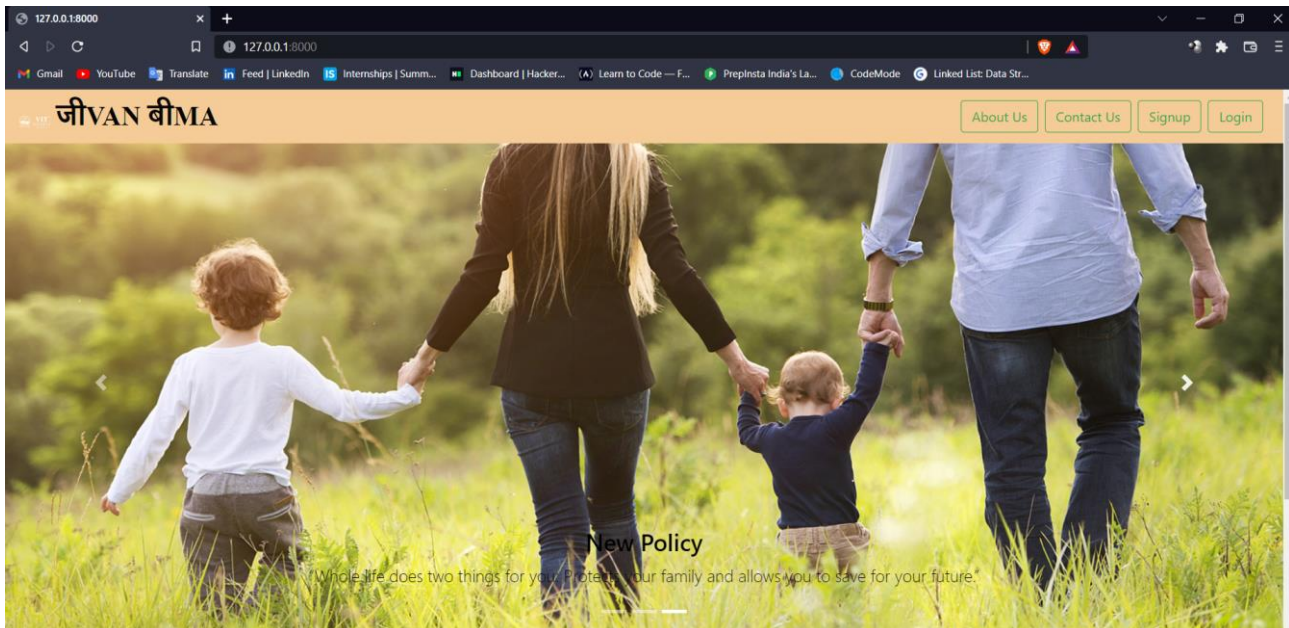
8.2 LOGIN PAGE

On this page, the client can log in to the website for Buy the Policy via username and password.
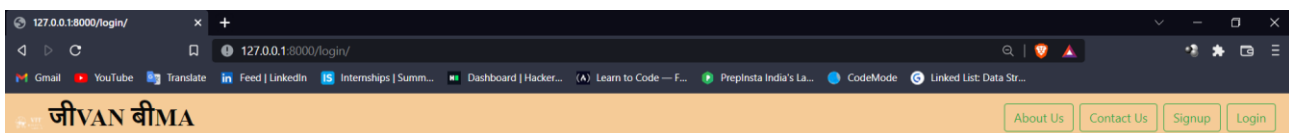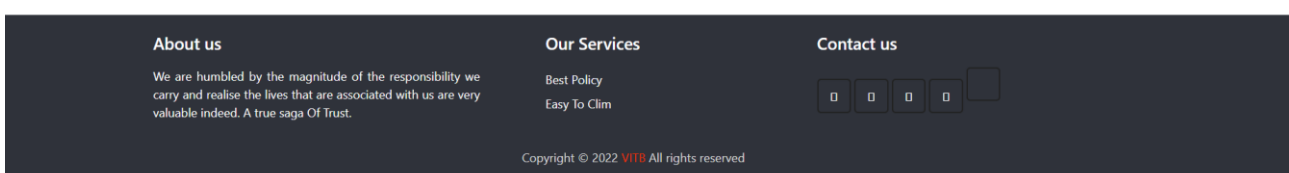
8.3 Signup

On the Signup page, the client can explore register on website fill the signup form

8.4 Client Dashboard

This is client dashboard, the client can see the all features and access it.

