

Insurance Policy Management

Group-4

Team Members

1. Vanshita Mittal	-	2530686
2. Mansi Bhagat	-	2527346
3. Mohammad Mustafa	-	2529993
4. Singuluri Bhargavi	-	2529929
5. Samyuktha N	-	2530276
6. Praveen A	-	2530190
7. Santhosh P	-	2528958
8. Ruthrapriya S	-	2530060
9. Mohamed Kaja	-	2527330
10. Aravind Soundarajan	-	2528220



OBJECTIVE

Create a dynamic and responsive web application for managing the insurance policies application online



BACKGROUND OF THE PROBLEM STATEMENT

- # Insurance will help the families in maintaining their standard of living in case if we are not around in the future.**

- # Insurance is all about providing financial safety to the individual or Vehicles.**

- # It helps us to enjoy financial security.**

- # If we have a proper application to access all the Insurance related activity,then it will be cheery on top.**

INTRODUCTION

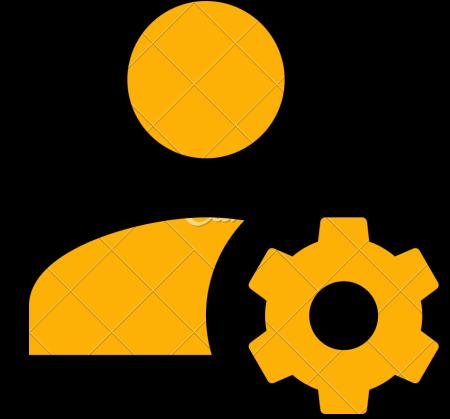


- # We have created a bug-free application for the admin and the user with the main objective to build a secured, robust Insurance Management System where the policies are managed systematically

 - # This web application maintains the record of customer history and policies efficiently so that it would be easily accessible.
- 

ADMIN MODULE

- # Login into portal
- # View customers
- # View/add/update/delete policy
- # View/add/update/delete policy category
- # Approve/disapprove policy applied by customer
- # View all customers and their policy statements
- # View queries of customer

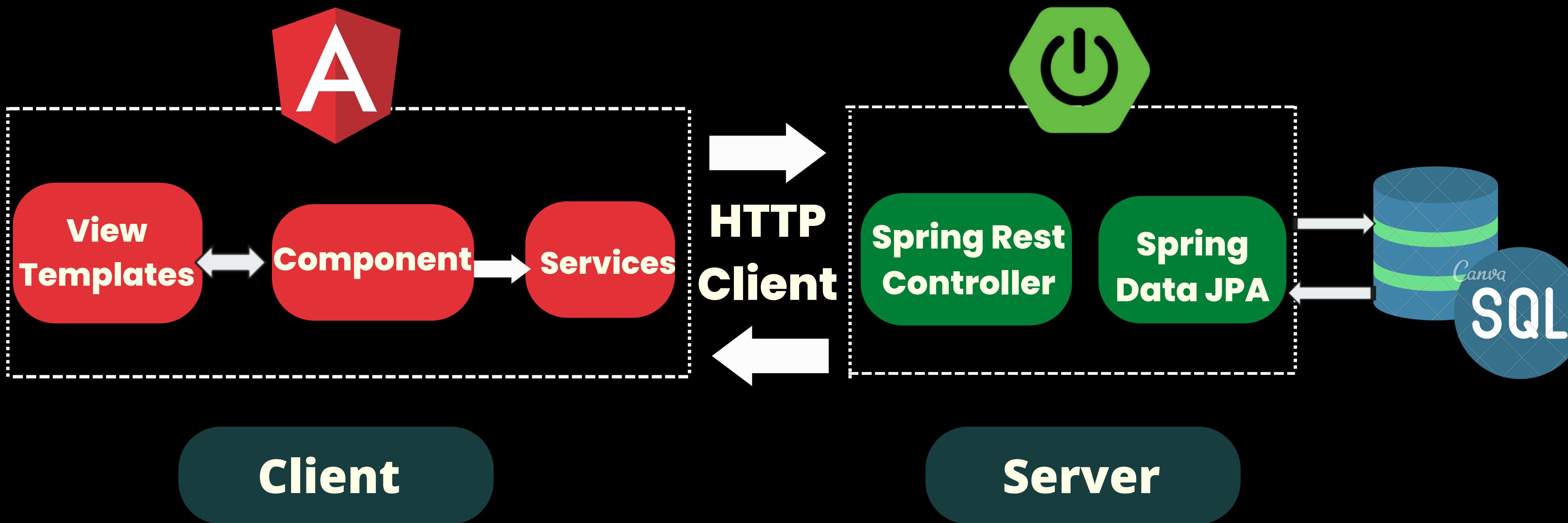


USER MODULE

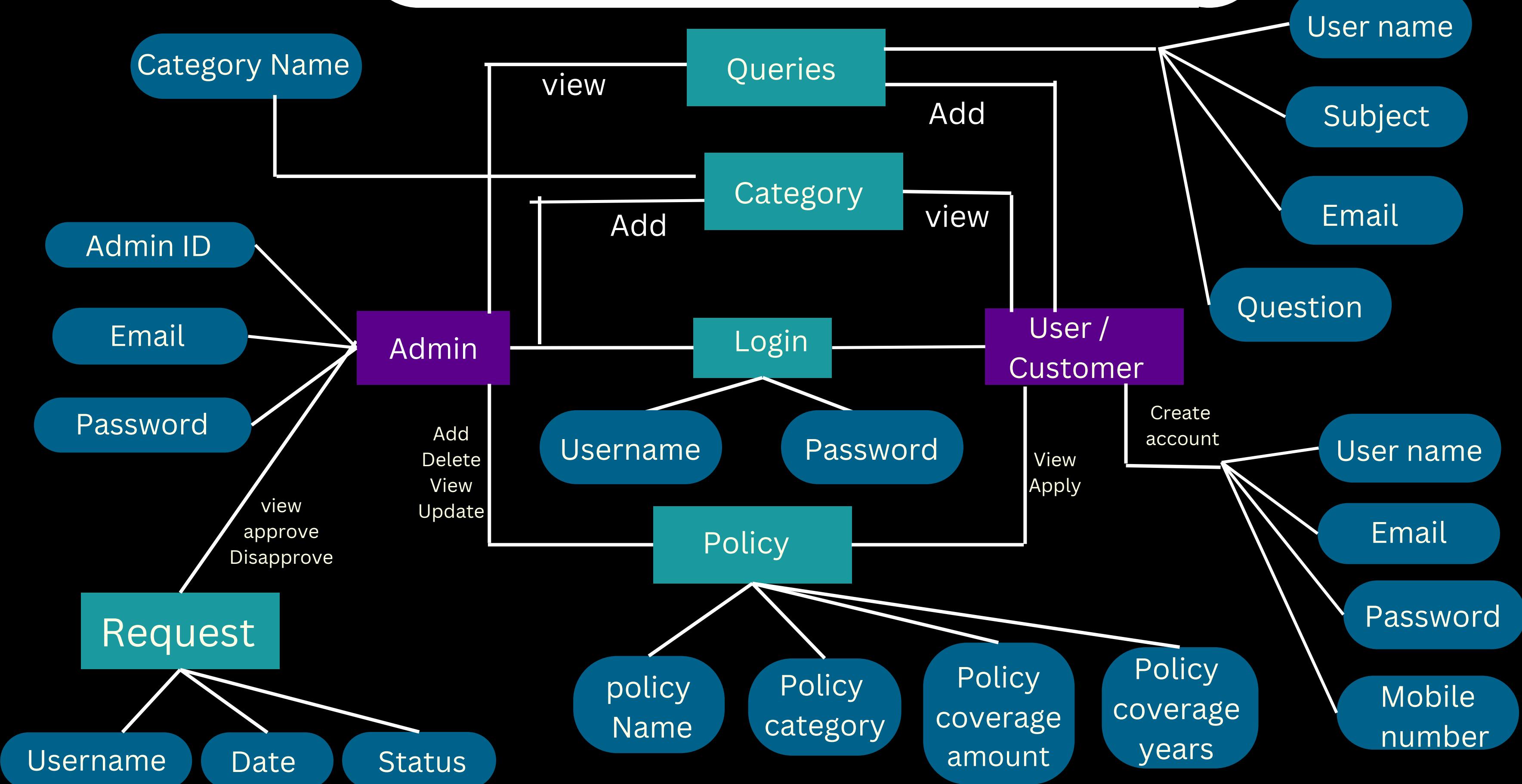
- # Create account and login
- # Update personal details
- # View all policies
- # Apply for any policy
- # View policy history
- # Post queries
- # Contact US



