

INTERVIEWPANEL

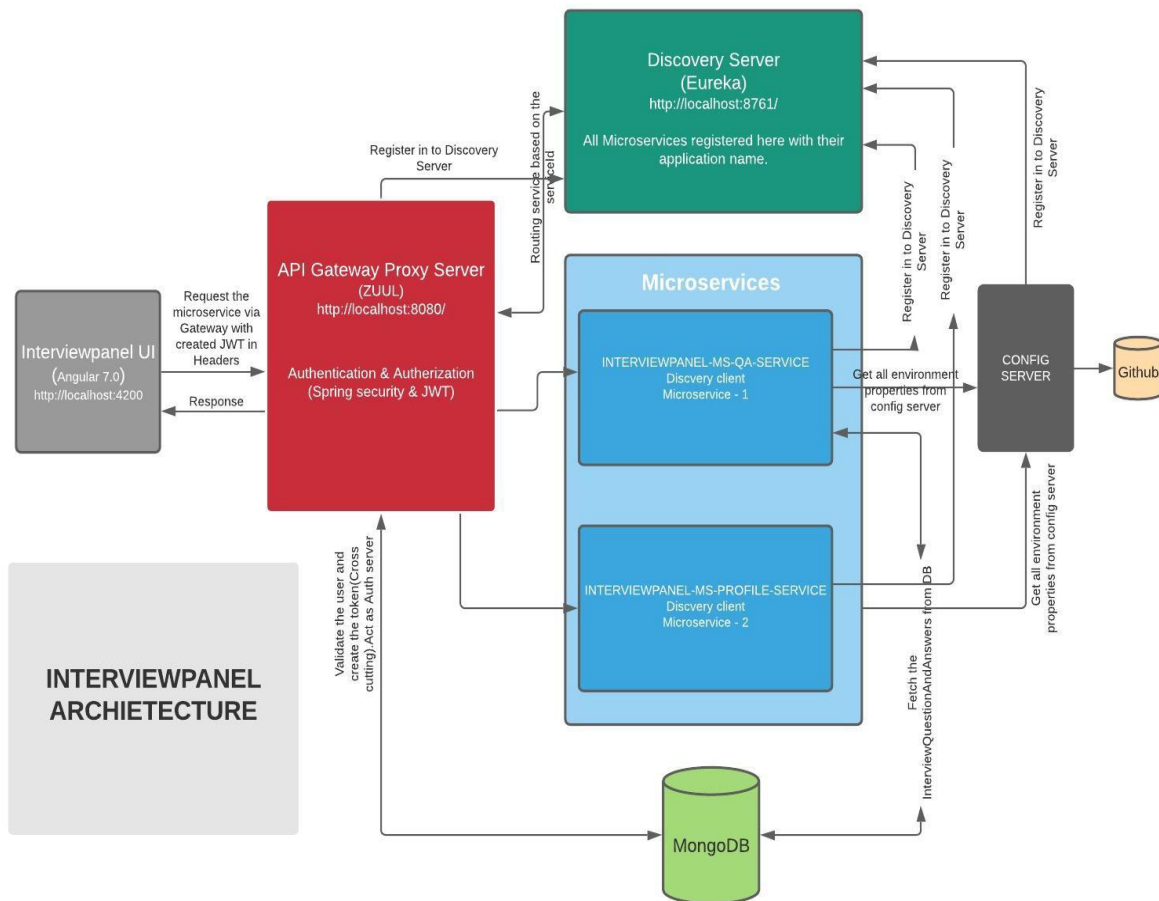


Fig: Interviewpanel - Architecture-Diagram

Project Local setup :

Frontend Development:

Git Repository:

> <https://github.com/karpomkarpipom/interviewpanel-ui>

Development Software and Tool:

> IDE - Visual Studio Code
> Angular **7.0**, Node.js, Angular CLI, <https://www.geeksforgeeks.org/angular-7-installation/>

Backend Development:

Git Repository:

> Gateway-server : <https://github.com/karpomkarpipom/interviewpanel-api-gateway-server>
> Discovery-server : <https://github.com/karpomkarpipom/interviewpanel-discovery-server>
> Question and Answer Service: <https://github.com/karpomkarpipom/interviewpanel-ms-questionandanswer-service>

