FULL STACK PROJECT REPORT

**On**

**“Health Insurance Company”**

**Submitted by**

**Anuj Agrawal**

**(191500136)**

**Anshika Jain**

**(191500131)**

**Piyash Rastogi**

**(191500537)**

**Piyush Sahu**

**(191500546)**

**Tanu Singh Jadaun**

**(191500846)**

Department of Computer Engineering & Applications

**Institute of Engineering & Technology**



**GLA University**

**Mathura- 281406, INDIA**

**2021-2022**

**Department of computer Engineering and Applications**

**GLA University, Mathura**

**17 km. Stone NH#2, Mathura-Delhi Road, P.O. – Chaumuha,**

**Mathura – 281406**



**Declaration**

We hereby declare that the work which is being presented in the Mini Project “**Health Insurance Company”,** in partial fulfillment of the requirements for Mini Project , is an authentic record of our own work carried by the team members under the supervision of our mentor Mr. Manoj Varshney.

Group Members:

Anuj Agrawal (191500136)

Anshika Jain (191500131)

Piyash Rastogi (191500537)

Piyush Sahu (191500546)

Tanu Singh Jadaun (191500846)

Course: B.Tech (Computer Science and Engineering)

Year: 3rd

Semester: 5th

## Supervised By:

Mr. Manoj Varshney, Assistant Professor,

GLA University, Department of Computer Engineering & Application

**Department of computer Engineering and Applications**

**GLA University, Mathura**

**17 km. Stone NH#2, Mathura-Delhi Road, P.O. – Chaumuha,**

**Mathura – 281406**



**Certificate**

This is to certify that the above statements made by the candidates are correct to the best of my/our knowledge and belief.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Supervisor

Mr. Manoj Varshney

Technical Trainer

Dept of CEA, GLA University

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Project Coordinator Program Coordinator

(Mr. Mayank Srivastava) (Mr. Shashi Shekar)

**About the Project**

Our Mini project “Health Insurance Company” is an online website. Health Insurance Company (HIC) pays for some or all of the cost of the health services you receive, like doctor’s visit, hospital stays, and visits to the emergency room. It helps keep your health care costs predictable and affordable. Health Insurance provides financial protection in case you have a serious accident or illness. In India it is nothing but a burden of inefficiency of a government run system. Moreover, the uncontrolled and no innovative attitude of Indian bureaucracy always argued against the private players in the health insurance sector in India .The first part gives an insight about Health insurance and its various aspects, the Company Profile, Objectives of the study, Research Methodology. One can have a brief knowledge about share market and its basics through the Project. In this Project The data collected has been well organized and presented. I hope the research findings and conclusion will be of use

**Motivation**

Having health insurance is important because coverage helps people get timely medical care and improves their lives and health. Some may believe that people always have access to medical care because they can always go to an emergency room. But even areas with well supported safetynet care do not remove barriers to access to the same extent as does having health insurance. “Coverage matters,” concluded the Institute of Medicine (IOM) during a recent multiyear appraisal.1 Indeed, the prestigious IOM estimated that lack of coverage was associated with about 18,000 extra deaths per year among uninsured adults.2 Several points deserve emphasis.

**Requirements**

**a). Software Requirements:**

* Technology Implemented: Full Stack Web Development
* Languages/Technologies Used: MERN Stack
* IDE Used: Visual Studio Code
* Web Browser: Google Chrome
* GitHub: GitHub is a code hosting platform for version control and collaboration. It lets you and others work together on projects from anywhere. GitHub Repository: A GitHub repository can be used to store a development project. It can contain folders and any type of files (HTML, CSS, JavaScript, Documents, Data, Images). A GitHub repository should also include a license file and a README file about the project. A GitHub repository can also be used to store ideas, or any resources that you want to share.
* Visual Studio Code: Visual Studio Code is a free source-code editor made by Microsoft for Windows, Linux and macOS. [7] Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality. Microsoft has released Visual Studio Code's source code on the VS Code repository of GitHub.com, under the permissive MIT License, while the compiled binaries are freeware.

**b). Hardware Requirements:**

* Processor Required: Intel i3
* Operating System: Windows 10
* RAM: 8GB
* Hardware Devices: Computer System
* Hard Disk: 256GB

**Acknowledgement**

We thank the almighty for giving us the courage and perseverance in completing the project. This project itself is an acknowledgement for all those people who have given us their heartfelt co-operation in making this project a grand success. We extend our sincere thanks to Mr. Manoj varshney, Assistant Professor at “GLA University, Mathura” for providing his valuable guidance at every stage of this project work. We are profoundly grateful towards the unmatched services rendered by him. And last but not least, we would like to express our deep sense of gratitude and earnest thanks giving to our dear parents for their moral support and heartfelt cooperation in doing the main project.

**Health Insurance Company**

**Abstract**

As the name suggests, our project is all about a Health Insurance Policy , Health insurance is one of the ways that people in various countries finance their medical needs. It is estimated that out-of-pocket expenditure of over 15–20 % of total health expenditure or 40 % of household net income of subsistence needs can lead to financial catastrophe. When people on low incomes with no financial risk protection fall ill, they face a dilemma: they can use health services and suffer further impoverishment in paying for them, or they can forego services, remain ill, and risk being unable to work or function. Variation in financing and organization structures in various countries notwithstanding, there is now nearly a unanimous commitment to assuring universal access to medically necessary care in high-income countries. Internationally, health insurance serves to improve service utilization and protect households against impoverishment from out-of-pocket expenditures. Analysis of how health insurance schemes function in a particular country, especially in relation to other funding aspects and health outcomes, can provide a glimpse of the performance of the whole healthcare system.