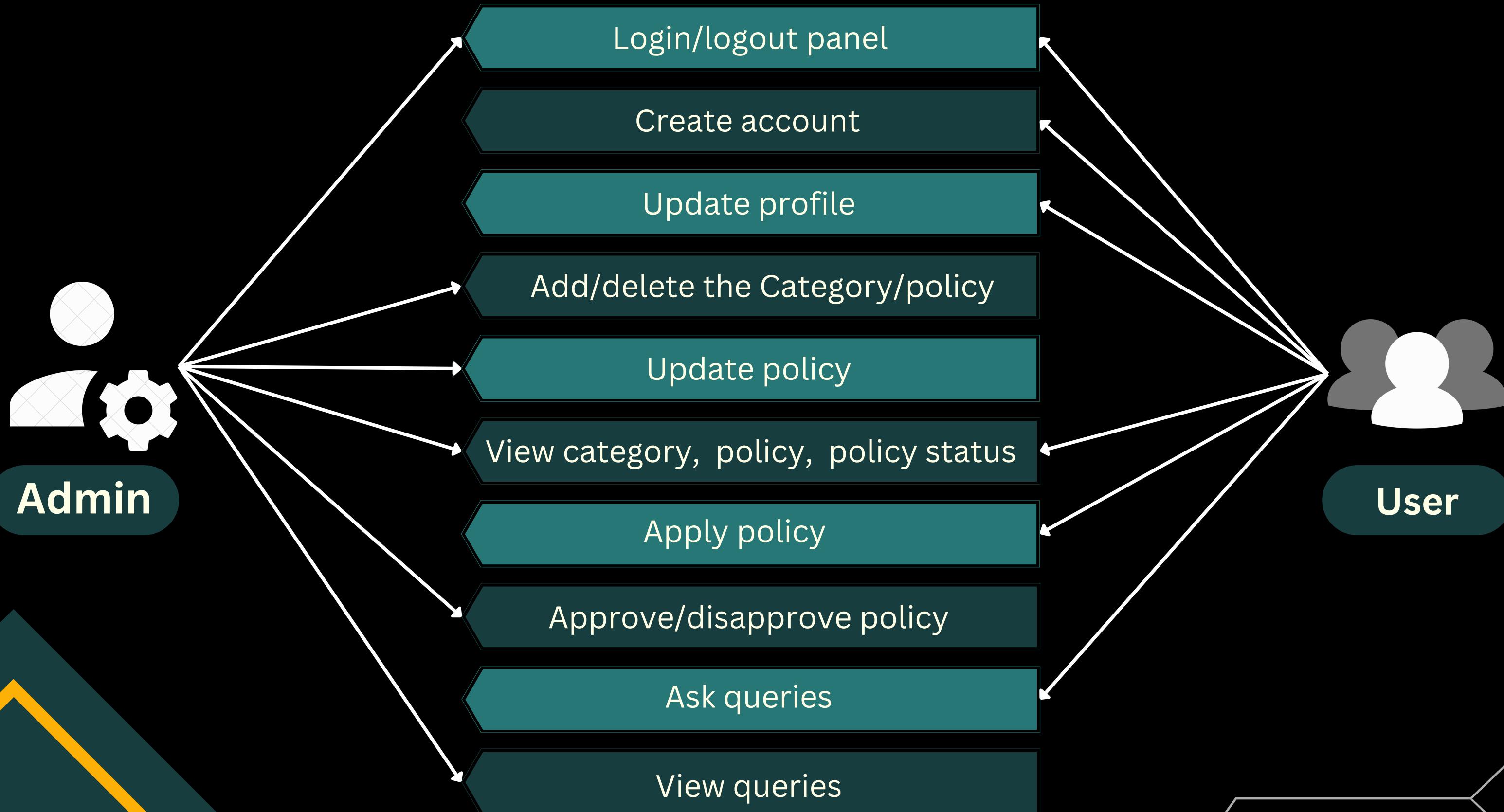
ARCHITECTURE



ER DIAGRAM



USE CASE DIAGRAM



Tools and Technologies

FRONTEND



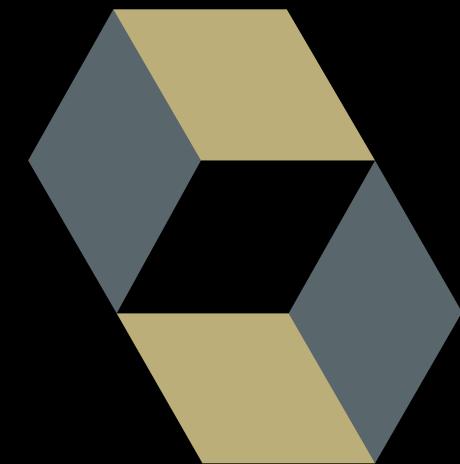
ANGULAR

BACKEND

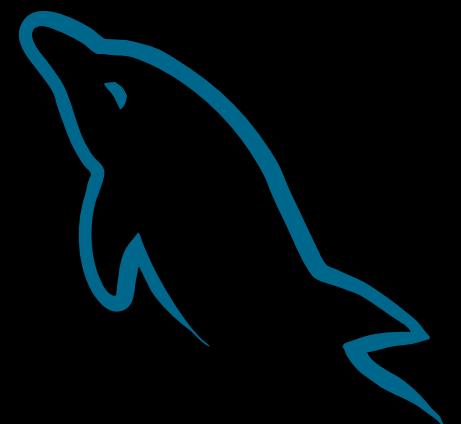


spring boot

DATABASE



HIBERNATE



MySQL

SERVER



Apache Tomcat

Technology Stack

ANGULAR



Angular is an open-source, JavaScript framework written in TypeScript. Google maintains it, and its primary purpose is to develop single-page applications.

SPRINGBOOT



Spring Boot is a Java framework, built on top of the Spring, used for developing web applications. It allows you to create REST APIs with minimal configurations