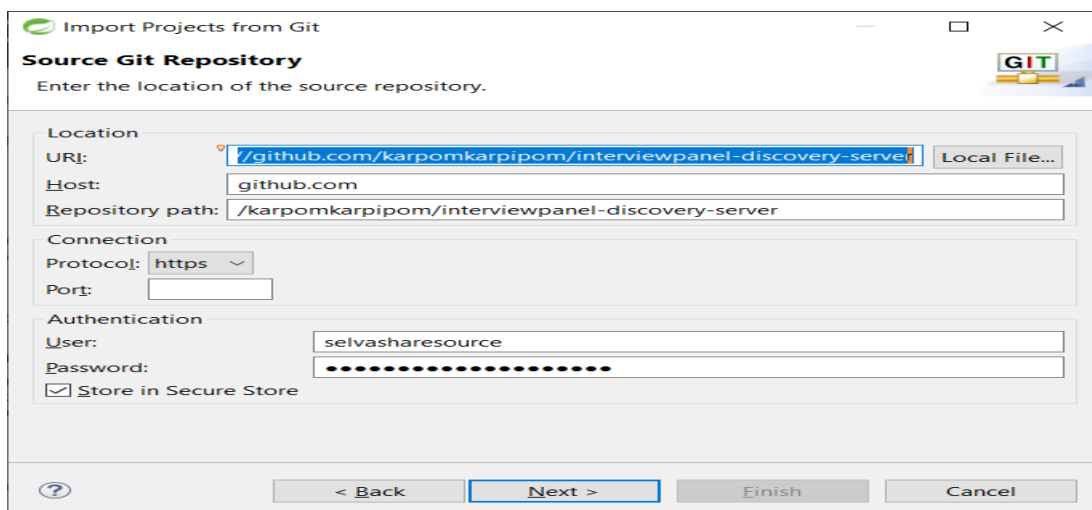
Development Software and Tool:

> IDE – STS or IDEA IntelliJ (I am using Version: 3.9.12.RELEASE)
> JDK 1.8
> Mongo db (Any latest version) installed and running in the port of 27017
> MongoDB Compass Community (GUI)
> Restclient: Postman or Insomnia

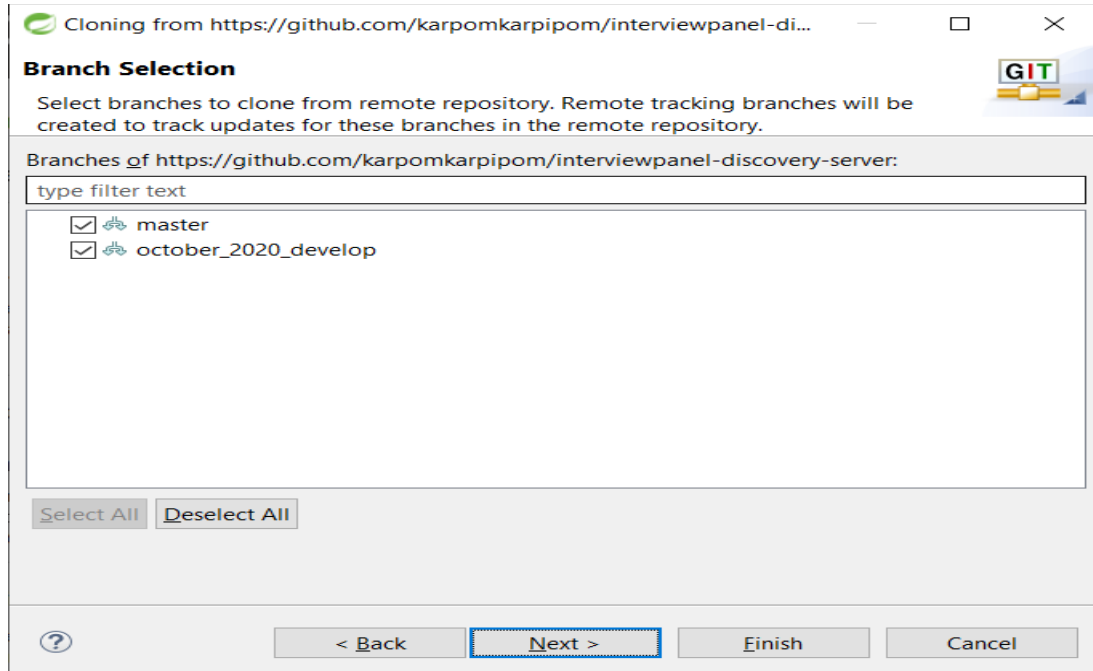
Backend set steps:

STEP 1 > File> import > Git > Projects from Git > Clone URI

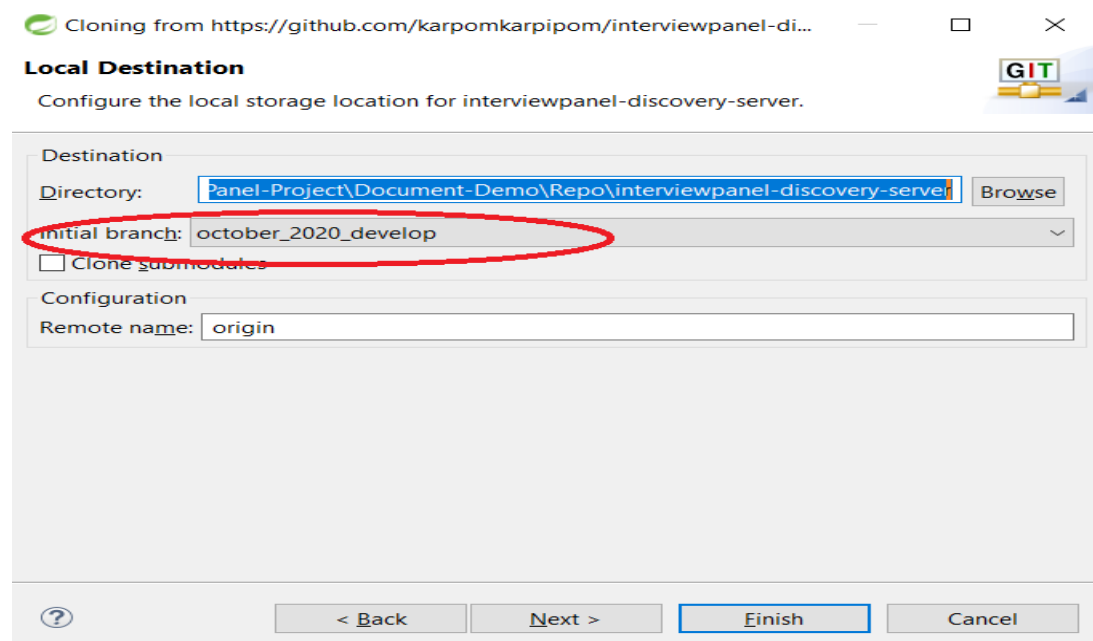
> In URI Place <https://github.com/karpomkarpipom/interviewpanel-discovery-server>
> **User**:selvasharesource **password**: [selvasharesource@789](#)