**Contents**

**Acknowledgment…………………………………….........**08

**Abstract…………………………………………………**09

1. **Introduction:**

Introduction to MERN Stack........…......12-13

Pre-requisites…………………………..14

1. **Technologies Used:**

Angular………………………………………15

HTML/CSS……………………......................16

Typescript ………………………………..............17-18

Firebase.....................................................................19

**3. List of Figures...……………………………………**20-26

**4. Software Testing..............................................................**27-31

**5.Conclusion……………………………………………**32

**6.Bibliography………………………………………….**33

**Chapter 1**

**Introduction**

Today Developers around the world are making efforts to enhance user experience of using application as well as to enhance the developer’s workflow of designing applications to deliver projects and rollout change requests under strict timeline. Stacks can be used to build web applications in the shortest span of time. The stacks used in web development are basically the response of software engineers to current demands. They have essentially adopted pre-existing frameworks (including JavaScript) to make their lives easier. While there are many, MEAN and MERN are just two of the popular stacks that have evolved out of JavaScript. Both stacks are made up of open-source components and offer an end-to-end framework for building comprehensive web apps that enable browsers to connect with databases. The common theme between the two is JavaScript and this is also the key benefit of using either stack. One can basically avoid any syntax errors or any confusion by just coding in one programming language, JavaScript. Another advantage of building web projects with MERN is the fact that one can benefit from its enhanced flexibility. In order to understand MERN stack, we need to understand the four components that make up the MERN namely

**Chapter 2**

**Technologies Used**

**Angular**

Angular is a development platform, built on [TypeScript](https://www.typescriptlang.org/). As a platform, Angular includes:

* A component-based framework for building scalable web applications
* A collection of well-integrated libraries that cover a wide variety of features, including routing, forms management, client-server communication, and more
* A suite of developer tools to help you develop, build, test, and update your code

With Angular, you're taking advantage of a platform that can scale from single-developer projects to enterprise-level applications. Angular is designed to make updating as straightforward as possible, so take advantage of the latest developments with a minimum of effort. Best of all, the Angular ecosystem consists of a diverse group of over 1.7 million developers, library authors, and content creators.

## Angular applications: The essentials

This section explains the core ideas behind Angular. Understanding these ideas can help you design and build your applications more effectively.

### **Components**

Components are the building blocks that compose an application. A component includes a TypeScript class with a @[Component](https://angular.io/api/core/Component)() decorator, an HTML template, and styles. The @[Component](https://angular.io/api/core/Component)() decorator specifies the following Angular-specific information:

* A CSS selector that defines how the component is used in a template. HTML elements in your template that match this selector become instances of the component.
* An HTML template that instructs Angular how to render the component.
* An optional set of CSS styles that define the appearance of the template's HTML elements.
* Angular's component model offers strong encapsulation and an intuitive application structure. Components also make your application painless to unit test and can improve the overall readability of your code.
* For more information on what to do with components, see the [Components](https://angular.io/guide/component-overview) section.

### **Templates**

* Every component has an HTML template that declares how that component renders. You define this template either inline or by file path.
* Angular extends HTML with additional syntax that lets you insert dynamic values from your component. Angular automatically updates the rendered DOM when your component’s state changes.

**HTML**

HTML is the standard markup language for creating Web pages.

* HTML stands for Hyper Text Markup Language
* HTML is the standard markup language for creating Web pages
* HTML describes the structure of a Web page
* HTML consists of a series of elements
* HTML elements tell the browser how to display the content
* HTML elements label pieces of content such as "this is a heading", "this is a paragraph", "this is a link", etc

**Advantages :** 

* HTML is widely used.
* Every browser supports HTML Language.
* Easy to learn and use.
* HTML is light weighted and fast to load.
* Do not get to purchase any extra software because it’s by default in every window.
* Easy to use
* Loose syntax (although, being too flexible won’t suit standards).
* HTML is easy enough to write
* HTML is that it is easy to code even for novice programmers.
* HTML also allows the utilization of templates, which makes designing a webpage easy.
* Very useful for beginners in the web designing field.
* HTML can be supported to each and every browser, if not supported to all the browsers.
* HTML is built on almost every website, if not all websites.
* HTML is increasingly used for data storage as like XML syntax.
* Free – You need not buy any software.
* HTML is present in every window by default so you not need to buy the software which cost too much.
* HTML has many tag and attributes which can short your line of code.

**Disadvantages :** 

* It cannot produce dynamic output alone, since it’s a static language.
* Making the structure of HTML documents becomes tough to understand.
* Errors can be costly.
* It is the time consuming as the time it consume to maintain on the colour scheme of a page and to make lists, tables and forms.
* It can create only static and plain pages so if we’d like dynamic pages then HTML isn’t useful.
* Required to write a lot of code for just creating a simple webpage.
* We have to check up the deprecated tags and confirm not to use them to appear because another language that works with HTML has replaced the first work of the tag, and hence the opposite language needs to be understood and learned.
* Security features offered by HTML are limited.
* If we need to write down long code for creating a webpage then it produces some complexity.
* HTML can create only static and plain pages so if we’d like dynamic pages then HTML isn’t useful.
* I need to write down tons of code for creating an easy webpage.
* Security features are not good at HTML.
* If we’d like to write down long code for creating a webpage then it produces some complexity.

**CSS**

**Cascading Style Sheets**, fondly referred to as **CSS**, is a simple design language intended to simplify the process of making web pages presentable.

**CSS** is a MUST for students and working professionals to become a great Software Engineer specially when they are working in Web Development Domain. I will list down some of the key advantages of learning CSS:

* **Create Stunning Web site** - CSS handles the look and feel part of a web page. Using CSS, you can control the color of the text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colors are used, layout designs,variations in display for different devices and screen sizes as well as a variety of other effects.
* **Become a web designer** - If you want to start a carrer as a professional web designer, HTML and CSS designing is a must skill.
* **Control web** - CSS is easy to learn and understand but it provides powerful control over the presentation of an HTML document. Most commonly, CSS is combined with the markup languages HTML or XHTML.
* **Learn other languages** - Once you understands the basic of HTML and CSS then other related technologies like javascript, php, or angular are become easier to understand.

**TypeScript**

TypeScript stands in an unusual relationship to JavaScript. TypeScript offers all of JavaScript’s features, and an additional layer on top of these: TypeScript’s type system.

For example, JavaScript provides language primitives like string and number, but it doesn’t check that you’ve consistently assigned these. TypeScript does.

This means that your existing working JavaScript code is also TypeScript code. The main benefit of TypeScript is that it can highlight unexpected behavior in your code, lowering the chance of bugs.

TypeScript lets you write JavaScript the way you really want to. TypeScript is a typed superset of JavaScript that compiles to plain JavaScript. TypeScript is pure object oriented with classes, interfaces and statically typed like C# or Java. The popular JavaScript framework **Angular 2.0** is written in TypeScript. Mastering TypeScript can help programmers to write object-oriented programs and have them compiled to JavaScript, both on server side and client side.

The Type System represents the different types of values supported by the language. The Type System checks the validity of the supplied values, before they are stored or manipulated by the program. This ensures that the code behaves as expected. The Type System further allows for richer code hinting and automated documentation too.

TypeScript provides data types as a part of its optional Type System.

**Firebase**

Firebase tutorial is designed for both beginners and professionals. Our tutorial provides all the basic and advanced services knowledge, such as Real-time Database, Cloud Messaging, Hosting and Crash Reporting, etc.