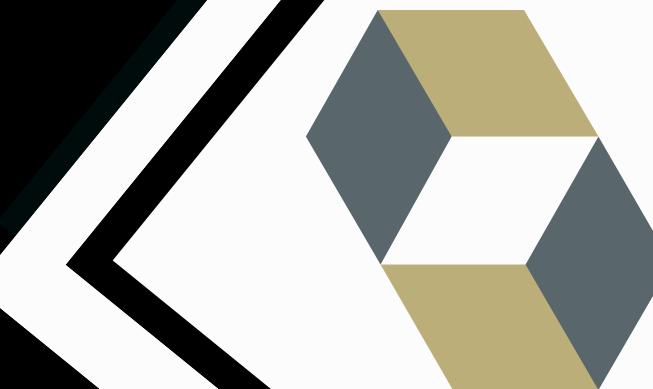
Technology Stack

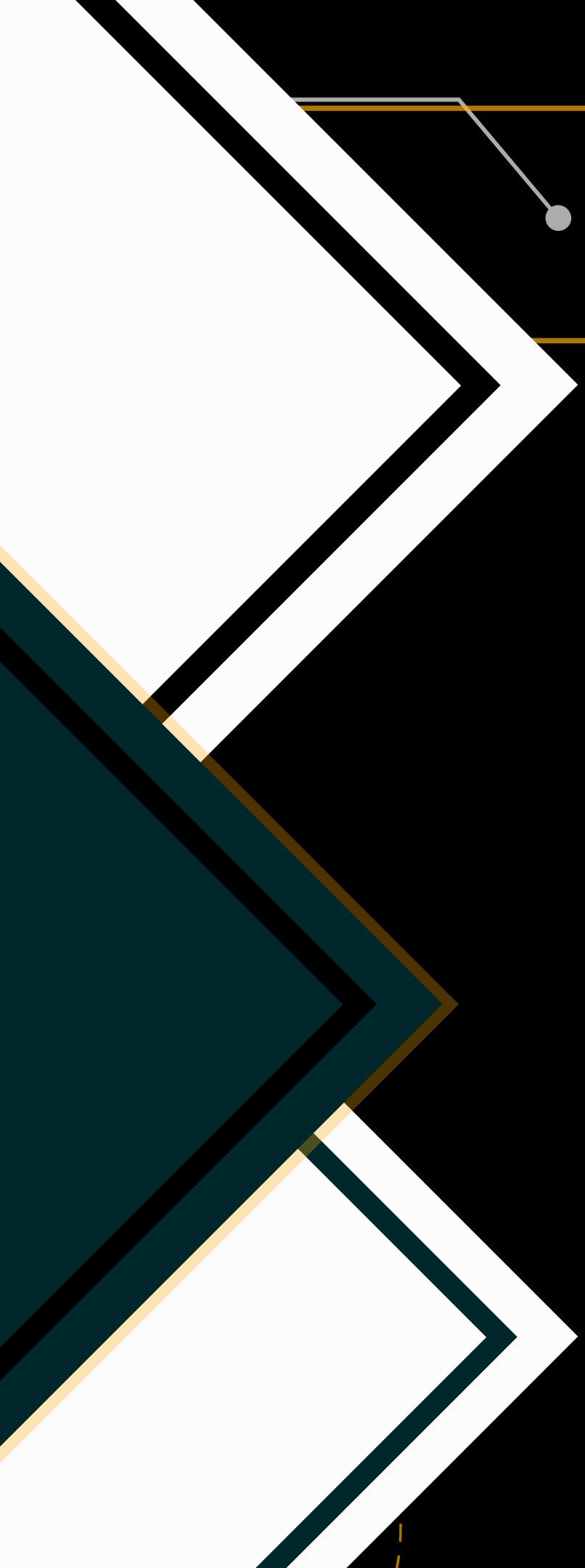
MySQL

MySQL is a tool used to manage databases and servers, so while it's not a database, it's widely used in relation to managing and organising data in databases.

Spring JPA

Spring Data JPA aims to significantly improve the implementation of data access layers by reducing the effort to the amount that's actually needed. It provides custom finder methods, and Spring will provide the implementation automatically





PROJECT DEMONSTRATION

Home Page

[Home](#)[Admin](#)[User](#)[Register](#)[About](#)

