

INSURANCE POLICY MANAGEMENT



GROUP 4

TEAM MEMBERS

- **Thiruwieddhi Hanumann**
- **Suraj Bhan**
- **Akshay Raj**
- **Perla Manikanta Sunil**
- **Piduru Gurunadha Chakradhar**
- **Gopika Madhu**
- **Kamali Ramraj**
- **Sanjay K**
- **Nancy X**

CONTENTS

- Abstract
- Project Objective
- Overview
- Tools And Technologies
- Admin Functional Requirements And Flow Chart
- User Functional Requirements And Flow Chart
- Angular Architecture
- Database Architecture
- Implementation
- Testing
- Conclusion

ABSTRACT

- Today, As We Are Living In Uncertainty, What May Happen Tomorrow No One Knows, Whether It Is Related To Our Health Or The Objects Around Us That We Use In Our Daily Life. So, Why Not Take Insurance. Insurance Is All About Providing Financial Safety To The Individual Or Vehicles.
- Our Project Provide A Bug-free Application To The Policymaker(admin) As Well As The Customer.
- The Main Objective Is To Build A Secured, Robust Insurance Management System Where The Policies Are Managed Properly.
- It Maintains The Record Of Customers, Policies, Buyers' Policies Efficiently So That It Would Be Easy To Access At Any Time 24*7.

PROJECT OBJECTIVE

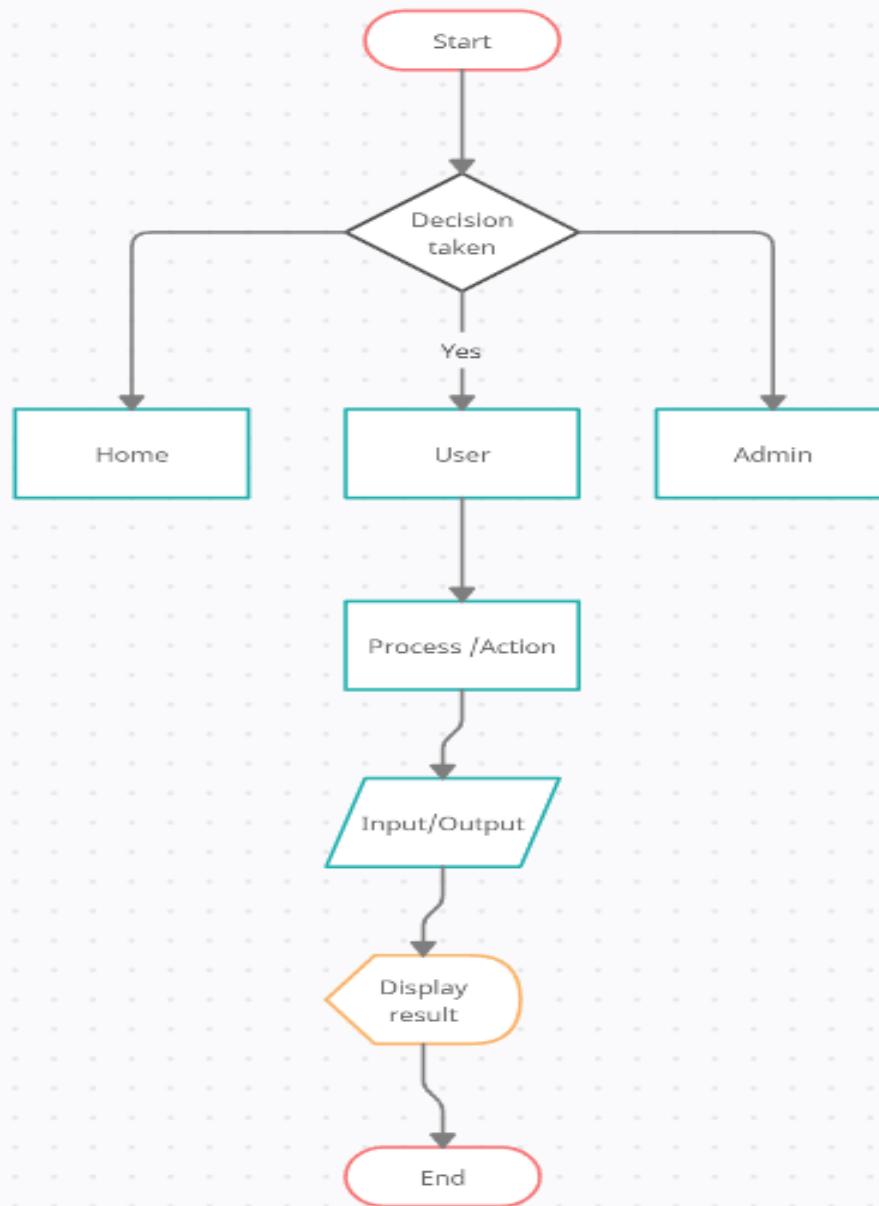


- Create A Dynamic And Responsive Web Application For Managing The Policies Online.

OVERVIEW

- It Is A Multi-role Application Project I.E. Multiple Admin And Multiple User, Where Admin Will Have The Main Control Over The System.
- Admin Will Be Responsible For Maintaining The System And Also Keep Checking On The Policies And Management Of The System. Admin Will Manage All The Applicants For Insurance And Their Details.
- Admin Can Add, Update And Delete The Policy And Can Approve Or Disapprove The Policy Applied By The Users.
- On The Other Hand, Users Can Register Themselves, Log In Using The Login Credentials, Apply For Registering The Type Of Policy They Wish To Apply And They Can Check Their Status In The History Section.
- Users Can Ask Question And Admin Can Answer For That.