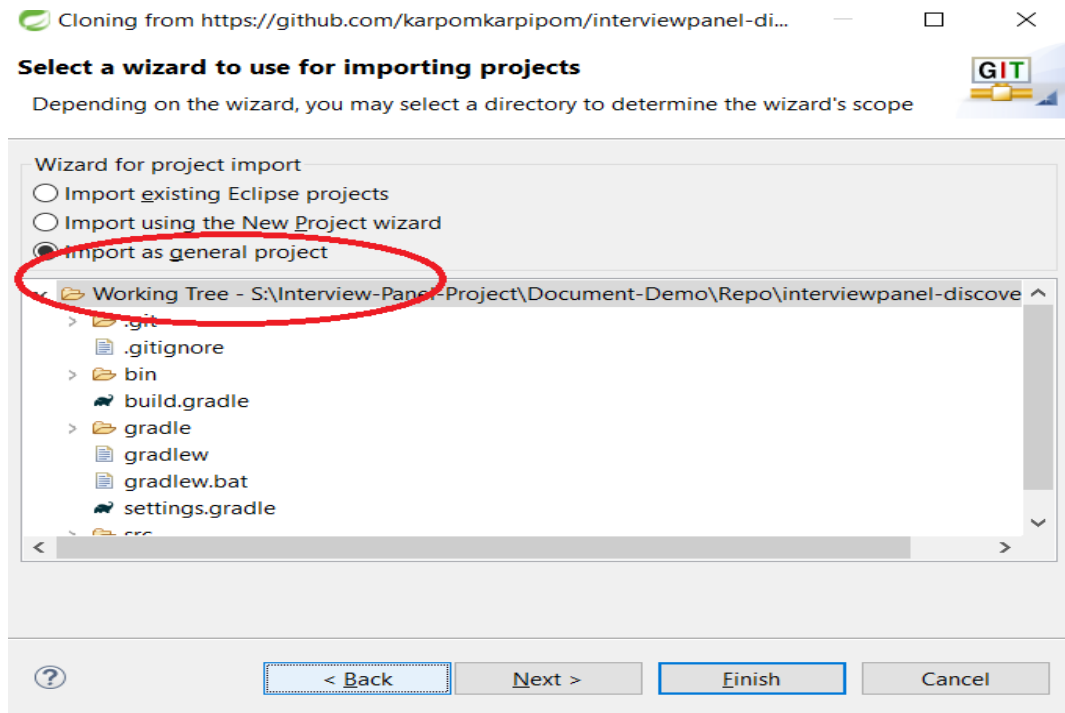
STEP 2 > select next



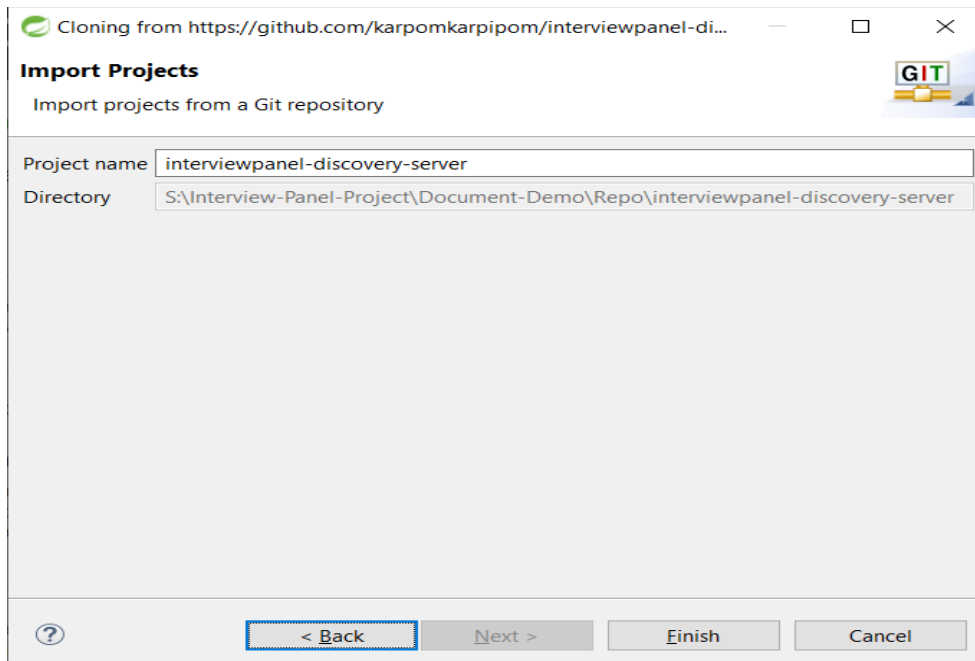
STEP 3 > Select your local directory and initial branch as **october_2020_develop**