Insurance Policy

Please Ensure yourself

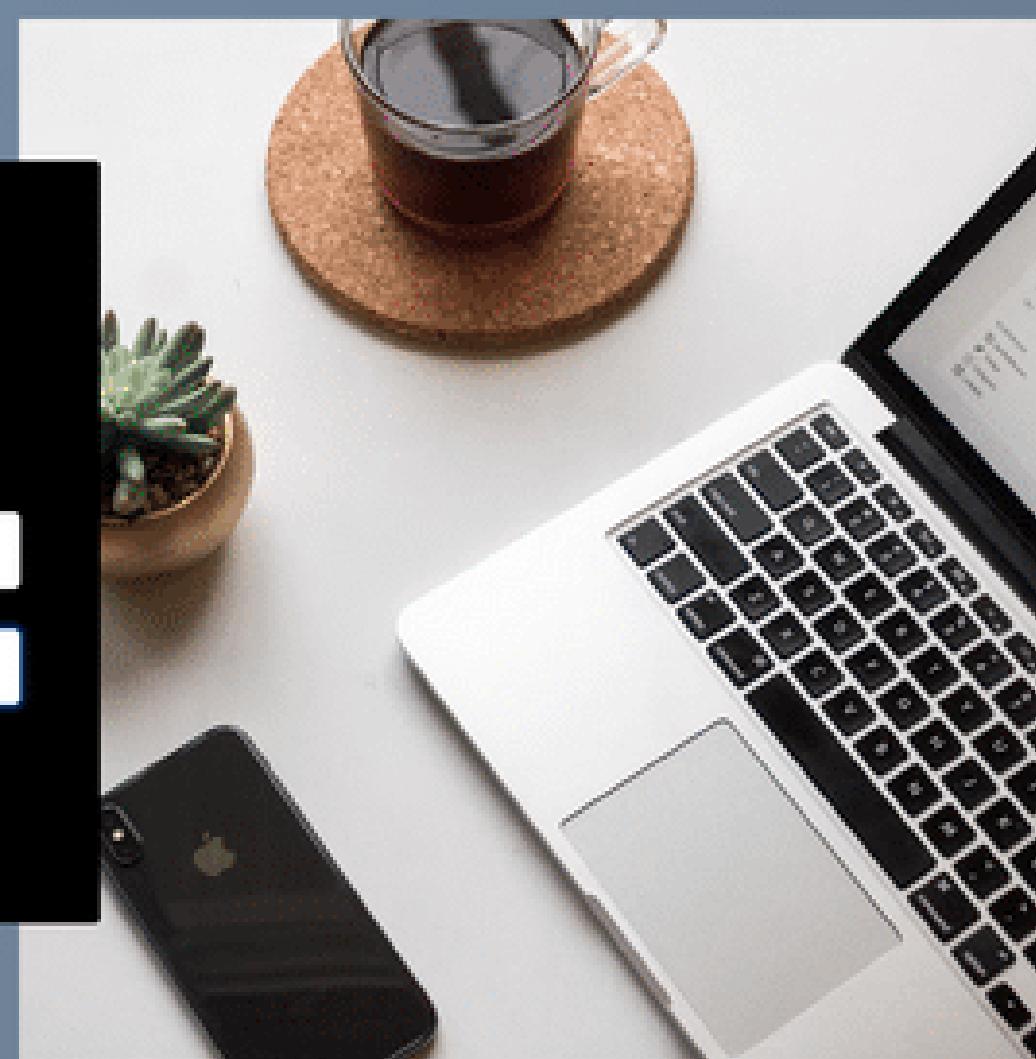


ADMIN LOGIN

Home Admin User Register About

Admin Login

Please login to proceed



ADMIN HOME PAGE

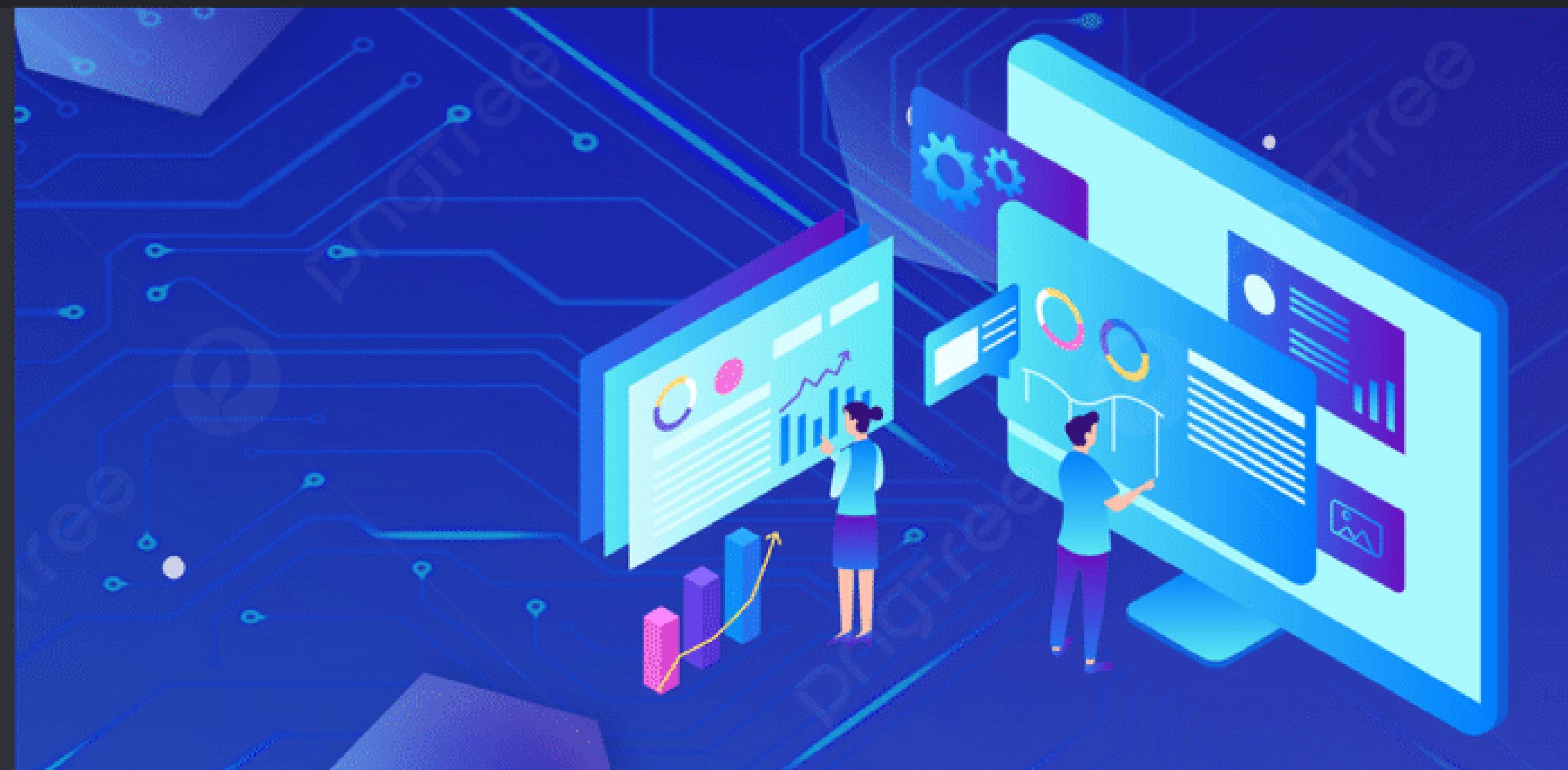
Insurance Management ≡

Logout



Admin

- Dashboard
- Requests
- Category
- Policy
- Quotations



A large, stylized digital illustration set against a dark blue background featuring glowing blue circuit board patterns. In the center, two small blue humanoid figures are interacting with a large, light blue 3D interface. This interface displays various data visualizations, including a bar chart with pink and blue bars, a line graph showing an upward trend, and several smaller charts and icons. A gear icon is visible above the interface, suggesting a process or system.

CUSTOMER REQUEST

[Home](#)[ListCategory](#)[AddCategory](#)[AddPolicies](#)[ViewPolicies](#)[Requests](#)[History](#)[Queries](#)[Logout](#)

Customer Requests

Username	Date	Status		
Mustafa	10/26/22	pending	Approve	Disapprove

CATEGORY

INSURANCE MANAGEMENT

 Admin

 Dashboard

 Requests

 Category

 Policy

 Queries

Logout

List Category



Add Category



CATEGORY LIST

[Home](#)[ListCategory](#)[AddCategory](#)[AddPolicies](#)[ViewPolicies](#)[Requests](#)[History](#)[Queries](#)[Logout](#)

Category List

Category Name	Action
Health Insurance	<button>Delete</button>
Home Insurance	<button>Delete</button>
Motor Insurance	<button>Delete</button>

ADD CATEGORY

Home ListCategory AddCategory AddPolicies ViewPolicies Requests History Queries Logout

Policy category details

Policy category
name

Enter the policy category name

Add

LIST CATEGORY

[Home](#)[ProfileUpdate](#)[ListCategory](#)[ApplyPolicies](#)[YourPolicies](#)[Queries](#)[Contactus](#)[Logout](#)

Category List

Category Name

Health Insurance

Home Insurance

Motor Insurance

POLICY

INSURANCE MANAGEMENT

[Logout](#)



Admin

-  [Dashboard](#)
-  [Requests](#)
-  [Category](#)
-  [Policy](#)
-  [Queries](#)

[View Policy](#)



[Policies](#)



ADD POLICY

[Home](#) [ListCategory](#) [AddCategory](#) [AddPolicies](#) [ViewPolicies](#) [Requests](#) [History](#) [Queries](#) [Logout](#)

Please enter policy details

Policy Name

Enter the policy name

Policy Type

Enter the policy type

Policy cost

Enter the cost of policy

Policy tenure

Enter the tenure

Add

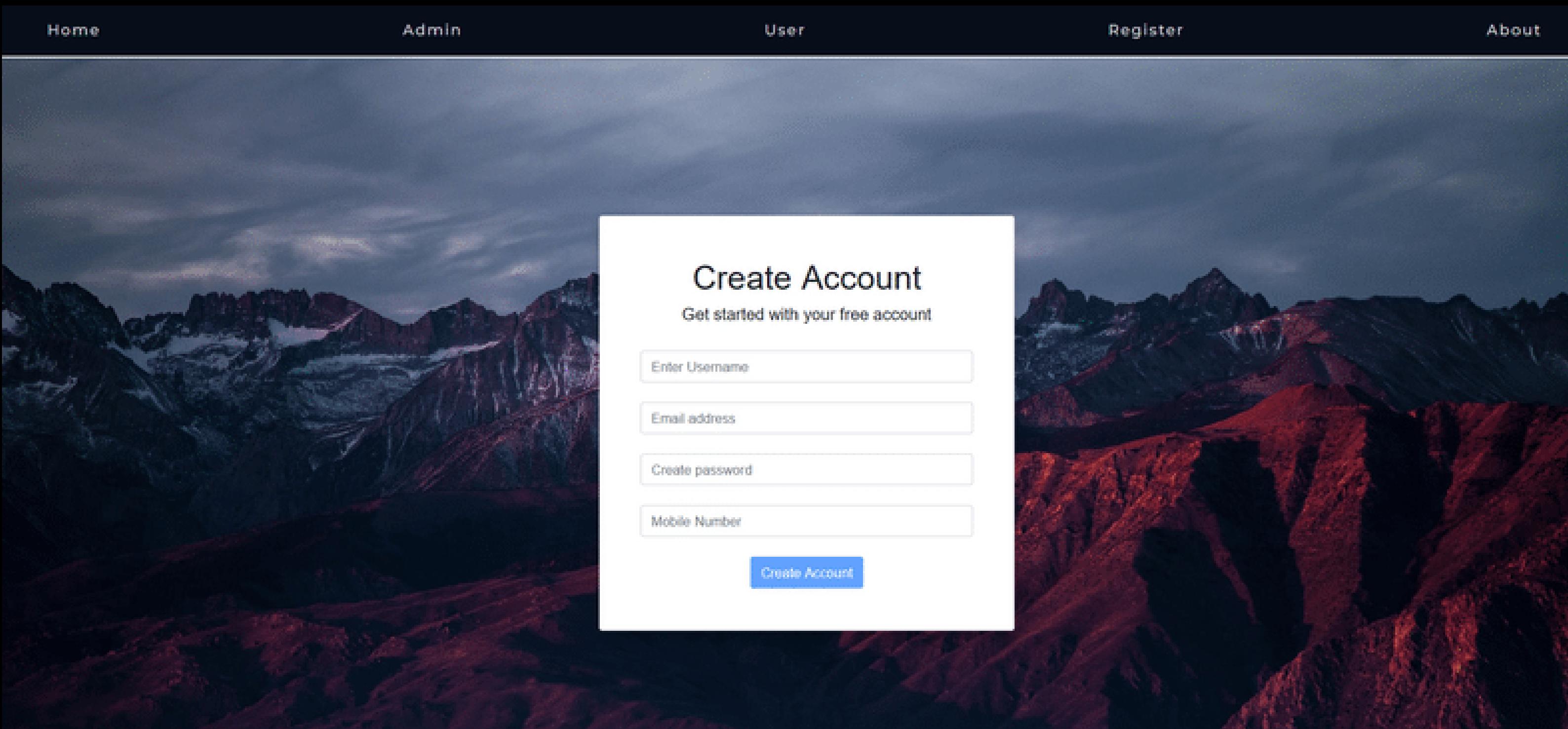
VIEW POLICY

[Home](#) [ListCategory](#) [AddCategory](#) [AddPolicies](#) [ViewPolicies](#) [Requests](#) [History](#) [Queries](#) [Logout](#)

Policies List

Policy Name	Category	Amount	Tenure	Action	
Health Policy	Pre-terminated	5800	3	Delete	Update
Home Policy	Pre-terminated	7200	3	Delete	Update
Motor Policy	Pre-terminated	3200	5	Delete	Update