8.5 Admin Dashboard

This is admin dashboard, the admin can manage all things.
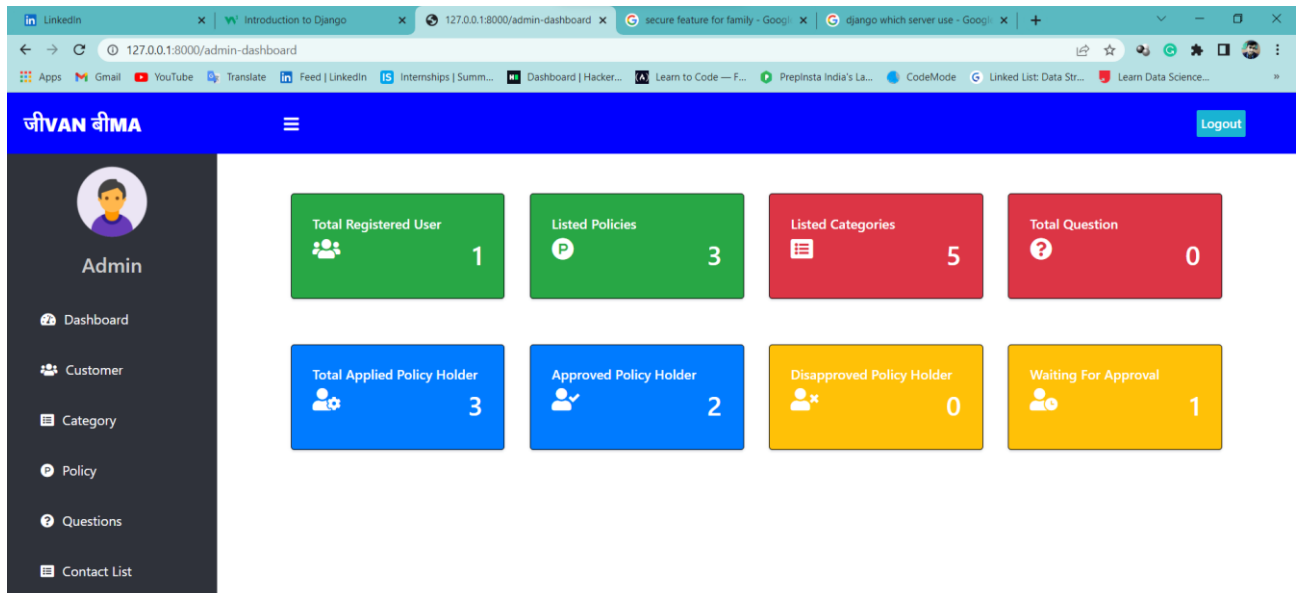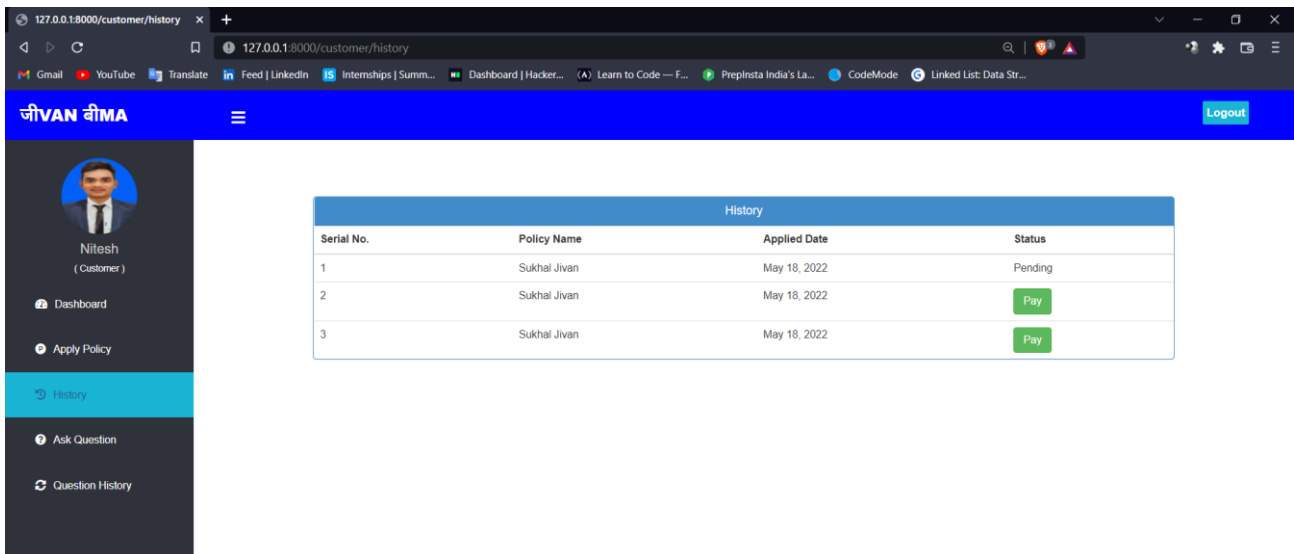
8.6 Policy