Firebase is a Backend-as-a-Service, and it is a real-time database which is basically designed for mobile applications. This tutorial is designed in such a way that we can easily understand or can perform the service of Firebase in a very efficient way.

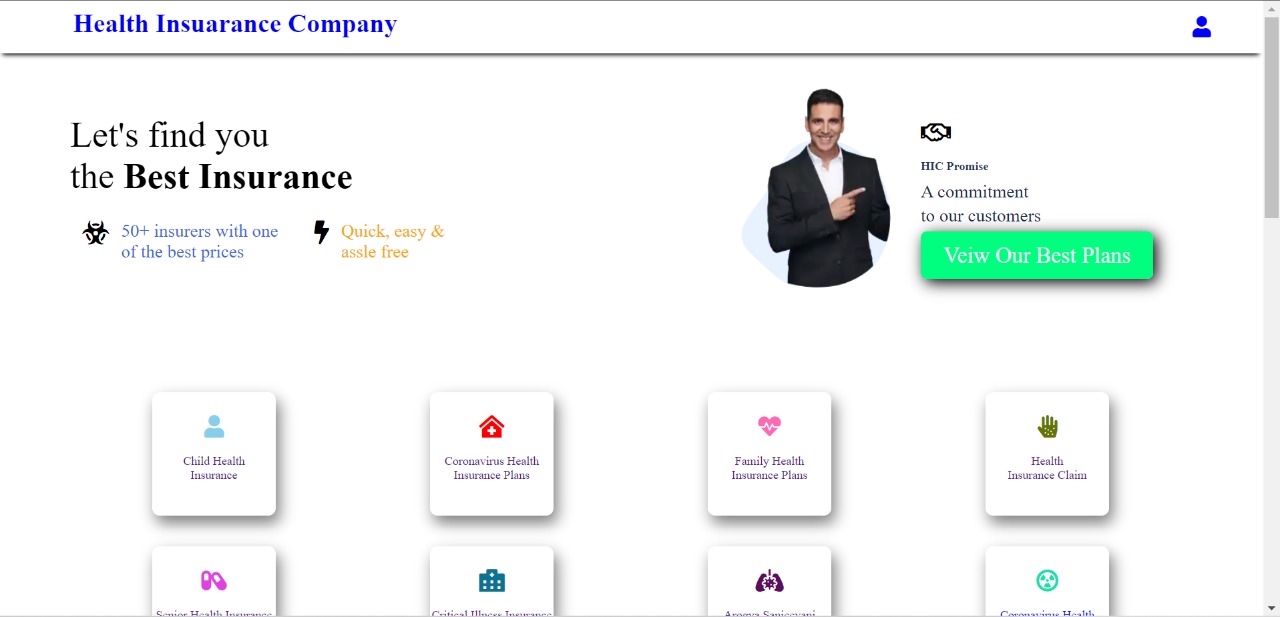
In the era of rapid prototyping, we can get bright ideas, but sometimes they are not applicable if they take too much work. Often, the back-end is the limiting factor - many considerations never apply to server-side coding due to lack of knowledge or time.

Firebase is a Backend-as-a-Service(BaaS) which started as a YC11 startup. It grew up into a next-generation app-development platform on Google Cloud Platform. Firebase (a NoSQL JSON database) is a real-time database that allows storing a list of objects in the form of a tree. We can synchronize data between different devices.