REGISTRATION

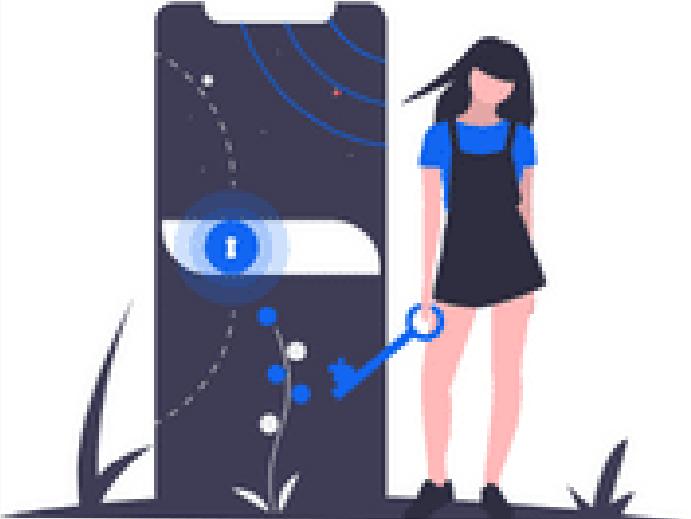


The registration page features a prominent navigation bar at the top with five items: Home, Admin, User, Register, and About. Below the navigation is a large, semi-transparent registration form. The form has a white background and contains the following elements:

- A title "Create Account" in bold black font.
- A subtitle "Get started with your free account" in smaller black font.
- A text input field labeled "Enter Username".
- A text input field labeled "Email address".
- A text input field labeled "Create password".
- A text input field labeled "Mobile Number".
- A blue "Create Account" button at the bottom right of the form.

USER LOGIN

Home Admin User Register About



Login

Sign into your account

Username

Password

Login

Don't have an account? [Register here](#)

USER HOME PAGE

Home ProfileUpdate ListCategory ApplyPolicies YourPolicies Queries Contactus Logout



The image shows a dark-themed landing page for health insurance. In the center, the words "HEALTH INSURANCE" are written in large, white, sans-serif capital letters. Above this text is a red heart icon containing a grey ECG line. Below the main title is a smaller line of text: "Those who love to their life and health,they get secure them by Insurance." At the bottom left, there is a blue button with the word "Apply" in white. To the right of the "Apply" button, the word "HEALTH" is written in large, light-grey capital letters. The background features silhouettes of three people: a woman on the left, a man in the center, and another person on the right. A large blue plus sign is positioned at the bottom left. On the far right, there is a stethoscope and a target symbol. Navigation arrows are located on the left and right sides of the central image.

PROFILE UPDATE

Home ProfileUpdate ListCategory ApplyPolicies YourPolicies Queries Contactus Logout

Update your profile

Instructions:

- Enter the username correctly
- Ensure you create a strong password
- Email and Mobile number can also be updated

Enter your Username

Enter your Email ID

Enter your new password

Enter your Mobile number

Submit

LIST CATEGORY

[Home](#)[ProfileUpdate](#)[ListCategory](#)[ApplyPolicies](#)[YourPolicies](#)[Queries](#)[Contactus](#)[Logout](#)

Category List

Category Name

Health Insurance

Home Insurance

Motor Insurance

APPLY POLICY

Home ProfileUpdate ListCategory ApplyPolicies YourPolicies Queries Contactus Logout

Policies List

Policy Name	Category	Amount	Tenure	
Health Policy	Pre-terminated	5800	3	<button>Apply</button>
Home Policy	Pre-terminated	7200	3	<button>Apply</button>
Motor Policy	Pre-terminated	3200	5	<button>Apply</button>

QUERIES

[Home](#)[ProfileUpdate](#)[ListCategory](#)[ApplyPolicies](#)[YourPolicies](#)[Queries](#)[Contactus](#)[Logout](#)

Enter Your Queries

User

mustafa@gmail.com

Regarding profile Update

I can't able to update my details

Submit

ABOUT US

[Home](#)[ProfileUpdate](#)[ListCategory](#)[ApplyPolicies](#)[YourPolicies](#)[Queries](#)[Contactus](#)[Logout](#)

About Us

We are more than just a company

- 25 Lakhs Families served and more
- More than 12000 marketing intermediaries to guide
- Rs 2 lakh crore and more is the loan book
- 283 marketing offices including one abroad
- Rs 3.35 lakh crore and more cummulative disbursement since inception
- Over the years widened our reach so as to be as close as possible to our customers

Contact Us

www.group4insurance.com

group4@insurance.com

Our Team

- Mustafa Mohammad
- Vanshita Mittal
- Mansi Bhagat
- Bhargavi Singuluri
- Samyuktha N
- Mohammed Kaja
- Praveen A
- Ruthrapriya S
- Santhosh P
- Aravind Soundarajan

QUERIES

[Home](#) [ListCategory](#) [AddCategory](#) [AddPolicies](#) [ViewPolicies](#) [Requests](#) [History](#) [Queries](#) [Logout](#)

Queries

User Name	Subject	Email	Question
Mustafa	Regarding profile Update	mustafa@gmail.com	Facing Issue while Updating the page
User	Regarding profile Update	mustafa@gmail.com	I can't able to update my details

CUSTOMER HISTORY

[Home](#) [ProfileUpdate](#) [ListCategory](#) [ApplyPolicies](#) [YourPolicies](#) [Queries](#) [Contactus](#) [Logout](#)

Customer History

Username	Date	Status
Mustafa	10/26/22	approved

JUNIT TESTING

The screenshot shows a Java development environment with the following details:

- Project Structure:** The left sidebar displays a tree view of Java files under packages like `com.insurance.management` and `com.insurance.model`.
- Code Editor:** The main editor window shows a `Controller.java` file containing JUnit test code. The code includes assertions for policy properties like `PolicyId`, `PolicyName`, `Category`, `Amount`, and `Tenure`.
- Console Output:** Below the editor, the console shows the test results: "Finished after 15.981 seconds".
- Test Summary:** A summary bar at the bottom indicates "Runs: 25/25", "Errors: 0", and "Failures: 0".
- Failure Trace:** A link labeled "Failure Trace" is visible in the bottom right corner.

POSTMAN TESTING

The screenshot shows the Postman application interface. In the top navigation bar, the 'Home' tab is selected. The main workspace is titled 'My Workspace' and contains a collection named 'Personal'. A new request is being prepared for 'http://localhost:9091/api/v1/adminlogin'. The method is set to 'POST' and the URL is 'http://localhost:9091/api/v1/adminlogin'. The 'Body' tab is active, showing a raw JSON payload:

```
1 {  
2   "id": "1",  
3   "username": "admin",  
4   "password": "admin@123"  
5 }  
6
```

The response status is '200 OK' with a response time of '33 ms' and a size of '317 B'. The response body is displayed in JSON format:

```
1 {  
2   "user": "admin",  
3   "password": "admin@123",  
4   "authenticated": true  
5 }
```

CONTD..

The screenshot shows the Postman application interface. On the left, there's a sidebar with 'My Workspace' and sections for 'Personal', 'Collections', 'APIs', 'Environments', 'Mock Servers', 'Monitors', 'Flows', and 'History'. A circular icon at the bottom left has a green checkmark and a red outline.

The main area shows a 'POST http://localhost:9091/' request with a status of '200 OK'. The URL is 'http://localhost:9091/api/v1/user'. The 'Body' tab is selected, showing the following JSON payload:

```
1: {  
2:   "userid": 5,  
3:   "userName": "Mansi",  
4:   "email": "mansi90@gmail.com",  
5:   "mobile": 9898989898,  
6:   "password": "mansi@123"  
7:  
8: }
```

Below the body, the response is displayed in 'Pretty' format:

```
1: {  
2:   "userid": 5,  
3:   "userName": "Mansi",  
4:   "email": "mansi90@gmail.com",  
5:   "mobile": 9898989898,  
6:   "password": "mansi@123"  
7: }
```

Other tabs visible include 'Cookies', 'Headers (8)', 'Test Results', 'Raw', 'Preview', 'Visualize', and 'JSON' dropdown.

CONTD..