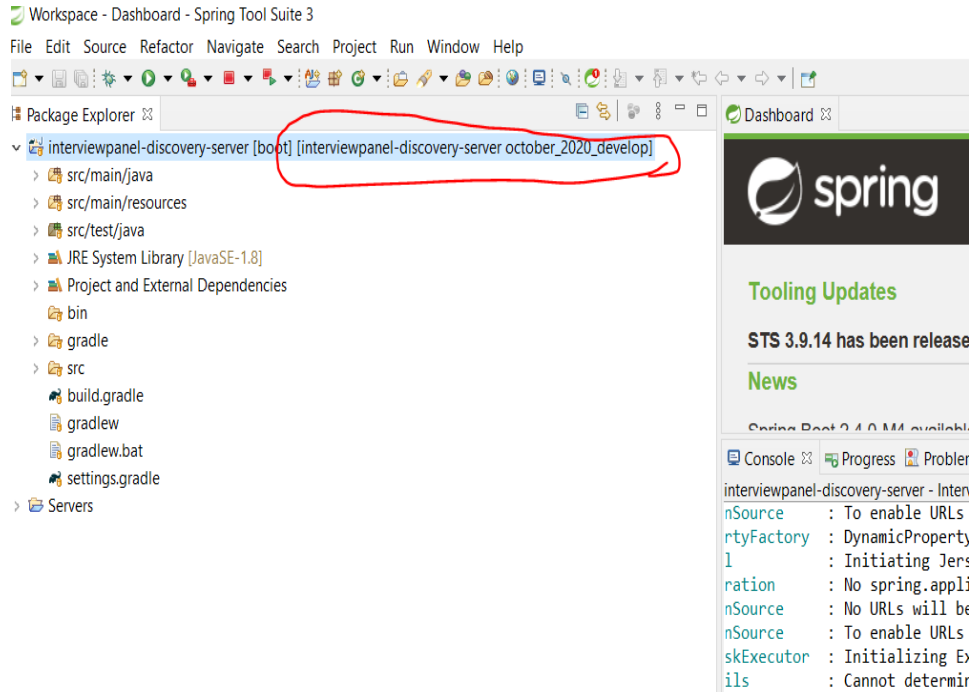
STEP 4> Select import as general project