OVERVIEW FLOW CHART



TOOLS AND TECHNOLOGIES



Tools

Visual studio code
Spring tool suite
MYSQL workbench



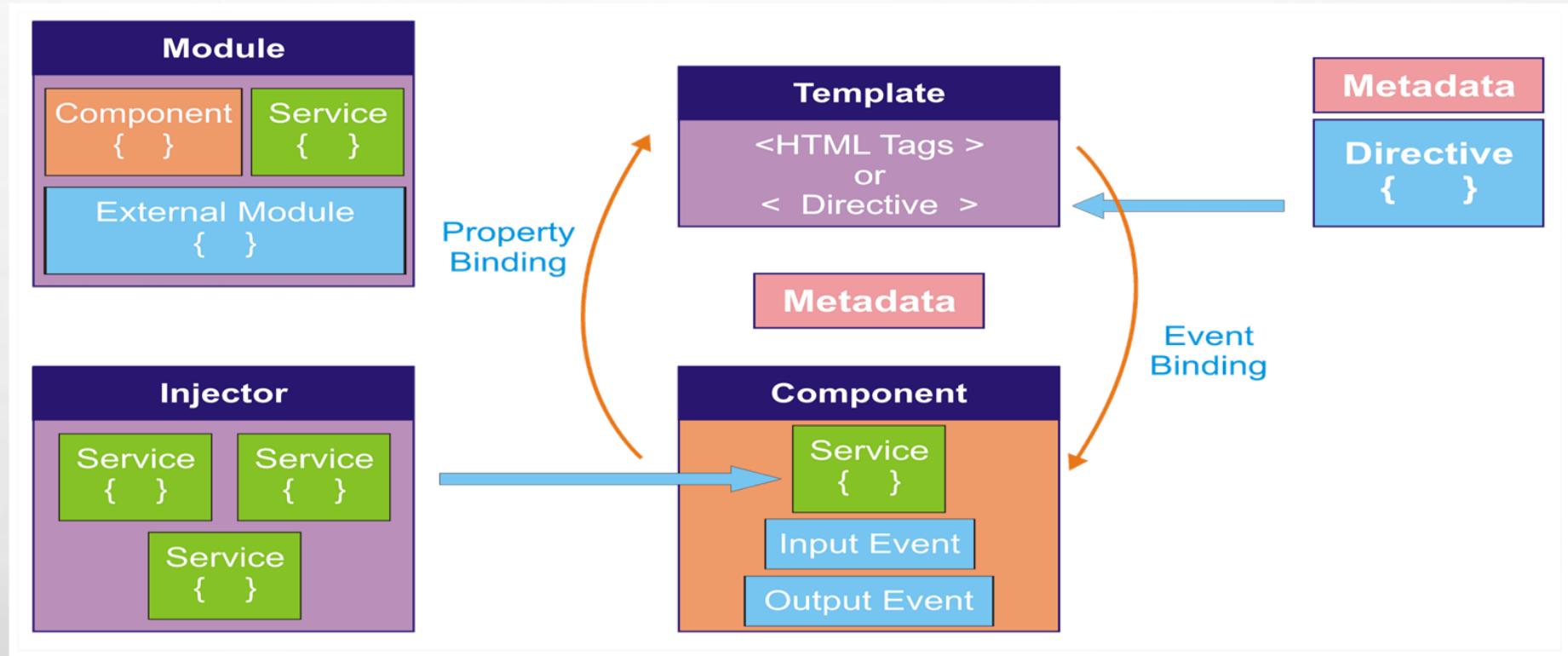
Technologies

Java spring boot
Angular
MY SQL



Server

Tomcat



ANGULAR ARCHITECTURE





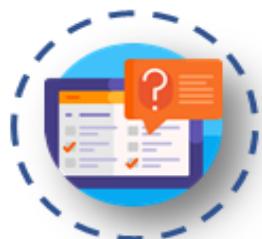
Custom
components



Data binding



Dependency
injection



Testing



Comprehensive



Browser
compatibility

ANGULAR ADVANTAGES



SPRING BOOT

- Spring Boot Is A Java Framework, Built On Top Of The Spring, Used For Developing Web Applications.
- Spring Boot Is A Backend Framework That Has Become A Major Player In The Enterprise Java EE.
- No Requirement For Complex XML Configurations.
- It Allows You To Create REST APIs With Minimal Configurations.
- A Few Benefits Of Using Spring Boot For Your REST APIs Include:

DATA BASE

- **Entities In The Database**

- User
- Policy
- Record
- Questions

DEPENDENCIES

- Spring Data JPA – Java Persistence API And Hibernate.
- Spring Web – To Include Spring MVC And Embed Tomcat Into Your Project.
- Spring Boot Dev-tools – Development Tools.
- Mysql Driver – JDBC Driver (Connecting Spring Boot To The Database)

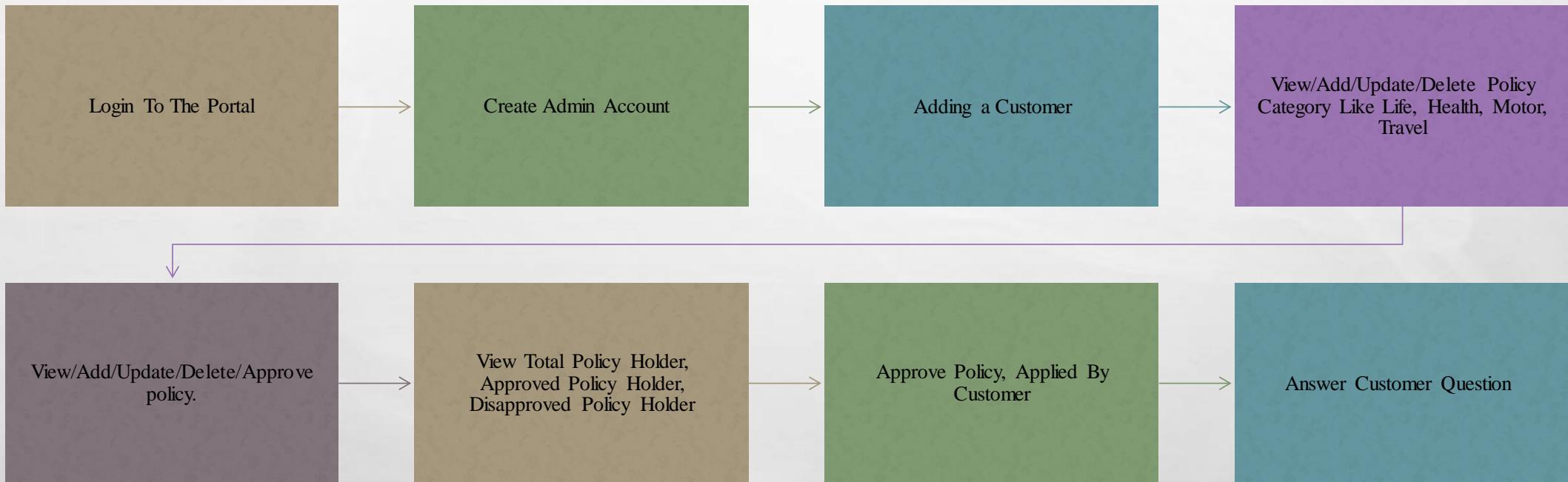
ANNOTATIONS

- **@Autowired** : To The User Repository For Dependency Injection.
- **@Getmapping And @Postmapping**: To Specify The Type Of HTTP Requests Accepted
- **@Restcontroller** : It Is A Combination Of **@Controller** And **@Responsebody**
- The **@Requestbody** : It Is Used To Map The Body Of The POST Request Sent To The Endpoint To The User Instance To Save.

