**Name of the Project** – Online Insurance Application

**Objectives/Vision**

Develop a web application where users can opt for an insurance policy based on their needs.

This application is a kind of insurance platform where a number of insurance policy is available for different users.

**Functional Requirements**

1. Application should have a home page with information about insurance and various insurance policy.
2. Complete registration process: We need to develop user registration, login, and forgotten passwords. Reset password and forgot password should implement with the registered email id of the user.
3. Once the user provides the correct login credential, the user will lend into the user dashboard. The dashboard should have all necessary features available to the specific user.
4. User type varies from the user, admin, and application owner.
5. Every user will have their own dashboard and respective features that can be used for specific operations.
6. User Dashboard: Can apply for an insurance policy and provide information like age, city, medical history, tobacco user, alcohol, etc.

Can see some add-on features which is available based on policy selection.

1. Discount: Discounts and offers can also be seen under the dashboard.
2. Proper cart should be implemented and all discounts and offers can be applied to the cart feature.
3. Admin dashboard: Can register as one of the insurance companies and add an insurance policy.
4. Application owner: Can provide some useful information to policy buyers and insurance companies.
5. Application owner can see the feedback coming from the user and insurance companies.

**Non-Function Requirement:**

1. Build and Test Responsive and Interactive Webpages
2. Proper authentication should be implemented using OAuth 2.0 / JWT.
3. Application should have proper validation implementation with all required forms.
4. Appropriate cloud service should be incorporated / can be used for deployment, developing features based on project or use case.
5. Proper front-end unit testing should be part of the implementation.
6. Junit test coverage should also be part of development.
7. Implement proper CI and CD pipeline.
8. Deploy application on AWS cloud. (like Deploy Java Microservices on Amazon ECS using AWS Fargate).

**Tools and Technologies to be used**

- VCS: Gitlab

- Front End: HTML, CSS, JavaScript, Bootstrap and REACT.

- Backend: Java 8, Spring Boot.

- Database: MySQL/MongoDB

- Testing: Mocha/Chai, Junit.

-Deployment: AWS Cloud services