**Chapter 3**

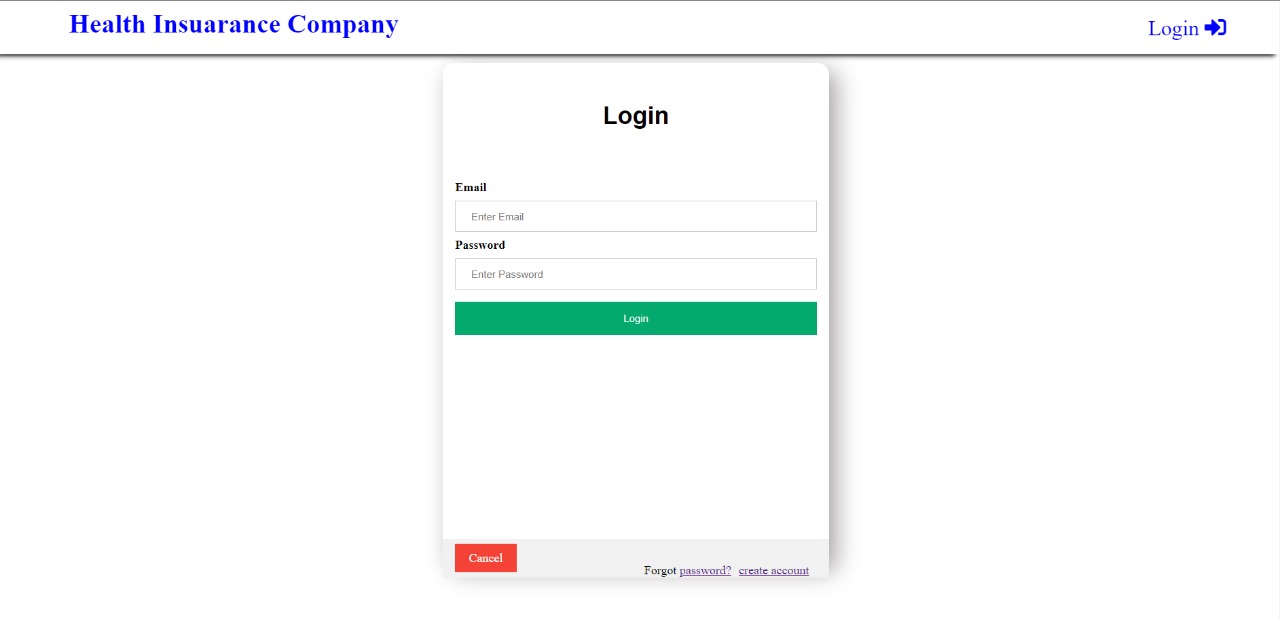
**List of Figures**

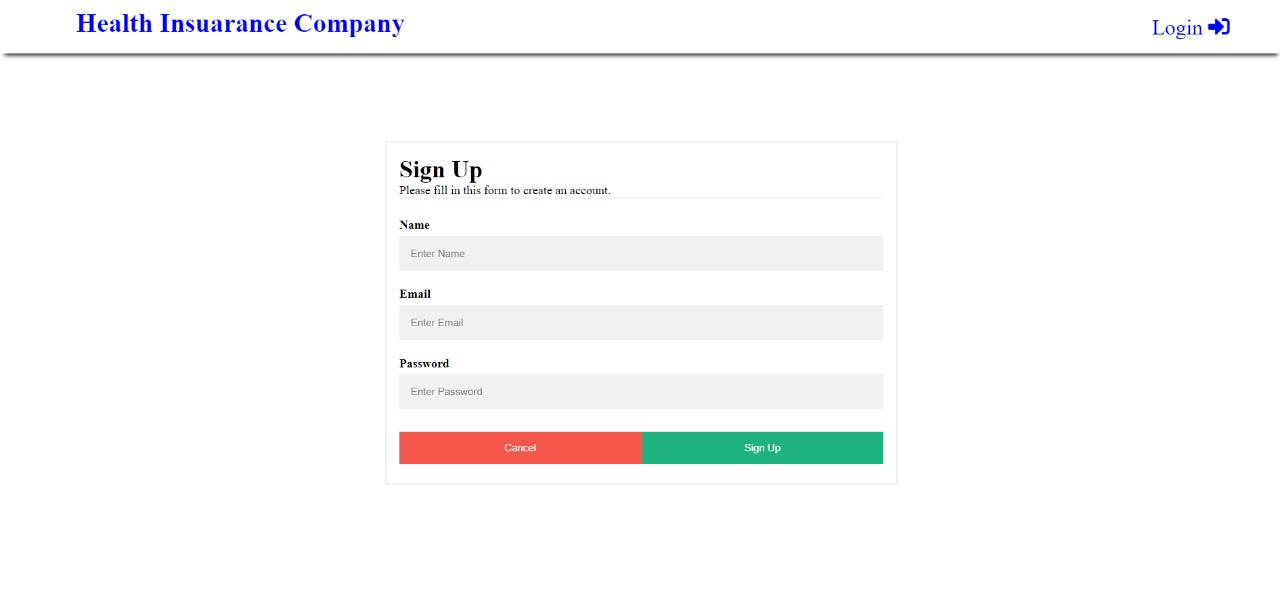
**1. Home Page**



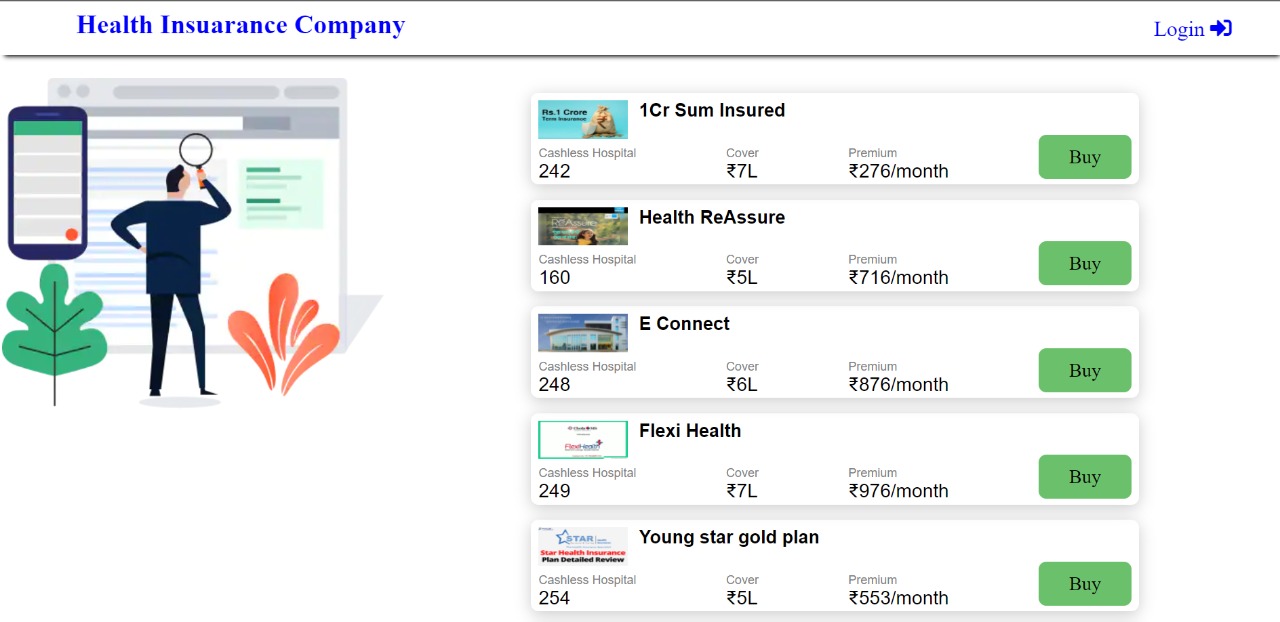


**2. Login Page**



**3. SignUp Page**

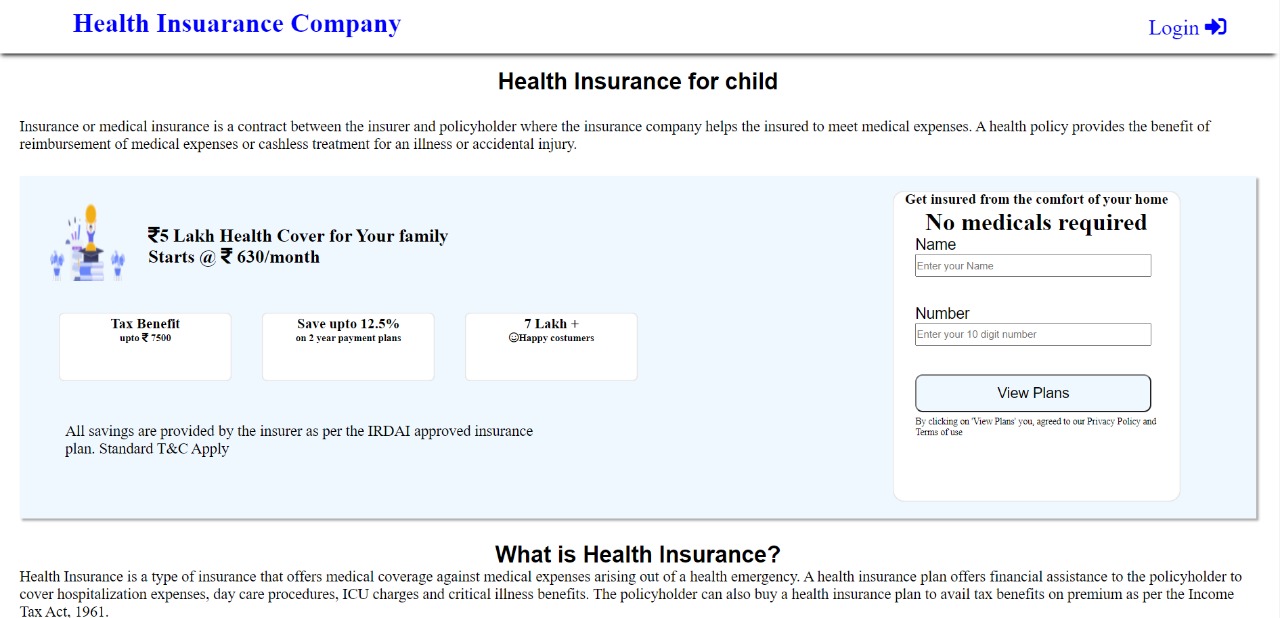
**4. Plans Page**



**5. General Form Plan Page**

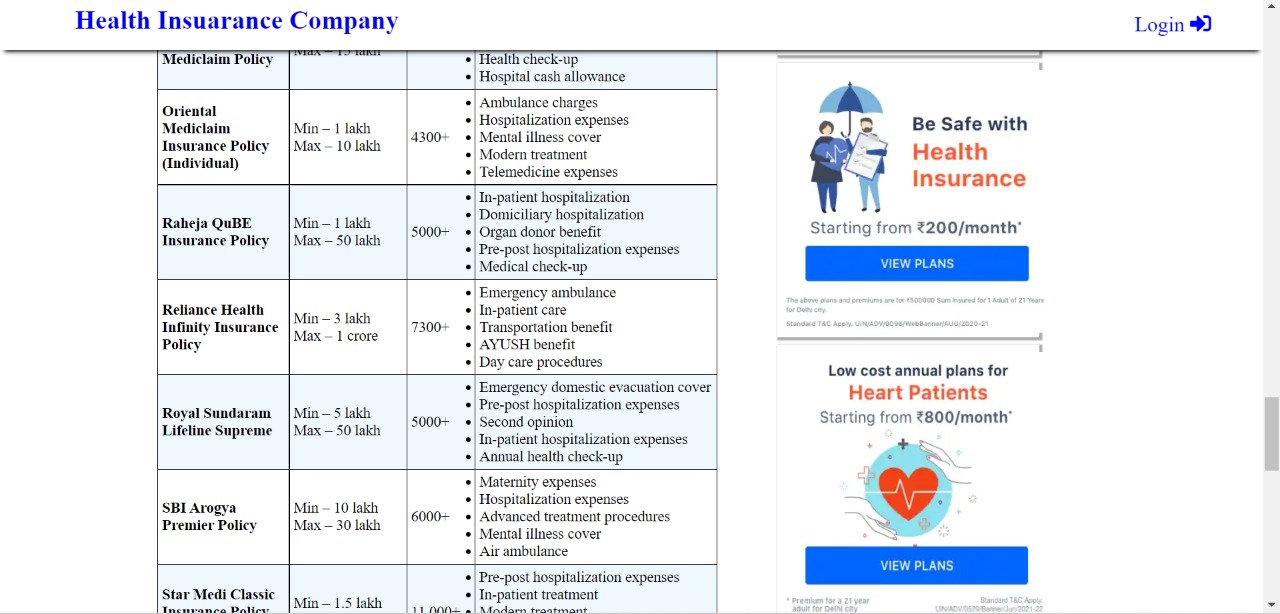


**6. Health Insurance For Child Page**

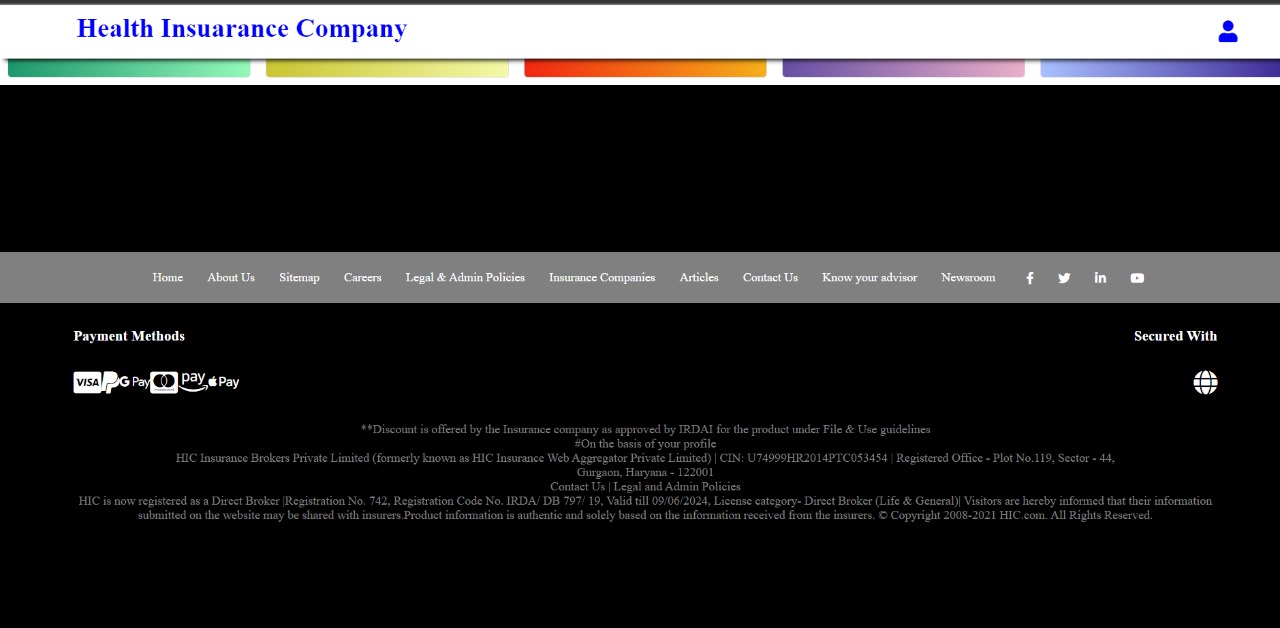








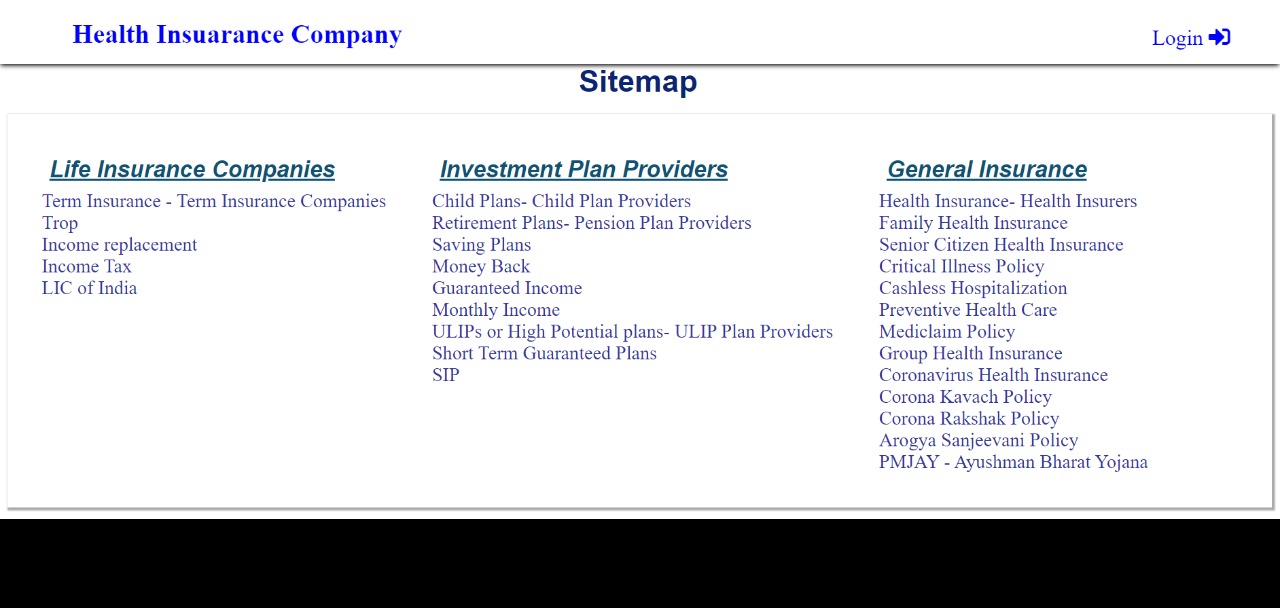
**7.Footer Page**



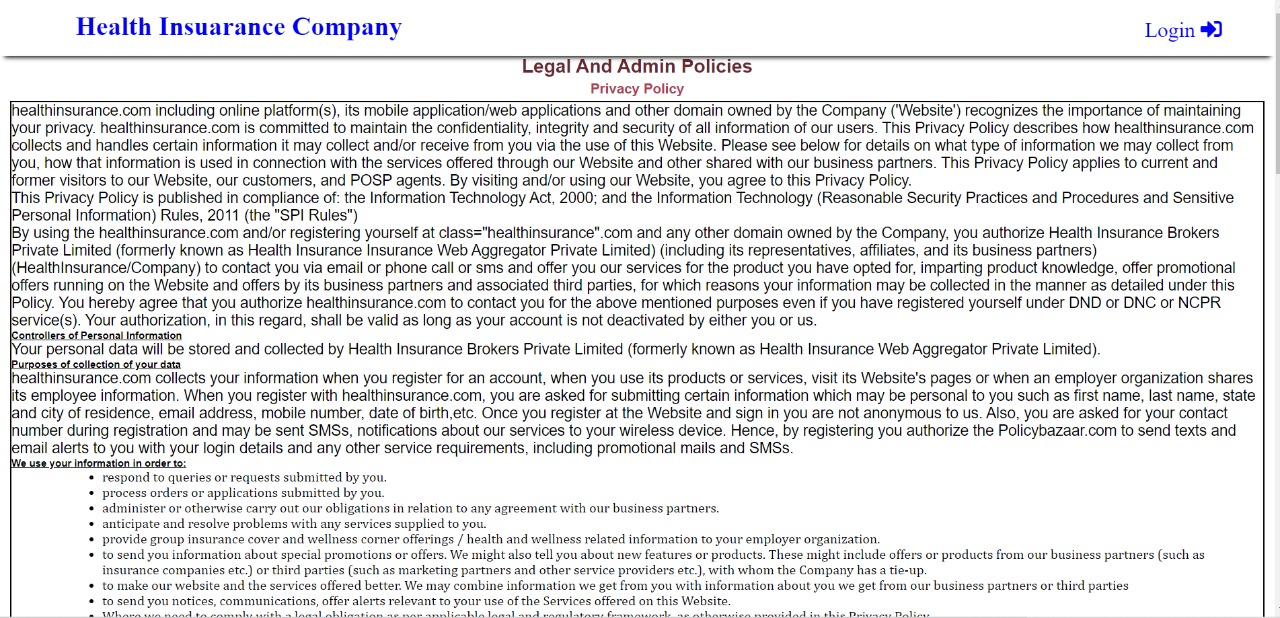
**8.About Page**



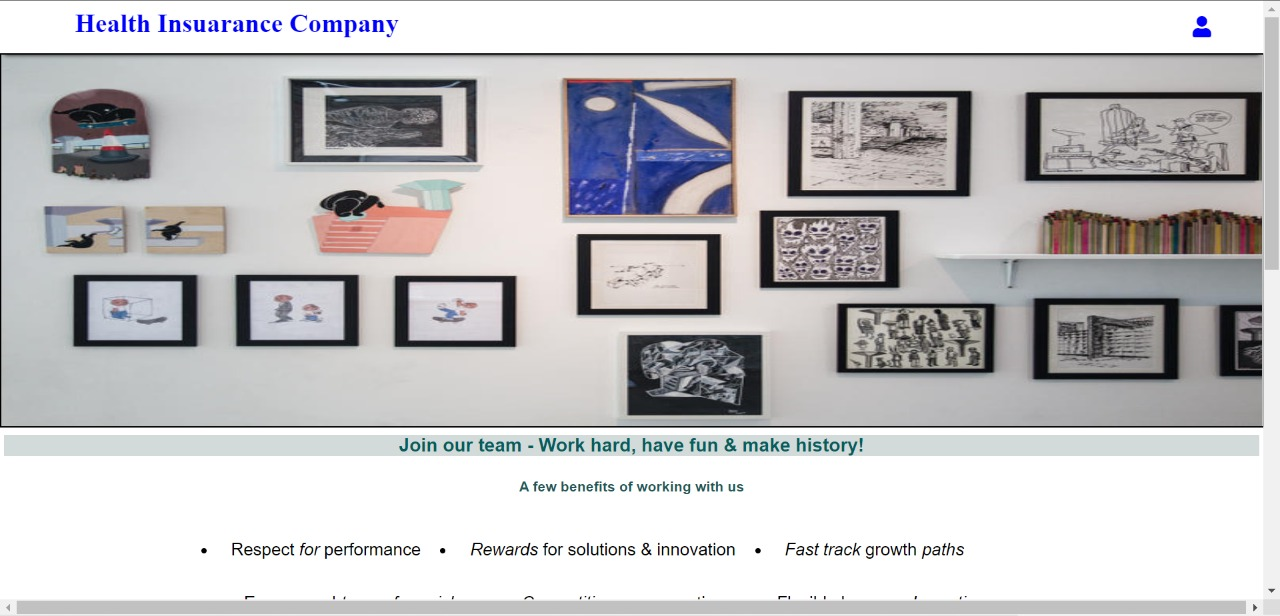
**9.Sitemap page**



**10.Legal and Admin Policy Page**



**11.Career page**



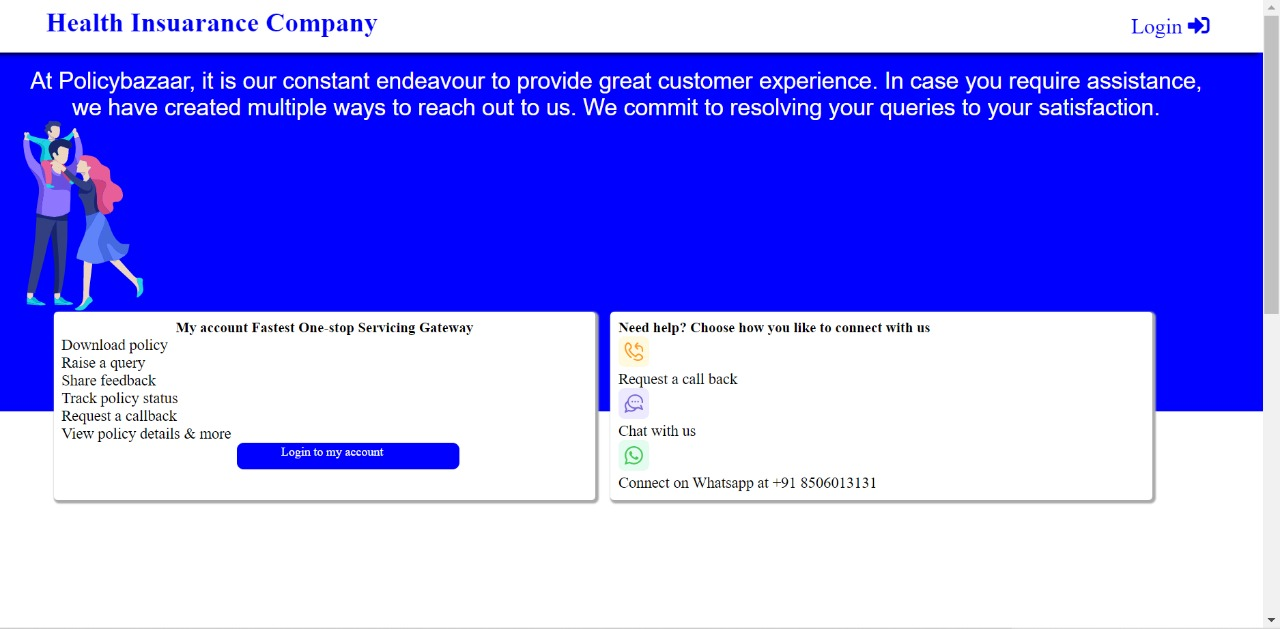
**12.Insurance Companies page**



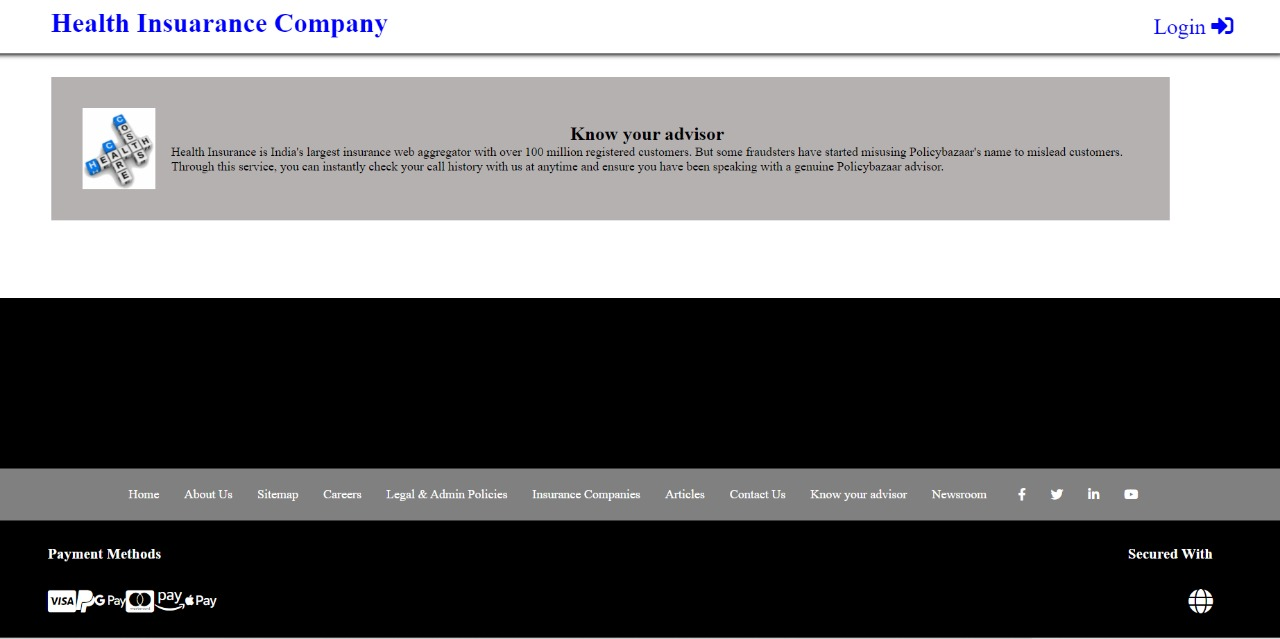
**13.Article Page**



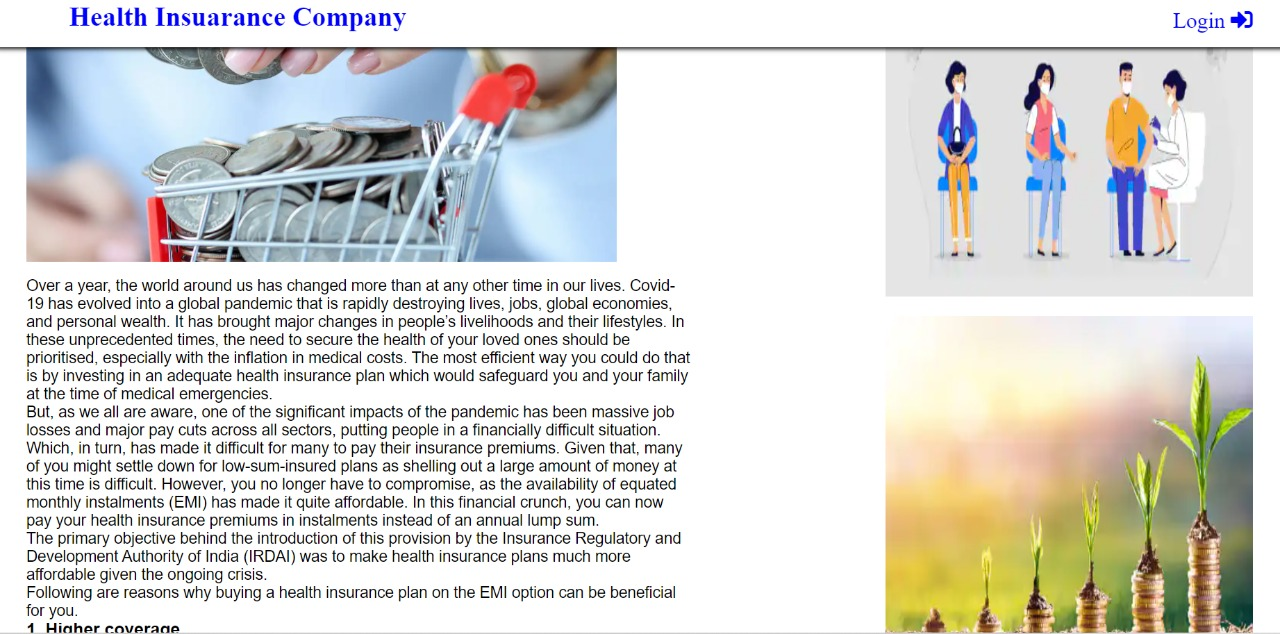
**14. Contact us Page**



**15. Know Your Advisor page**



**16.Newsroom Page**



**Chapter 4**

**Software Testing**

Once source code has been generated, software must be tested to uncover as many errors as possible before delivery. It is very important to work the system successfully and achieve high quality of software. Testing include designing a series of test cases that have a high likelihood of finding errors by applying