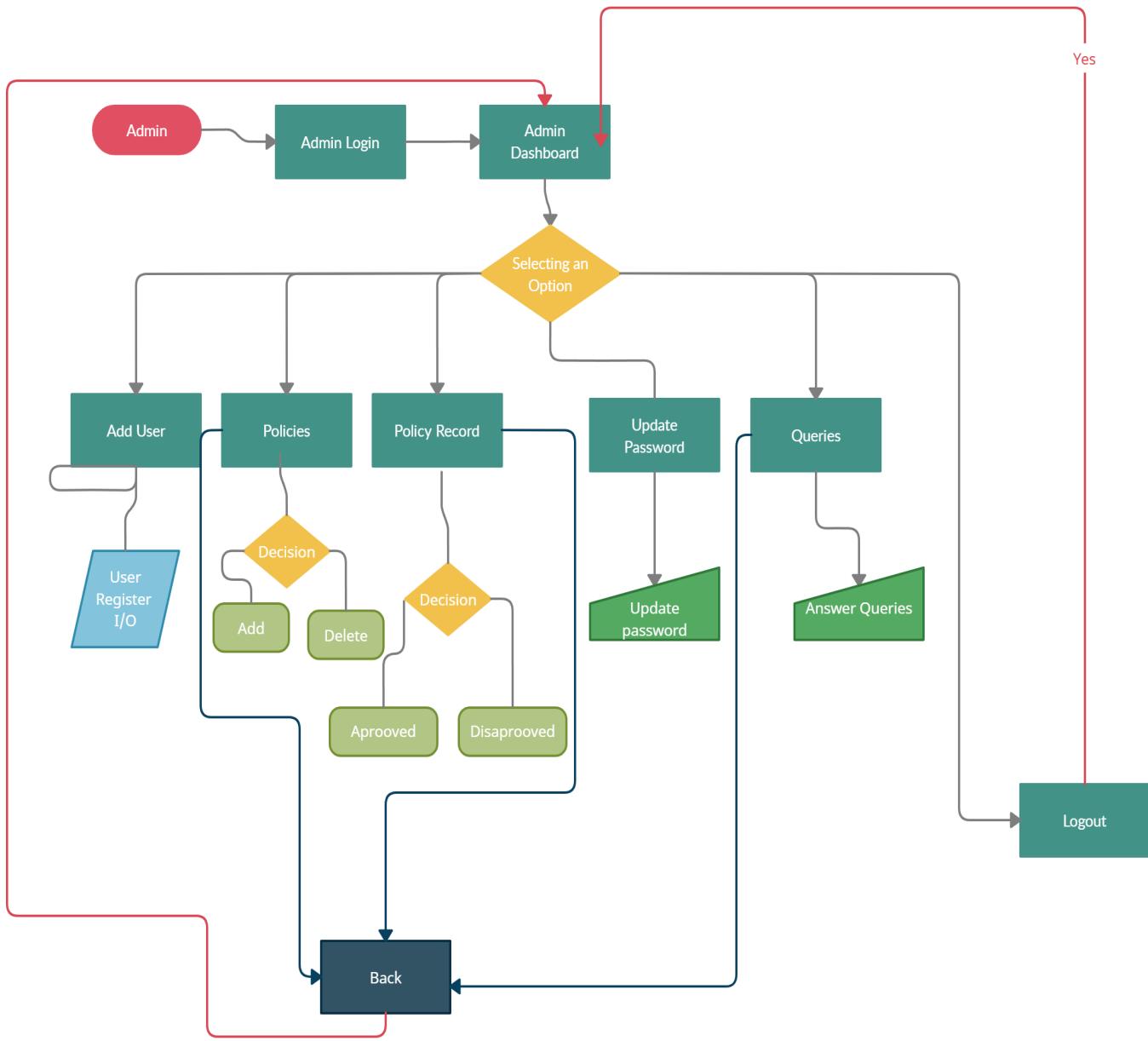
REST API

- API Is To Standardize Data Exchange Between Web Services. Depending On The Type Of API, The Choice Of Protocol Changes.
- REST API Is An Architectural Style For Building Web Services That Interact Via An HTTP Protocol.
- In Spring's Approach To Building Restful Web Services, HTTP Requests Are Handled By A Controller
- REST Stands For Representational State Transfer.
- Postman Is An Application Used For API Testing. It Is An HTTP Client That Tests HTTP Requests, Utilizing A Graphical User Interface, Through Which We Obtain Different Types Of Responses That Need To Be Subsequently Validated.

Admin Functional Requirements



ADMIN FLOW CHART



USER FUNCTIONAL REQUIREMENTS

Create account (no approval required by admin)

Change the password.

Change address, email, mobile.