The screenshot shows the Postman application interface. On the left, there's a sidebar with 'My Workspace' and sections for 'Personal', 'Collections', 'APIs', 'Environments', 'Mock Servers', 'Monitors', 'Flows', and 'History'. A central workspace shows a 'PUT' request to 'http://localhost:9091/api/v1/user/Mustafa'. The 'Body' tab is selected, showing a JSON payload:

```
1  {
2     "userId": 1,
3     "userName": "Mustafa",
4     "email": "mustafa@gmail.com",
5     "mobile": 7777777777,
6     "password": "M@ll1"
7 }
```

Below the body, the response status is '200 OK' with a time of '9 ms' and a size of '352 B'. The response body is identical to the request body. At the bottom, there are tabs for 'Pretty', 'Raw', 'Preview', 'Visualize', and 'JSON'.

CONTD..

The screenshot shows the Postman application interface. On the left, there's a sidebar with 'My Workspace' and sections for 'Collections', 'APIs', 'Environments', 'Mock Servers', 'Monitors', 'Flows', and 'History'. A circular icon at the bottom left has a green checkmark and a magnifying glass. The main area shows a collection named 'New Collection'. A request card for 'GET http://localhost:9091/api/v1/users' is selected. The 'Body' tab is active, showing a JSON response with two user objects. The response body is:

```
1 {
2   "userid": 5,
3   "userName": "mansi",
4   "email": "mansi90@gmail.com",
5   "mobile": 9898989898,
6   "password": "mansi@123"
7 },
8 {
9   "userid": 1,
10  "userName": "Mustafa",
11  "email": "Mustafa@gmail",
12  "mobile": 9876543219,
13  "password": "123456"
14 }
```

The status bar at the bottom shows '200 OK 9 ms 455 B' and various navigation icons.

CONTD..

The screenshot shows the Postman application interface. On the left, there's a sidebar with 'My Workspace' and sections for 'Collections', 'APIs', 'Environments', 'Mock Servers', 'Monitors', 'Flows', and 'History'. A circular icon at the bottom left has a green checkmark and a magnifying glass. The main area shows a collection named 'Personal' with a 'New Collection' option. A request card is displayed with the method 'GET', URL 'http://localhost:9091/api/v1/user/mansi', and a status bar indicating '200 OK' with 13 ms and 355 B. The 'Body' tab is selected, showing a JSON response with fields: 'userid': 5, 'userName': 'mansi', 'email': 'mansi90@gmail.com', 'mobile': 9898989898, and 'password': 'mansi@123'. Other tabs include 'Cookies', 'Headers (8)', 'Test Results', 'Pretty', 'Raw', 'Preview', 'Visualize', and 'JSON'. Buttons for 'Save', 'Send', and 'Edit' are also visible.

```
1 {  
2   "userid": 5,  
3   "userName": "mansi",  
4   "email": "mansi90@gmail.com",  
5   "mobile": 9898989898,  
6   "password": "mansi@123"  
7 }
```

CONTD..

The screenshot shows the Postman application interface. On the left, there's a sidebar with icons for Collections, APIs, Environments, Mock Servers, Monitors, Flows, and History. The main area displays a request configuration for a DELETE operation:

- Request URL:** `http://localhost:9091/api/v1/user/Mustafa`
- Method:** `DELETE`
- Headers:** (6 items listed)
- Body:** `raw` (selected) - contains the value "1"
- Response Status:** `200 OK`, `22 ms`, `279 B`
- Response Body:** `1 User deleted successfully`

At the bottom, there are navigation links for Online, Find and Replace, Console, Cookies, Capture requests, Bootcamp, Runner, Trash, and Help.

CONTD..

The screenshot shows the Postman application interface. On the left, there's a sidebar with 'My Workspace' and various categories like Collections, APIs, Environments, Mock Servers, Monitors, Flows, and History. The main area displays a POST request to `http://localhost:9091/api/v1/login`. The 'Body' tab is selected, showing the following JSON payload:

```
1 {  
2   "userName": "mansi",  
3   "password": "mansi@123"  
4 }
```

The response section shows a 200 OK status with a response time of 46 ms and a size of 419 B. The response body is also JSON:

```
1 {  
2   "userName": "mansi",  
3   "password": "mansi@123",  
4   "authenticated": true  
5 }
```

At the bottom, there are tabs for Body, Cookies (1), Headers (9), and Test Results, along with buttons for Pretty, Raw, Preview, Visualize, and JSON.

The screenshot shows the Postman application interface. On the left, the sidebar displays the "My Workspace" section with a collection named "Personal" selected. The main workspace shows a GET request to `http://localhost:9091/api/v1/categories`. The "Body" tab is active, showing the raw JSON content:

```
1 {  
2   "category": "Life"  
3 }
```

The response section shows the following JSON data:

```
1 [ {  
2   "category": "Health"  
3 }, {  
4   "category": "Life"  
5 } ]
```

Postman

Home Workspaces API Network Explore Search Postman Invite Upgrade

New Import GET http://localhost:9091/a • GET Untitled Request + No Environment

My Workspace Personal Collections APIs Environments Mock Servers Monitors Flows History

+

http://localhost:9091/api/v1/category/Life

GET http://localhost:9091/api/v1/category/Life

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON Beautify

Send

1

Body Cookies (1) Headers (8) Test Results 200 OK 47 ms 272 B Save Response

Pretty Raw Preview Visualize JSON

1 "category": "Life"

Online Find and Replace Console Cookies Capture requests Bootcamp Runner Trash

The screenshot shows the Postman application interface. On the left, there's a sidebar with options like Home, Workspaces, API Network, Explore, and a search bar. Below that is a 'My Workspace' section with a 'Personal' tab selected. A 'Collections' icon is highlighted. The main area shows a collection named 'Personal' with a single request. The request URL is 'http://localhost:9091/api/v1/category/Life' with a GET method. The 'Body' tab is selected, showing a JSON response with one item: { "category": "Life" }. The status bar at the bottom indicates a 200 OK response with 47 ms and 272 B.

Bill-project-management/src/main/java/com/insurance/management/controller/Controller.java - Spring Tool Suite 4

Home Workspaces API Network Explore Search Postman Invite Upgrade

My Workspace Personal New Import No Environment

Collections APIS Environments Mock Servers Monitors Flows History

DELETE http://localhost:9091/api/v1/category/Health

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON

Send

Body Cookies (1) Headers (8) Test Results 200 OK 29 ms 293 B Save Response

Pretty Raw Preview Visualize Text

1 Insurance category deleted successfully

Online Find and Replace Console Cookies Capture requests Bootcamp Runner Trash

The screenshot shows the Postman application interface. At the top, there's a header bar with tabs for Home, Workspaces, API Network, Explore, and a search field. Below the header is a navigation bar for 'My Workspace' with options for Personal, New, Import, and a link to 'No Environment'. On the left side, there's a sidebar with icons for Collections, APIS, Environments, Mock Servers, Monitors, Flows, and History. The main workspace shows a DELETE request to 'http://localhost:9091/api/v1/category/Health'. The 'Body' tab is selected, showing the raw response: '1 Insurance category deleted successfully'. The status bar at the bottom indicates a 200 OK response with 29 ms and 293 B. There are also buttons for Save, Send, and various response formats like Pretty, Raw, Preview, Visualize, and Text.