software-testing techniques. System testing makes logical assumptions that if all the parts of the system are correct, the goal will be successfully achieved. The system should be checked logically. Validations and cross checks should be there. Avoid duplications of record that cause redundancy of data. In other Words, Testing is the process of evaluating a system or its component(s) with the intent to find whether it satisfies the specified requirements or not. It is executing a system in order to identify any gaps, errors, or missing requirements in contrary to the actual requirements.

The preliminary goal of implementation is to write source code and internal documentation so that conformance of the code to its specifications can be easily verified, and so that debugging, testing and modifications are eased.. Simplicity, clarity and elegance are the hallmark of good programs, obscurity, cleverness, and complexity are indications of inadequate design and misdirected thinking. Source code clarity is enhanced by structured coding techniques, by good coding style, by, appropriate supporting documents, by good internal comments, and by feature provided in modern programming languages. The implementation team should be provided with a well-defined set of software requirement, an architectural design specification, and a detailed design description. Each team member must understand the objectives of implementation.

4.1 TERMINOLOGY

Error The term error is used in two ways. It refers to the difference between the actual output of software and the correct output, in this interpretation, error is essential a measure of the difference between actual and ideal. Error is also to used to refer to human action that result in software containing a defect or fault.

Fault is a condition that causes to fail in performing its required function. A fault is a basic reason for software malfunction and is synonymous with the commonly used term Bug.

Failure is the inability of a system or component to perform a required function according to its specifications. A software failure occurs if the behavior of the software is the different from the specified behavior. Failure may be caused due to functional or performance reasons.

4.2 TYPES OF TESTING

**a. Unit Testing:** The term unit testing comprises the sets of tests performed by an individual programmer prior to integration of the unit into a larger system. A program unit is usually small enough that the programmer who developed it can test it in great detail, and certainly in greater detail than will be possible when the unit is integrated