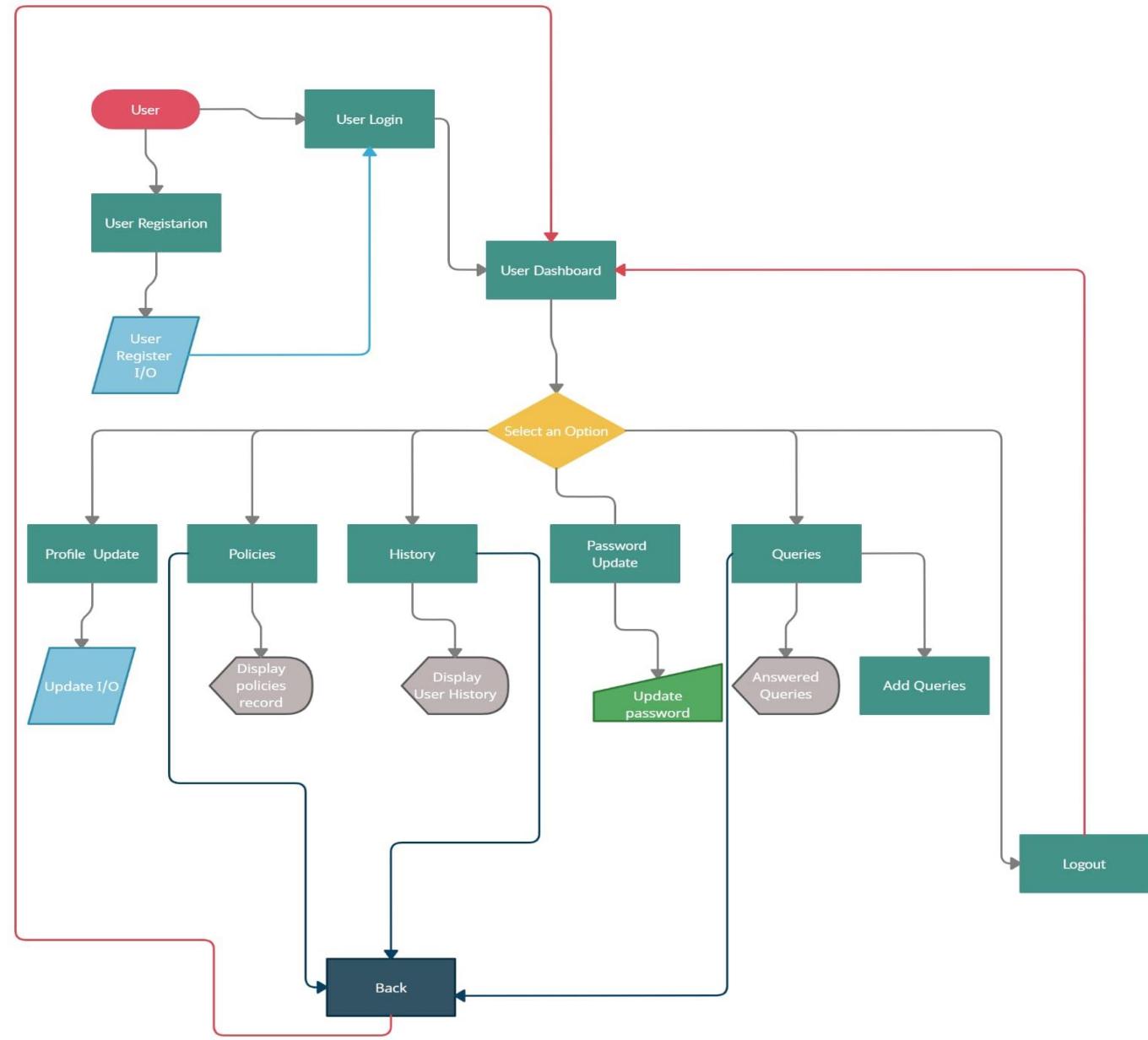
View all policy that are added by admin

If customer likes any policy, then they can apply for it.

When customer applies for any policy, it will go into pending status, admin can approve it.

Customer can check status of his policy under history section8. Customer can ask question from admin.

USER FLOW CHART



IMPLEMENTATION





We provide the best value **insurance**

Simple Steps You Can Take to Improve Your Financial Well-Being for the rest of Your Life



Registration

Your name

Email

Password

Age

Phone

Get Registered

USER PAGES



Insurance Policy

Call Us Now +91-7897568592

• Home Policy Admin User

We provide the best value **insurance**

Simple Steps You Can Take to Improve Your Financial Well-Being for the rest of Your Life

User Login

User Name

Password

Login

Insurance Policy

Call Us Now +91-7897568592

• Home Policy

User

logout

We provide the best value **insurance**

Simple Steps You Can Take to Improve Your Financial Well-Being for the rest of Your Life



localhost:4200/userpolicy

Use Google Chrom... Mphasis Batch 3 ... CodingBat Java GitHub Mail - Thiruweddhi... Unified Access Maven Repository: j... Doc1 - Google Docs SL_Phase03_Assess...

Call Us Now +91-7897568592

Home Policy User logout

Policies

Back

ID	Policy Name	Category	Premium	Maturity	Action
1	Issues in Health	Health	2500	50000	<button>Apply</button>
13	For Insurance	Life	500	25000	<button>Apply</button>
20	For Insurance	Life	500	25000	<button>Apply</button>

Home About Us News Terms & Conditions

f i t l

We provide the best value **insurance**

Simple Steps You Can Take to Improve Your Financial Well-Being for the rest of Your Life

Profile Update

User email

Phone

Age

Update

localhost:4200/userpassword

Use Google Chrom... Mphasis Batch 3 ... CodingBat Java GitHub Mail - Thiruweddhi... Unified Access Maven Repository: j... Doc1 - Google Docs SL_Phase03_Assess...

13 M A S T : | Other bookmarks

Insurance Policy

Call Us Now +91-7897568592

• Home Policy User logout

We provide the best value **insurance**

Simple Steps You Can Take to Improve Your Financial Well-Being for the rest of Your Life

>

Password Updater

Current Password

New Password

Update Password

Queries

Add Query

Back

Id	Question	Customer Name	Answer
----	----------	---------------	--------

Insurance Policy



Call Us Now +91-7897568592

• Home Policy

User

logout

Ask Your Queries

Insurance Policy



1500 Treat Ave, Suite 200 +604-680-9785
San Francisco, CA 94110



ashskgn@mphasis.com



1-800-800-9999 (Support)

your Query

Enter your Query

Submit Query

localhost:4200/userrecord

Use Google Chrom... Mphasis Batch 3 ... CodingBat Java GitHub Mail - Thiruweddhi... Unified Access Maven Repository: j... Doc1 - Google Docs SL_Phase03_Assess... Other bookmarks

Insurance Policy

Call Us Now +91-7897568592

Home Policy User logout

History

Back

Id	Policy Name	Category	Customer Name	Applied Date	Status
25	For Insurance	Life	Venu	2022-10-27	approved

Home About Us News Terms & Conditions

All rights Reserved © Your Company, 2022

Made with ❤ by **Group 4 Team**

f i t l

ADMIN PAGES



Insurance Policy

Call Us Now +91-7897568592

• Home

Policy

Admin

User



We provide the best value **insurance**

Simple Steps You Can Take to Improve Your Financial Well-Being for the rest of Your Life

Admin Login

User Name

Password

Login

Insurance Policy

Call Us Now +91-7897568592

• Home Add User Policies Policy Record Admin logout

We provide the best value **insurance**

Simple Steps You Can Take to Improve Your Financial Well-Being for the rest of Your Life

Admin Registration

Your name

Email

Password

Age

Phone

localhost:4200/addpolicy

Use Google Chrom... Mphasis Batch 3 ... CodingBat Java GitHub Mail - Thiruweddhi... Unified Access Maven Repository: j... Doc1 - Google Docs SL_Phase03_Assess...