Here, the client can see the policy. Users can Apply the policy according to their convenience.
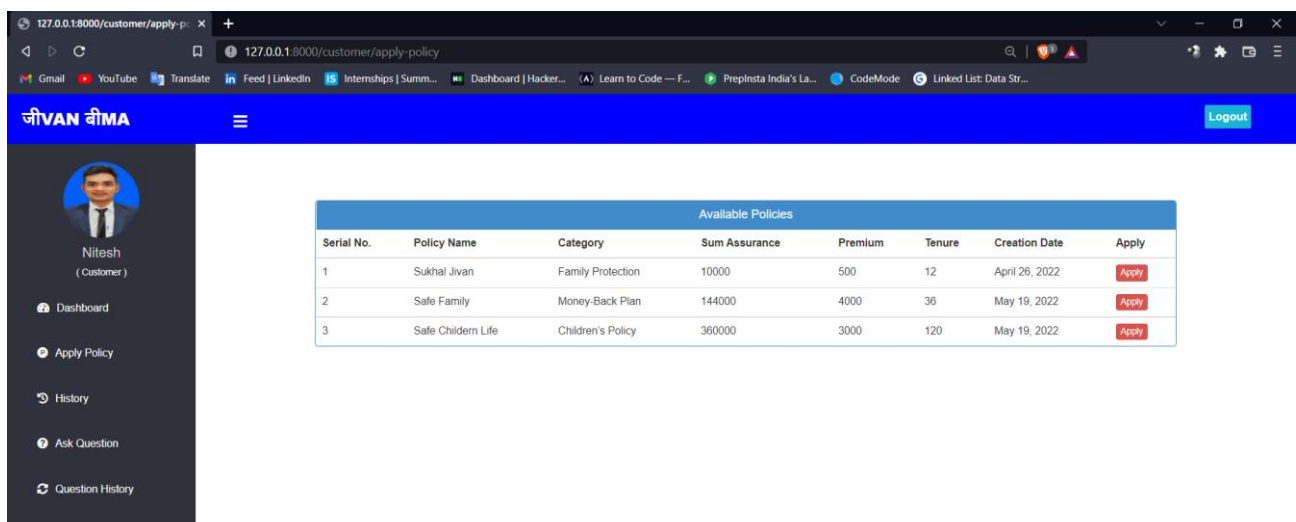
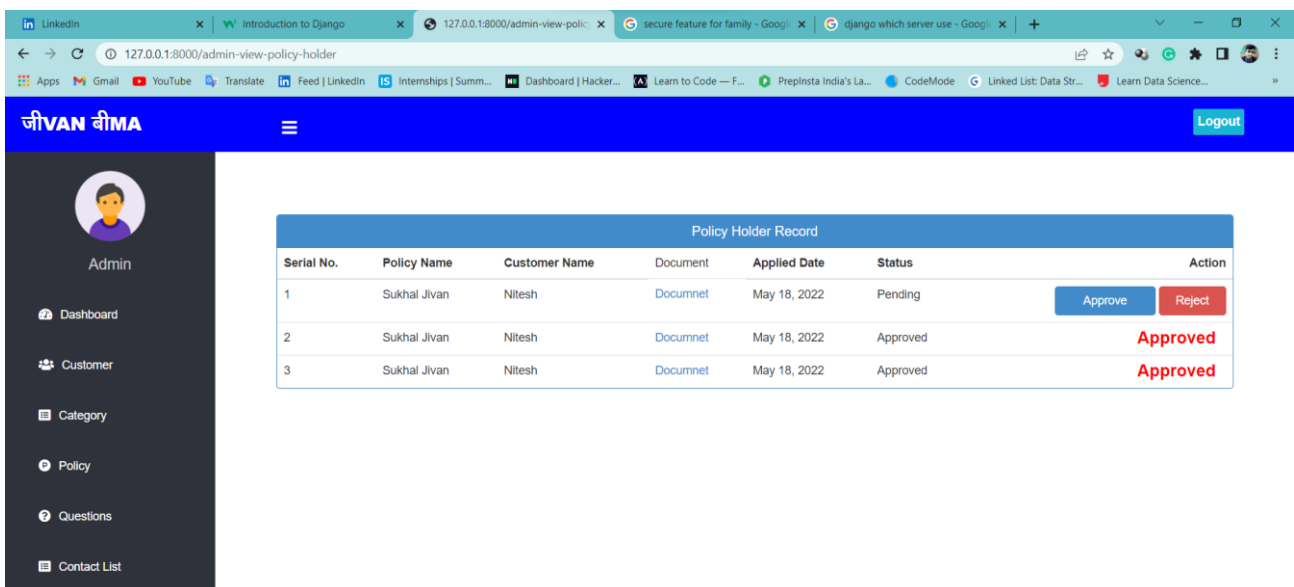## 8.7 History