RIL project - management/src/main/java/com/insurance/management/controller/Controller.java - Sonar Tool Suite 4

Home Workspaces API Network Explore Search Postman Invite Upgrade

My Workspace Personal New Import POST http://localhost:9091/ ● GET Untitled Request + *** No Environment

New Collection

Collections APIs Environments Mock Servers Monitors Flows History

POST http://localhost:9091/api/v1/policy

Params Authorization Headers (9) Body Pre-request Script Tests Settings Cookies

Body (1) Headers (8) Test Results 200 OK 53 ms 334 B Save Response

Body (Pretty Raw Preview Visualize JSON) 1 {
2 "policyName": "JeevanBima",
3 "category": "Life",
4 "amount": "500000",
5 "tenureInYears": "2"
6 }

Body (Pretty Raw Preview Visualize JSON) 1 {
2 "policyName": "JeevanBima",
3 "category": "Life",
4 "amount": 500000.0,
5 "tenureInYears": 2
6 }

Online Find and Replace Console Cookies Capture requests Bootcamp Runner Trash

RIL project - management/src/main/java/com/insurance/management/controller/Controller.java - Spring Tool Suite 4

Home Workspaces API Network Explore Search Postman Invite Upgrade

My Workspace Personal New Import GET http://localhost:9091/a • GET Untitled Request + ... No Environment

Collections APIs Environments Mock Servers Monitors Flows History

New Collection

http://localhost:9091/api/v1/policies

GET http://localhost:9091/api/v1/policies

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON Beautify

1

Body Cookies (1) Headers (8) Test Results 200 OK 15 ms 428 B Save Response

Pretty Raw Preview Visualize JSON

```
1
2   [
3     {
4       "policyName": "CarInsurance",
5       "category": "Vehicle",
6       "amount": 200000.0,
7       "tenureInYears": 2
8     },
9     {
10       "policyName": "JeevanBima",
11       "category": "Financial",
12       "amount": 300000.0,
13       "tenureInYears": 3
14     }
15   ]
```

Online Find and Replace Console Cookies Capture requests Bootcamp Runner Trash

Ell project - management/src/main/java/com/insurance/management/controller/Controller.java - Spring Tool Suite 4

Home Workspaces API Network Explore Search Postman Invite Upgrade

My Workspace Personal New Import No Environment

Collections + New Collection APIs Environments Mock Servers Monitors Flows History

PUT http://localhost:9091/api/v1/policy/JeevanBima GET Untitled Request + ***

http://localhost:9091/api/v1/policy/JeevanBima

PUT http://localhost:9091/api/v1/policy/JeevanBima Save Send

Params Authorization Headers (9) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON Beautify

1 {
2 "policyName": "JeevanBima",
3 "category": "Financial",
4 "amount": "300000",
5 "tenureInYears": "3"
6 }

Body Cookies (1) Headers (8) Test Results 200 OK 41 ms 339 B Save Response

Pretty Raw Preview Visualize JSON

1 {
2 "policyName": "JeevanBima",
3 "category": "Financial",
4 "amount": 300000.0,
5 "tenureInYears": 3
6 }

Online Find and Replace Console Cookies Capture requests Bootcamp Runner Trash

All project - management/src/main/java/com/insurance/management/controller/Controller.java - Spring Tool Suite 4

Home Workspaces API Network Explore Search Postman Invite Upgrade

My Workspace Personal New Import GET http://localhost:9091/a • GET Untitled Request + ... No Environment

http://localhost:9091/api/v1/policy/CarInsurance

GET http://localhost:9091/api/v1/policy/CarInsurance Send

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON Beautify

Body Cookies (1) Headers (8) Test Results 200 OK 23 ms 339 B Save Response

Pretty Raw Preview Visualize JSON

```
1 "policyName": "CarInsurance",
2 "category": "Vehicle",
3 "amount": 200000.0,
4 "tenureInYears": 2
```

Online Find and Replace Console Cookies Capture requests Bootcamp Runner Trash

The screenshot shows the Postman application interface. In the top navigation bar, there are tabs for Home, Workspaces, API Network, and Explore. A search bar labeled 'Search Postman' is located at the top right. Below the navigation, there's a section for 'My Workspace' with a 'Personal' tab selected. A 'Collections' icon is highlighted with a red border. On the left sidebar, there are icons for APIs, Environments, Mock Servers, Monitors, Flows, and History. A circular icon with a green arrow and a magnifying glass is at the bottom left. The main workspace shows a collection named 'http://localhost:9091/api/v1/policy/CarInsurance'. A specific GET request is selected, with the URL being 'http://localhost:9091/api/v1/policy/CarInsurance'. The 'Body' tab is active, showing a JSON response with fields: policyName, category, amount, and tenureInYears. The response status is 200 OK with a time of 23 ms and a size of 339 B. At the bottom, there are buttons for Online, Find and Replace, Console, Cookies, Capture requests, Bootcamp, Runner, Trash, and a settings gear icon.

RIL project - management/src/main/java/com/insurance/management/controller/Controller.java - Spring Tool Suite 4

Home Workspaces API Network Explore Search Postman Invite Upgrade

My Workspace Personal New Import No Environment

Collections APIs Environments Mock Servers Monitors Flows History

DEL http://localhost:9091/a • GET Untitled Request + ...

http://localhost:9091/api/v1/policy/CarInsurance

DELETE http://localhost:9091/api/v1/policy/CarInsurance

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

Body (1) none form-data x-www-form-urlencoded raw binary GraphQL JSON

Send

Body Cookies (1) Headers (8) Test Results 200 OK 34 ms 281 B Save Response

Pretty Raw Preview Visualize Text

1 policy deleted successfully

Online Find and Replace Console Cookies Capture requests Bootcamp Runner Trash

RIL project - management/src/main/java/com/insurance/management/controller/Controller.java - Spring Tool Suite 4

Home Workspaces API Network Explore Search Postman Invite Upgrade

My Workspace Personal New Import POST http://localhost:9091/ ● GET Untitled Request + No Environment

Collections APIs Environments Mock Servers Monitors Flows History

http://localhost:9091/api/v1/approvals

POST http://localhost:9091/api/v1/approvals

Params Authorization Headers (9) Body Pre-request Script Tests Settings Cookies

Body (1) none form-data x-www-form-urlencoded raw binary GraphQL JSON Beautify

1 {
2 ... "userName": "Vanshita",
3 ... "policyId" : 2,
4 ... "requestId":1,
5 ... "date" :2,
6 ... "status" : "pending"
7 }

Body Cookies (1) Headers (8) Test Results 200 OK 27 ms 257 B Save Response

Pretty Raw Preview Visualize Text

1 added

Online Find and Replace Console Cookies Capture requests Bootcamp Runner Trash

All project - management/src/main/java/com/insurance/management/controller/Controller.java - Spring Tool Suite 4

Home Workspaces API Network Explore Search Postman Invite Upgrade

My Workspace Personal New Import POST http://localhost:9091/ ● GET Untitled Request + No Environment

Collections APIs Environments Mock Servers Monitors Flows History

POST http://localhost:9091/api/v1/user

Params Authorization Headers (9) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON Beautify

```
1: {  
2:   "userId": 6,  
3:   "userName": "Vanshita",  
4:   "email": "vanshita@gmail.com",  
5:   "mobile": 7777777777,  
6:   "password": "Vans"  
7:  
8: }
```

Body Cookies (1) Headers (8) Test Results 200 OK 115 ms 254 B Save Response

Pretty Raw Preview Visualize JSON

```
1: {  
2:   "userId": 6,  
3:   "userName": "Vanshita",  
4:   "email": "vanshita@gmail.com",  
5:   "mobile": 7777777777,  
6:   "password": "Vans"  
7: }
```

Online Find and Replace Console Cookies Capture requests Bootcamp Runner Trash

RIL project - management/src/main/java/com/insurance/management/controller/Controller.java - Spring Tool Suite 4

Home Workspaces API Network Explore Search Postman Invite Upgrade

My Workspace Personal New Import GET http://localhost:9091/a • GET Untitled Request + ... No Environment

Collections + New Collection APIs Environments Mock Servers Monitors Flows History

http://localhost:9091/api/v1/approvals

GET http://localhost:9091/api/v1/approvals

Params Authorization Headers (9) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON

Send

Body Cookies (1) Headers (8) Test Results 200 OK 12 ms 494 B Save Response

Pretty Raw Preview Visualize JSON

```
1 [ { 2   "user": "Mansi", 3   "policyId": "Life Insurance", 4   "requestId": null, 5   "date": "2022-10-25T17:04:27.521+00:00", 6   "status": "pending" 7 }, 8 ], 9 { 10   "user": "Vanshita", 11   "policyId": null, 12   "requestId": null, 13   "date": "2022-10-25T17:03:40.229+00:00", 14   "status": "pending" }
```

Online Find and Replace Console Cookies Capture requests Bootcamp Runner Trash

RII project - management/src/main/java/com/insurance/management/service/ApprovalsService.java - Spring Tool Suite 4

Home Workspaces API Network Explore Search Postman Invite Upgrade

My Workspace Personal New Import POST http://localhost:9091/ ● GET Untitled Request + ... No Environment

New Collection http://localhost:9091/api/v1/approvals Save Edit

POST http://localhost:9091/api/v1/approvals Send

Params Authorization Headers (9) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON Beautify