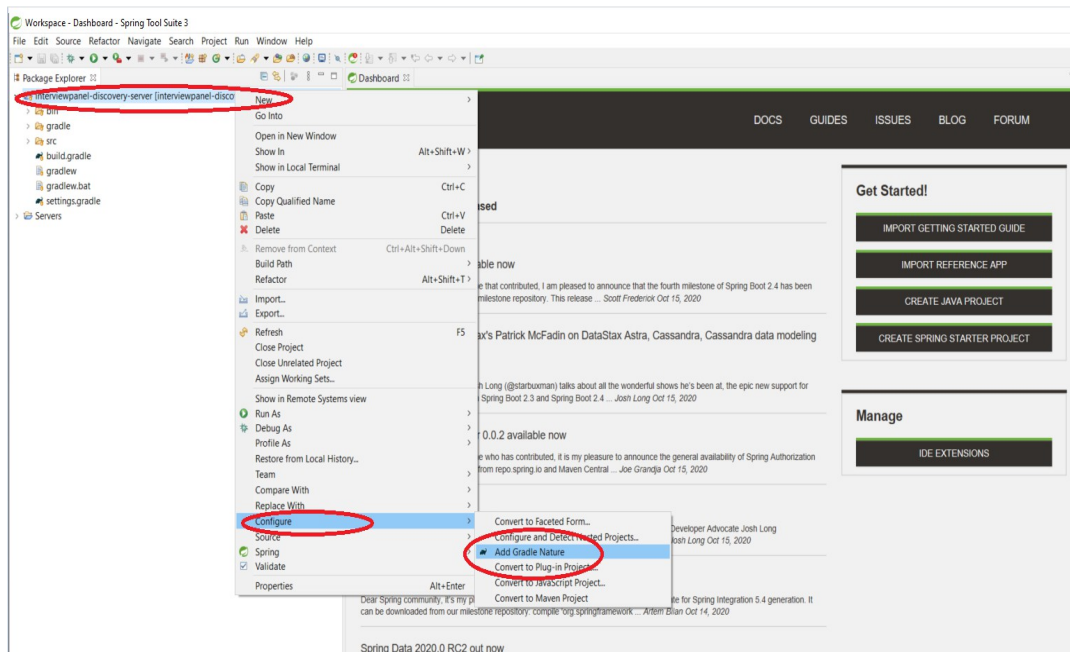
STEP 5 > Finish



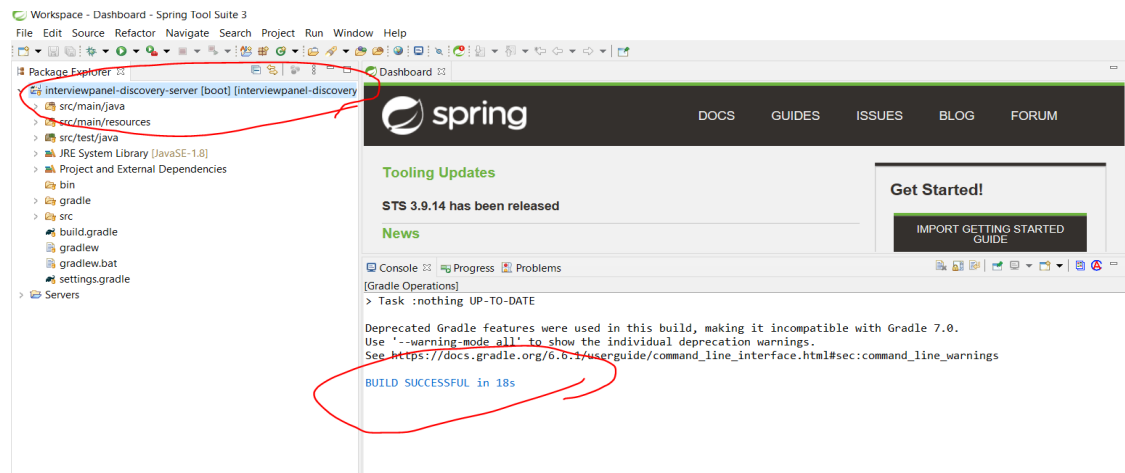
STEP 6 > Make sure remote branch `october_2020_develop`



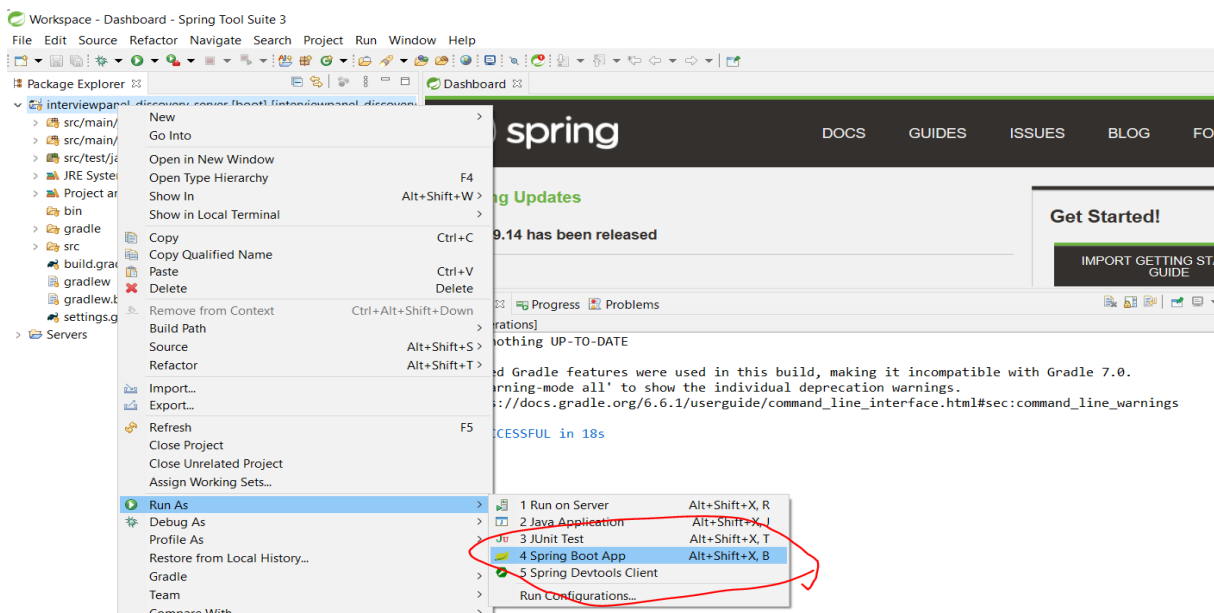
STEP 7> Right click > Configure > Add Gradle Nature