13 M F Camera Refresh Home Add User Policies Policy Record Admin logout

Other bookmarks

Insurance Policy

Call Us Now +91-7897568592

- Home
- Add User
- Policies
- Policy Record

We provide the best value **insurance**

Simple Steps You Can Take to Improve Your Financial Well-Being for the rest of Your Life

Policy Details

Policy name

Maturity (tenure)

Premium (cost of policy)

Add Policy

Policy Information

Add Policy

Back

ID	Policy Name	Category	Premium	Maturity	Action
1	Issues in Health	Health	2500	50000	Delete
13	For Insurance	Life	500	25000	Delete
20	For Insurance	Life	500	25000	Delete

Policy Records								
Id		Policy Name	Category	Customer Name	Applied Date	Status		
2	Issues in Health	Health	Hanumann		2022-10-26	approved	<button>Approve</button>	<button>Disapprove</button>
7	For INsurance	Life	Ravi		2022-10-26	approved	<button>Approve</button>	<button>Disapprove</button>
14	Issues in Health	Health	Ravi		2022-10-26	pending	<button>Approve</button>	<button>Disapprove</button>
15	Issues in Health	Health	Ravi		2022-10-26	pending	<button>Approve</button>	<button>Disapprove</button>
16	Issues in Health	Health	Ravi		2022-10-26	pending	<button>Approve</button>	<button>Disapprove</button>
17	Issues in Health	Health	Ravi		2022-10-26	pending	<button>Approve</button>	<button>Disapprove</button>
18	Issues in Health	Health	Ravi		2022-10-26	approved	<button>Approve</button>	<button>Disapprove</button>
21	Issues in Health	Health	Ravi		2022-10-26	pending	<button>Approve</button>	<button>Disapprove</button>
25	For Insurance	Life	Venu		2022-10-27	approved	<button>Approve</button>	<button>Disapprove</button>

localhost:4200/addanswer

Use Google Chrom... Mphasis Batch 3 ... CodingBat Java GitHub Mail - Thiruweddhi... Unified Access Maven Repository: j... Doc1 - Google Docs SL_Phase03_Assess...

13 M A S T : | Other bookmarks

Insurance Policy

Call Us Now +91-7897568592

- Home
- Add User
- Policies
- Policy Record

Admin logout

We provide the best value **insurance**

Simple Steps You Can Take to Improve Your Financial Well-Being for the rest of Your Life

>

Query Resolver

Question Id

Answer

Send Answer

localhost:4200/adminpassword

Use Google Chrom... Mphasis Batch 3 ... CodingBat Java GitHub Mail - Thiruvieddi... Unified Access Maven Repository: j... Doc1 - Google Docs SL_Phase03_Assess... Other bookmarks

Insurance Policy Call Us Now +91-7897568592 Home Add User Policies Policy Record Admin logout

We provide the best value **insurance**

Simple Steps You Can Take to Improve Your Financial Well-Being for the rest of Your Life

>

Password Updater

Current Password

New Password

Update Password

TESTING



User Registration :

The screenshot shows the Postman application interface for a POST request to `localhost:8083/user/register`.

Request Headers: Body (selected), Headers (9), Params, Authorization, Pre-request Script, Tests, Settings, Cookies.

Request Body: JSON (Pretty)

```
1
2   ...
3     "username": "sanjay",
4     "phone": 9791574508,
5     "email": "san@g.com",
6     "password": "kumar",
7     "age": 28
```

Response Headers: 200 OK, 383 ms, 279 B, Save Response.

Response Body: Pretty, Raw, Preview, Visualize, Text (selected).

```
1 user inserted successfully
```

Result Grid:

	username	age	email	password	phone	role
▶	hanuman	22	han@g.com	random	9045674522	user
▶	sanjay	28	san@g.com	kumar	9791574508	user
▶	suraj	24	sur@g.com	surajkumar	90423054235	user
*	HULL	HULL	HULL	HULL	HULL	HULL

User Login :

The screenshot shows the Postman interface for a POST request to `localhost:8083/login`. The request body is set to raw JSON:

```
1 {
2   "username": "sanjay",
3   "password": "sanjaykumar"
4 }
```

The response status is 200 OK with a response body of "Hi sanjay".

Request Headers (9):

- Content-Type: application/json
- User-Agent: PostmanRuntime/7.29.0
- Accept: */*
- Host: localhost:8083
- Connection: keep-alive
- Content-Length: 26
- Accept-Encoding: gzip, deflate
- Cookie: JSESSIONID=4D9E8A8A8A8A8A8A8A8A8A8A8A8A8A8A
- Postman-Token: 3a2a2a2a-2a2a-2a2a-2a2a-2a2a2a2a2a2a

Response Headers (1):

- Content-Type: text/plain; charset=UTF-8

Response Body:

```
1 Hi sanjay
```

Admin Registration:

The screenshot shows the Postman application interface for testing a REST API endpoint.

Request Details:

- Method: POST
- URL: localhost:8083/admin/register
- Body tab is selected, showing JSON data:

```
1
2   {
3     "username": "admin",
4     "phone": 8870353009,
5     "email": "admin@g.com",
6     "password": "admin",
7     "age": 40
8 }
```
- Headers tab shows 9 items.
- Tests and Settings tabs are visible.