1 {
2 ... "userName": "Mansi",
3 ... "policyId": "Life Insurance"
4 }

Body Cookies (1) Headers (8) Test Results 200 OK 29 ms 257 B Save Response

Pretty Raw Preview Visualize Text

1 added

Online Find and Replace Console Cookies Capture requests Bootcamp Runner Trash

RIL project - management/src/main/java/com/insurance/management/controller/Controller.java - Spring Tool Suite 4

Home Workspaces API Network Explore Search Postman Invite Upgrade

My Workspace Personal New Import POST http://localhost:9091/ ● GET Untitled Request + No Environment

http://localhost:9091/api/v1/approved Save Edit

POST http://localhost:9091/api/v1/approved Send

Params Authorization Headers (9) Body * Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON Beautify

```
1 {  
2   "user": "Mansi",  
3   "policyId": "Life Insurance"  
4 }
```

Body Cookies (1) Headers (8) Test Results 200 OK 27 ms 260 B Save Response

Pretty Raw Preview Visualize Text

1 approved

Online Find and Replace Console Cookies Capture requests Bootcamp Runner Trash

The screenshot shows the Postman application interface. On the left, there's a sidebar with 'My Workspace' and various tabs like 'Collections', 'APIs', 'Environments', etc. The main area shows a POST request to 'http://localhost:9091/api/v1/approved'. The 'Body' tab is selected, showing a JSON payload:

```
1 {  
2   "user": "Mansi",  
3   "policyId": "Life Insurance"  
4 }
```

The response section shows a 200 OK status with a response body of 'approved'.

All project - management/src/main/java/com/insurance/management/controller/Controller.java - Spring Tool Suite 4

Home Workspaces API Network Explore Search Postman Invite Upgrade

My Workspace Personal New Import GET http://localhost:9091/a ● GET Untitled Request + ... No Environment

Collections + New Collection http://localhost:9091/api/v1/approved/Mansi Save Edit

APIs Environments Mock Servers Monitors Flows History

GET http://localhost:9091/api/v1/approved/Mansi Send

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON

Body Cookies (1) Headers (8) Test Results 200 OK 14 ms 378 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {  
2   "userName": "Mansi",  
3   "policyId": "Life Insurance",  
4   "requestId": null,  
5   "date": "2022-10-25T17:18:55.205+00:00",  
6   "status": "approved"  
7 }
```

All project - management/src/main/java/com/insurance/management/controller/Controller.java - Spring Tool Suite 4

Home Workspaces API Network Explore Search Postman Invite Upgrade

My Workspace Personal New Import GET http://localhost:9091/a ● GET Untitled Request + No Environment

Collections APIs Environments Mock Servers Monitors Flows History

http://localhost:9091/api/v1/disapproved

GET http://localhost:9091/api/v1/disapproved

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies Beautify

none form-data x-www-form-urlencoded raw binary GraphQL JSON

1

Body Cookies (1) Headers (8) Test Results 200 OK 16 ms 374 B Save Response

Pretty Raw Preview Visualize JSON

1 [2 3 4 5 6 7 8 9]

"userNmae": "Vanshita",
"policyId": "LIC",
"requestId": null,
"date": "2022-10-25T17:21:16.187+00:00",
"status": "disapproved"

Online Find and Replace Console Cookies Capture requests Bootcamp Runner Trash

RIL project - management/src/main/java/com/insurance/management/controller/Controller.java - Spring Tool Suite 4

Home Workspaces API Network Explore Search Postman Invite Upgrade

My Workspace Personal New Import POST http://localhost:9091/ ● GET Untitled Request + No Environment

Collections APIs Environments Mock Servers Monitors Flows History

http://localhost:9091/api/v1/query

POST http://localhost:9091/api/v1/query

Params Authorization Headers (9) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON Beautify

1 {
2 ... "userName": "Mansi",
3 ... "queryid": "1",
4 ... "mailid": "mansi@gmail.com",
5 ... "question": "What is my policy request status?"
6 }

Body Cookies (1) Headers (8) Test Results 200 OK 45 ms 361 B Save Response

Pretty Raw Preview Visualize JSON

1 {
2 ... "userName": "Mansi",
3 ... "queryid": "1",
4 ... "mailid": "mansi@gmail.com",
5 ... "question": "What is my policy request status?"
6 }

Online Find and Replace Console Cookies Capture requests Bootcamp Runner Trash

RIL project - management/src/main/java/com/insurance/management/controller/Controller.java - Spring Tool Suite 4

Home Workspaces API Network Explore Search Postman Invite Upgrade

My Workspace Personal New Import GET http://localhost:9091/a Untitled Request No Environment

Collections APIs Environments Mock Servers Monitors Flows History

New Collection

http://localhost:9091/api/v1/query/Samyuktha

GET http://localhost:9091/api/v1/query/Samyuktha

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies Beautify

none form-data x-www-form-urlencoded raw binary GraphQL JSON

1

Body Cookies (1) Headers (8) Test Results 200 OK 18 ms 358 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {  
2   "userName": "Samyuktha",  
3   "queryId": "2",  
4   "mailId": "sam@gmail.com",  
5   "question": "How to update policy amount?"  
6 }
```

Online Find and Replace Console Cookies Capture requests Bootcamp Runner Trash

Home Workspaces API Network Explore Search Postman Invite Upgrade

My Workspace Personal New Import GET http://localhost:9091/a ● GET Untitled Request + No Environment

Collections + New Collection

APIs Environments Mock Servers Monitors Flows History

GET http://localhost:9091/api/v1/query Headers (7) Body Pre-request Script Tests Settings Cookies

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON Beautify

Body Cookies (1) Headers (8) Test Results 200 OK 18 ms 657 B Save Response

Pretty Row Preview Visualize JSON

```
13 },
14 {
15   "userName": "Praveen",
16   "queryid": "Password Update issue",
17   "mailid": null,
18   "question": "Appupdate"
19 },
20 {
21   "userName": "Samyuktha",
22   "queryid": "2",
23   "mailid": "sam@gmail.com",
24   "question": "How to update policy amount?"
25 }
26 ]
```

Online Find and Replace Console Cookies Capture requests Bootcamp Runner Trash

The screenshot shows the Postman application interface. On the left, there's a sidebar with icons for Collections, APIs, Environments, Mock Servers, Monitors, Flows, and History. The main area displays a collection named 'Personal' with a single request titled 'Untitled Request'. The request URL is 'http://localhost:9091/api/v1/query'. The 'Body' tab is selected, showing a JSON response with two entries. The JSON output is:

```
13 },
14 {
15   "userName": "Praveen",
16   "queryid": "Password Update issue",
17   "mailid": null,
18   "question": "Appupdate"
19 },
20 {
21   "userName": "Samyuktha",
22   "queryid": "2",
23   "mailid": "sam@gmail.com",
24   "question": "How to update policy amount?"
25 }
26 ]
```

All project - management/src/main/java/com/insurance/management/controller/Controller.java - Spring Tool Suite 4

Home Workspaces API Network Explore Search Postman Invite Upgrade

My Workspace Personal New Import No Environment

Collections APIs Environments Mock Servers Monitors Flows History

DEL http://localhost:9091/a • GET Untitled Request + +

http://localhost:9091/api/v1/query/Mustafa

DELETE http://localhost:9091/api/v1/query/Mustafa

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON Beautify

Body Cookies (1) Headers (8) Test Results 200 OK 25 ms 280 B Save Response

Pretty Raw Preview Visualize Text

1 Query deleted successfully

Online Find and Replace Console Cookies Capture requests Bootcamp Runner Trash

The screenshot shows the Postman application interface. In the top navigation bar, there are tabs for Home, Workspaces, API Network, and Explore. A search bar labeled 'Search Postman' is located at the top right. Below the navigation, there's a section for 'My Workspace' with a 'Personal' tab selected. On the left sidebar, there are icons for Collections, APIs, Environments, Mock Servers, Monitors, Flows, and History. The main workspace area has a 'New Collection' button and a 'New' button for creating a new request. A request card for a 'DELETE' operation is selected, with the URL 'http://localhost:9091/api/v1/query/Mustafa'. The 'Body' tab is active, showing the raw response: '1 Query deleted successfully'. The status bar at the bottom indicates a 200 OK response with a 25 ms duration and 280 B size.

Thank You!!