into an evolving software product. In the unit testing the programs are tested separately, independent of each other. Since the check is done at the program level, it is also called program teasing.

**b. Module Testing:** A module and encapsulates related component. So can be tested without other system module.

**c. Subsystem Testing:** Subsystem testing may be independently design and implemented common problems are sub-system interface mistake in this checking we concent on it. There are four categories of tests that a programmer will typically perform on a program unit.

i Functional test

ii Performance test

iii Stress test

iv Structure test

**Functional Test:** Functional test cases involve exercising the code with Nominal input values for which expected results are known; as well as boundary values (minimum values, maximum values and values on and just outside the functional boundaries) and special values.

**Performance Test:** Performance testing determines the amount of execution time spent in various parts of the unit, program throughput, response time, and device utilization by the program unit. A certain amount of avoid expending too much effort on fine-tuning of a program unit that contributes little to the overall performance of the entire system. Performance testing is most productive at the subsystem and system levels.

**Stress Test** Stress test are those designed to intentionally break the unit. A great deal can be learned about the strengths and limitations of a program by examining the manner in which a program unit breaks.

**Structure Test** Structure tests are concerned with exercising the internal logic of a program and traversing particular execution paths. Some authors refer collectively to functional performance and stress testing as “black box” testing. While structure testing is referred to as “white box” or “glass box” testing. The major activities in structural testing are deciding which path to exercise, deriving test date to exercise those paths, determining the test coverage criterion to be used, executing the test, and measuring the test coverage achieved when the test cases are exercised.

**Chapter 5**

**Conclusion**

We have completed our project within time limit with the coordination of our team members under the supervision of our mentor Mr. Manoj Varshney.

Our project repository is available at

<https://github.com/piyashrastogi-1/HIC>

**Chapter 6**

**Bibliography**

[**www.google.com**](http://www.google.com)

**www.geeksforgeeks.org**

[**www.youtube.com**](http://www.youtube.com)

**www.w3schools.com**

**www.beta-labs.in**