Response Details:

- Status: 200 OK
- Time: 17 ms
- Size: 280 B
- Save Response button is present.
- Body tab shows the response message:

```
1 Admin inserted successfully
```
- Pretty, Raw, Preview, Visualize, Text dropdown, and copy/paste icons are available.
- Result Grid table shows the inserted data:

	username	age	email	password	phone	role
▶	admin	40	admin@g.c...	admin	8870353009	admin
▶	hanuman	22	han@g.com	random	9045674522	user
▶	sanjay	28	san@g.com	kumar	9791574508	user
▶	suraj	24	sur@g.com	surajkumar	90423054235	user
*	NULL	NULL	NULL	NULL	NULL	NULL

Admin Login :

POST ▼ localhost:8083/adminlogin Send ▼

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

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

```
1 {  
2   "username": "admin",  
3   "password": "admin28"  
4 }  
5
```

Body Cookies (1) Headers (9) Test Results 200 OK 35 ms 335 B Save Response ▼

Pretty Raw Preview Visualize Text Text ▼

1 Hi admin

Result Grid Filter Rows: Edit:    Export/In

	username	age	email	password	phone	role
▶	admin	40	admin@g.com	admin28	8870353009	admin
	hanuman	22	han@g.com	random	9045674522	user
*	sanjay	28	san@g.com	sanjaykumar	9791574508	user
*	NULL	NULL	NULL	NULL	NULL	NULL

Update Admin password:

PUT localhost:8083/updateAdmin 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": "admin",  
3   "password": "admin28"  
4 }
```

Body Cookies (1) Headers (8) Test Results 200 OK 71 ms 275 B Save Response

Pretty Raw Preview Visualize Text CSV

1 Admin password updated

	username	age	email	password	phone	role
▶	admin	40	admin@g.c...	admin28	8870353009	admin
	hanuman	22	han@g.com	random	9045674522	user
	sanjay	28	san@g.com	kumar	9791574508	user
	suraj	24	sur@g.com	surajkumar	90423054235	user
*	NULL	NULL	NULL	NULL	NULL	NULL

Update User Password:

The screenshot shows the Postman application interface for making a **PUT** request to **localhost:8083/updateUser**. The **Body** tab is selected, containing the following JSON payload:

```
1
2   ...
3   ...
4   ...
```

The payload is intended to be:

```
1
2   {
3     "username": "sanjay",
4     "password": "sanjaykumar"
```

The response status is **200 OK** with a response time of **32 ms** and a size of **274 B**. The response body is:

1 User password updated

	username	age	email	password	phone	role
▶	admin	40	admin@g.c...	admin28	8870353009	admin
	hanuman	22	han@g.com	random	9045674522	user
	sanjay	28	san@g.com	sanjaykumar	9791574508	user
	suraj	24	sur@g.com	surajkumar	90423054235	user
*	NULL	NULL	NULL	NULL	NULL	NULL

Delete User:

The screenshot shows the Postman interface with a successful DELETE request to `localhost:8083/cancel/suraj`.

Request Headers:

- Method: `DELETE`
- URL: `localhost:8083/cancel/suraj`
- Headers (7):
 - Content-Type: `x-www-form-urlencoded`
 - Accept: `raw`

Response Status: 200 OK | 70 ms | 282 B | Save Response

Body:

```
1 Username suraj gets deleted :
```

Result Grid:

	username	age	email	password	phone	role
▶	admin	40	admin@g.c...	admin28	8870353009	admin
	hanuman	22	han@g.com	random	9045674522	user
*	sanjay	28	san@g.com	sanjaykumar	9791574508	user
*		NULL	NULL	NULL	NULL	NULL

Add Policy :

POST <localhost:8083/admin/addPolicy> Send

Params Authorization Headers (8) **Body** • Pre-request Script Tests Settings Cookies

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

```
1
2   "policyno": "5",
3   "category": "motor",
4   "maturity": "100000",
5   "policyname": "accident insurance",
6   "premium": "3000"
```

Body Cookies Headers (8) Test Results 200 OK 34 ms 277 B Save Response

Pretty Raw Preview Visualize Text

1 Policy added sucessfully

Result Grid					
	policyno	category	maturity	policyname	premium
▶	5	health	100000	accident i...	3000
▶	6	motor	40000	bike insur...	3000
*	NULL	NULL	NULL	NULL	NULL

Delete policy:

DELETE localhost:8083/admin/deletePolicy/:id Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies

KEY	VALUE	DESCRIPTION	...	Bulk Edit
Key	Value	Description		

Path Variables

KEY	VALUE	DESCRIPTION	...	Bulk Edit
id	6	Description		

Body Cookies Headers (8) Test Results 200 OK 61 ms 360 B Save Response

Pretty Raw Preview Visualize JSON

```
1 [ ] {  
2   "policyno": 5,  
3   "policyname": "accident insurance",  
4   "category": "health",  
5   "premium": "3000",  
6   "maturity": "100000"  
7 } [ ]
```

Result Grid Filter Rows: Edit: Export

	policyno	category	maturity	policyname	premium
▶	5	health	100000	accident i...	3000
*	NULL	NULL	NULL	NULL	NULL

View policies:

GET localhost:8083/policies

Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies

Query Params

KEY	VALUE	DESCRIPTION	...	Bulk Edit
Key	Value	Description		

Body Cookies Headers (8) Test Results

Pretty Raw Preview Visualize JSON

```
1
2
3   "policyno": 5,
4   "policyname": "accident insurance",
5   "category": "health",
6   "premium": "3000",
7   "maturity": "100000"
8 },
9 {
10   "policyno": 6,
11   "policyname": "bike insurance",
12   "category": "motor",
13   "premium": "3000",
14   "maturity": "40000"
15 }
```

Result Grid | Filter Rows: | Edit: | Export/Import: |

policyno	category	maturity	policyname	premium
5	health	100000	accident i...	3000
6	motor	40000	bike insur...	3000
NULL	NULL	NULL	NULL	NULL

Register policy:

POST <localhost:8083/policyregister> Send

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

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

```
1
2     "policynname": "bike insurance",
3     "category": "motor",
4     "date": "2022-10-25"
5
6
```

Body Cookies (1) Headers (8) Test Results 200 OK 62 ms 277 B Save Response

Pretty Raw Preview Visualize Text

1 sanjay registered policy

recordno	category	customername	date	policynname	status
3	health	sanjay	2022-10-24	accident insurance	pending
4	motor	sanjay	2022-10-24	bike insurance	pending
*	HULL	HULL	HULL	HULL	HULL

View Customer Record:

GET localhost:8083/customer/record Send

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

Query Params

KEY	VALUE	DESCRIPTION	...	Bulk Edit
Body	Cookies (1)	Headers (8)	Test Results	200 OK 52 ms 513 B Save Response

Pretty Raw Preview Visualize JSON ↻