On this page, client can see their policy history.

8.8 Admin Approve or Reject

On this page, admin can approve the policy or reject

8.9 CONTACT US

If the user has any doubts or queries regarding their trip, they can directly contact us with the help of this page.

CHAPTER 9

SYSTEM TESTING

Testing is the process of evaluating a system or its components with the intent to find whether it satisfies the specified requirements or not. This activity results in the actual, expected and the difference between their results i.e., testing is executing a system to identify any gaps, errors, or missing requirements contrary to the actual desire or requirements.

9.1 Testing strategies

To make sure that the system does not have any errors, the different levels of testing strategies that are applied at different phases of software development are

9.1.1 Unit testing

The goal of unit testing is to isolate each part of the program and show that individual parts are correct in terms of requirements and functionality.

9.1.2 Integration testing

The testing of combined parts of an application to determine if they function correctly together is Integration testing. This testing can be done by using two different methods.

Top-down integration testing

In Top-Down integration testing, the highest-level modules are tested first, and then progressively lower-level modules are tested.

Bottom-up integration testing

Testing can be performed starting from the smallest and lowest level modules and proceeding one at a time. When bottom-level modules are tested attention turns to those on the next level that use the lower-level ones they are tested individually and then linked with the previously examined lower-level modules. In a comprehensive software development environment, bottom-up testing is usually done first, followed by top-down testing.

CHAPTER 10

Conclusion And Future Scope

Life Insurance is currently recognized as a global industry that is growing at a high rate than any other industry. Access to relevant and accurate information is at the heart of Life Insurance. Here,the proposed project on Life Insurance Portal (JivanBima) tries to bridge the gap by noting whata Life Insurance as relevant. Hence, the aim of this project entails the design and implementation of a platform that will assist Life Insurance in gaining access to Policy. The project also helped to provide knowledge about the latest technology used in developing web-enabled applications and client-server technology that will be ingreat demand in the future.

It is worth mentioning that this project work is open for further enhancement, with the expectation that it becomes more robust and better enhanced; covering every single Life Insurance site. For a modified system, the user needs to just login into the website and can view policy, buy policy, pay premium and update profile.

Remaining Areas working on

- Working on Payment panel.

- Add report generating system.

CHAPTER_11

References

11.1   Book References

1. Hector Garcia-Molina, Jeffrey D. Ullman, Jennifer Widom, "DATABASE SYSTEMS The Complete Book", 2nd Edition, Upper Saddle River, NJ: Prentice-Hall, 2002.

2. Robin Nixon, "Learning Python, Django, JavaScript, CSS & HTML5: A Step-by-Step Guide to Creating Dynamic Websites", 3rd Edition, O′Reilly, 2014.

3. Laura Thomson and Luke Welling, "Django and SQLite Web Development", 4th Edition, Pearson Education, 2001.

4. Alok Ranjan, Abhilasha Sinha, Ranjit Battewad, "JavaScript for Modern Web Development: Building a Web Application Using HTML, CSS, and JavaScript", 1st Edition, BPB Publications, 2020.

5. Paul DuBois, "SQLite", 1st Edition, Pearson Education, 2008.

11.2   Website References

1. https://www.w3schools.com/css/css_navbar.asp

2. https://www.geeksforgeeks.org/javascript-events/?ref=lbp

3. https://www.tutorialspoint.com/html/html_formatting.htm

4. https://www.freecodecamp.org/news/tag/web-development/

5. https://stackoverflow.com/questions/tagged/frontend

6. https://www.youtube.com/playlist?list=PLbGui_ZYuhigchy8DTw4pX4duTTpvqlh6