```
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content

	recordno	category	customername	date	policyname	status
▶	3	health	sanjay	2022-10-24	accident insurance	pending
▶	4	motor	sanjay	2022-10-24	bike insurance	pending
*	HULL	HULL	HULL	HULL	HULL	HULL

Admin Approved:

PUT localhost:8083/approved/:id

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

Query Params

KEY	VALUE	DESCRIPTION
Key	Value	Description

Path Variables

KEY	VALUE	DESCRIPTION
id	3	Description

Body Cookies (1) Headers (8) Test Results

200 OK 33 ms 518 B Save

Pretty Raw Preview Visualize JSON

```
1 {  
2   "recordno": 3,  
3   "policyname": "accident insurance",  
4   "category": "health",  
5   "customername": "sanjay",  
6   "status": "approved",  
7   "date": "2022-10-24"  
8 },  
9
```

Result Grid Filter Rows: Edit: Export/Import: Wrap Cell Content:

	recordno	category	customername	date	policyname	status
▶	3	health	sanjay	2022-10-24	accident insurance	approved
▶	4	motor	sanjay	2022-10-24	bike insurance	pending
*	NULL	NULL	NULL	NULL	NULL	NULL

Admin Disapproved:

PUT <localhost:8083/disapproved/:id> Send

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

Query Params

KEY	VALUE	DESCRIPTION	...	Bulk Edit
Key	Value	Description	...	

Path Variables

KEY	VALUE	DESCRIPTION	...	Bulk Edit
id	4	Description	...	

Body Cookies (1) Headers (8) Test Results

200 OK 33 ms 518 B Save Response

Pretty Raw Preview Visualize JSON

```
0   "customername": "sanjay",
1   "status": "approved",
2   "date": "2022-10-24"
3,
4
5
6
7
8
9
10
11   "recordno": 4,
12   "policynname": "bike insurance",
13   "category": "motor",
14   "customername": "sanjay",
15   "status": "disapproved",
16   "date": "2022-10-24"
17
18
```

Result Grid Filter Rows: Edit: Export/Import: Wrap Cell Content:

	recordno	category	customername	date	policynname	status
▶	3	health	sanjay	2022-10-24	accident insurance	approved
▶	4	motor	sanjay	2022-10-24	bike insurance	disapp...
▶	NULL	NULL	NULL	NULL	NULL	NULL

Add Question:

The screenshot shows the Postman interface for making a POST request to `localhost:8083/addQuestion`. The request body is set to `raw` JSON, containing the following data:

```
1
2   "question": "How many days it will take to get Approved"
3
```

The resulting database row in the Result Grid is:

qid	answer	customername	question
3	Not Answered	sanjay	How many days it will take to get Approved
*	NULL	NULL	NULL

Update Answer:

The screenshot shows the Postman application interface for making a PUT request to update an answer.

Request Details:

- Method: PUT
- URL: `localhost:8083/approved/:id`
- Body tab selected
- Body content (JSON):

```
1 {  
2   "qid": 3,  
3   "answer": "It will take one or two days"  
4 }  
5
```

Response Details:

- Status: 200 OK
- Time: 50 ms
- Size: 384 B
- Save Response button

Body Content (Pretty Print):

```
1 {  
2   "qid": 3,  
3   "question": "How many days it will take to get Approved",  
4   "answer": "It will take one or two days",  
5   "customername": "sanjay"  
6 }  
7  
8 }
```

Result Grid:

	qid	answer	customername	question
▶	3	It will take on...	sanjay	How many days it will take to get Approved
*	NUL	NUL	NUL	NUL

AUTOMATED TESTING WITH SELENIUM

- Selenium Is One Of The Most Widely Used Open Source Web UI (User Interface) Automation Testing Suite. It Was Originally Developed By Jason Huggins
- In 2004 As An Internal Tool At Thought Works. Selenium Supports Automation Across Different Browsers, Platforms And Programming Languages.
- Selenium Can Be Easily Deployed On Platforms Such As Windows, Linux, Solaris And Macintosh. Moreover,
- It Supports OS (Operating System) For Mobile Applications Like Ios, Windows Mobile And Android.
- Selenium Supports A Variety Of Programming Languages Through The Use Of Drivers Specific To Each Language.
- Currently, Selenium Web Driver Is Most Popular With Java And C#.
- Selenium Test Scripts Can Be Coded In Any Of The Supported Programming Languages And Can Be Run Directly In Most Modern Web Browsers.
- Browsers Supported By Selenium Include Internet Explorer, Mozilla Firefox, Google Chrome And Safari.

SELENIUM WEB ELEMENTS

- Web Elements:
 - Identify The Tag Unique
 - Wd.FindElement(by.WebElement("identifier"));
 - Id,
 - name,
 - cssselector,
 - linktext,
 - partiallinktext,
 - tagname,
 - classname,
 - xpath
 - Field Level: Id, name, cssselector, tagname, classname, xpath
 - Best: Id, name, xpath
 - Hyperlink Level: Linktext, partiallinktext
 - Best: Linktext

TESTNG

- Testng Is A Powerful Testing Framework, An Enhanced Version Of Junit Which Was In Use For A Long Time Before Testng Came Into Existence. NG Stands For 'Next Generation'.
- Testng Framework Provides The Following Features –
 - Annotations Help Us Organize The Tests Easily.
 - Flexible Test Configuration.
 - Test Cases Can Be Grouped More Easily.
 - Parallelization Of Tests Can Be Achieved Using Testng.
 - Support For Data-driven Testing.
 - Inbuilt Reporting.

- @Aftertest
 - The Annotated Method Will Be Run After all The Test Methods Belonging To The classes Inside The <Test> Tag Have Run.
- @Beforetest
 - The Annotated Method Will Be Run before Any Test method Belonging To the Classes Inside The <Test> Tag is run.
- @Test
 - Marks A Class Or A Method As Part Of The Test.

```
Something similar to org.seleniumhq.selenium.devtools v0.0.4.17+ where the version ( v00 ) matches the version of the
chromium-based browser you're using and the version number of the artifact is the same as Selenium's.

http://localhost:4200/addpolicy
http://localhost:4200/admin
http://localhost:4200/admin
Dropdown works
http://localhost:4200/user|
http://localhost:4200/user
http://localhost:4200/user
User Login
PASSED: userLoginfail
PASSED: userPasswordUpdate
PASSED: userDashboard
PASSED: adminadduser
PASSED: userLoginsuccess
PASSED: userRegistration
PASSED: adminLoginfail
PASSED: adminAddPolicy
PASSED: profileupdate
PASSED: aminLoginsuccess
PASSED: history
PASSED: logout
PASSED: userQuery

=====
Default test
Tests run: 13, Failures: 0, Skips: 0
=====

=====
Default suite
Total tests run: 13, Passes: 13, Failures: 0, Skips: 0
```

@ Javadoc □ Console □ Results of running class NewTest ×

Search: □ Passed: 13 □ Failed: 0 □ Skipped: 0 Tests: 1/1 Methods: 13 (46153 ms)

All Tests Failed Tests Summary

Default suite (13/0/0/0) (37.428 s)

Default test (37.428 s)

com.example.NewTest

adminAddPolicy (4.861 s)
adminLoginfail (1.153 s)
adminadduser (4.96 s)
aminLoginsuccess (1.009 s)
history (0.726 s)
logout (5.727 s)
profileupdate (1.884 s)
userDashboard (1.948 s)
userLoginfail (1.002 s)
userLoginsuccess (0.835 s)
userPasswordUpdate (5.898 s)
userQuery (5.958 s)
userRegistration (1.467 s)

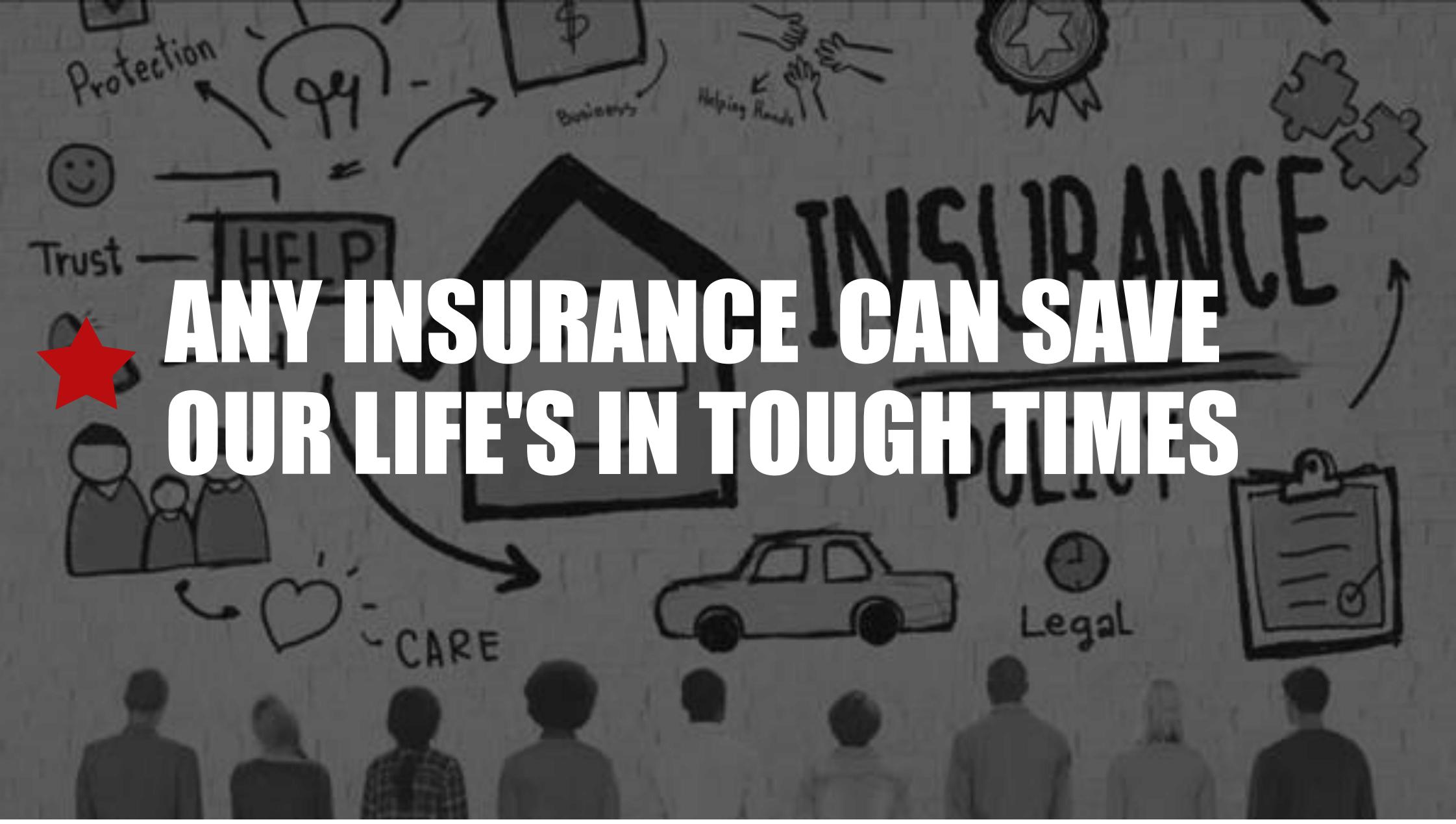
Failure Exception



Insurance Policy

- Insurance Is A Large Investment And You Will Most Likely Purchase Multiple Policies Throughout Your Lifetime.
- It Is Essential That You Know What Each Type Of Insurance Covers And How It Works So You Can Make The Best Decision About What To Buy.
- Do Not Base Your Decision On Just What Is Cheapest, But Look At What It Provides.
- Take The Time To Shop Around And Find The Right Insurance For Your Situation.
- It Can Save Them From Thousands Or More Dollars In Unplanned Expenses When Unexpected Situations Arise.

CONCLUSION



**ANY INSURANCE CAN SAVE
OUR LIFE'S IN TOUGH TIMES**

THANK YOU