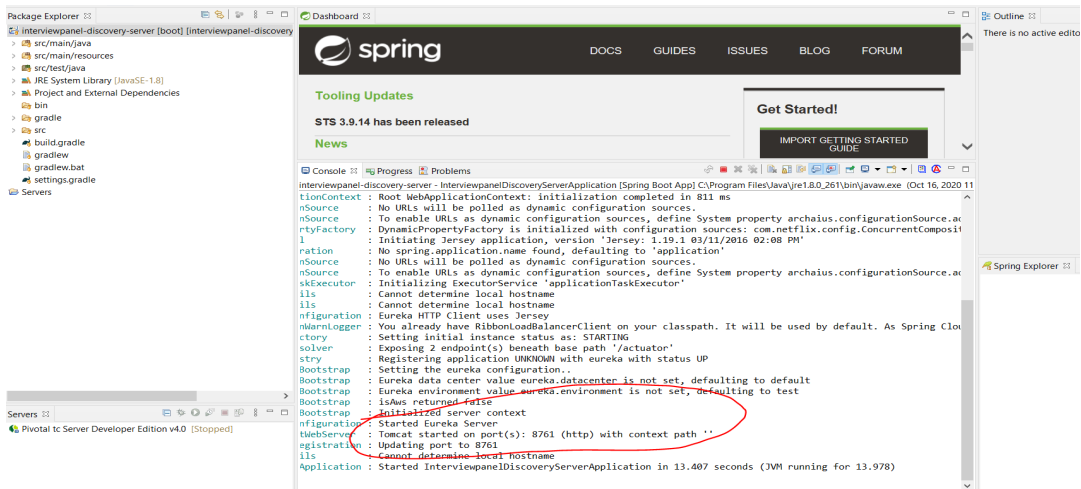
STEP 8 > After success build you will see the screen



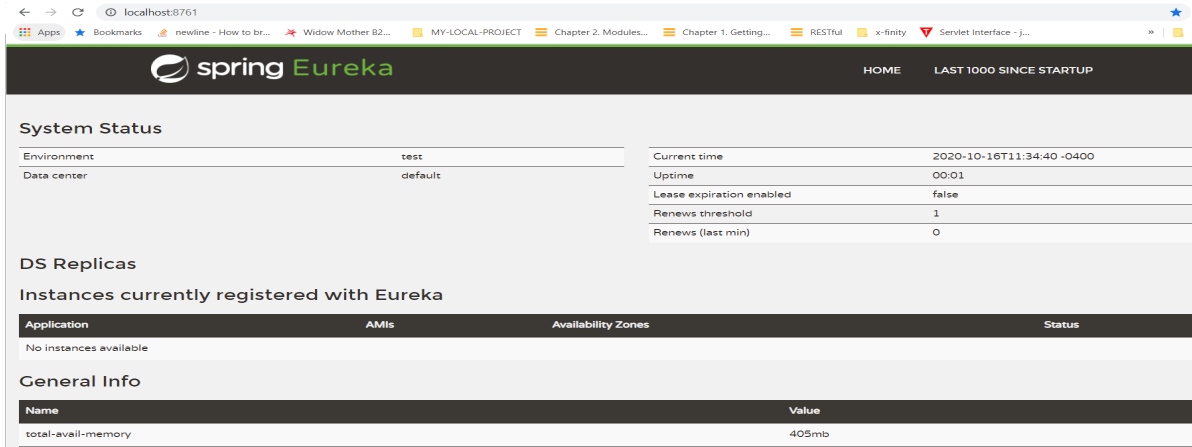
STEP 9 > Rightclick on the project> Run As > Spring Boot App



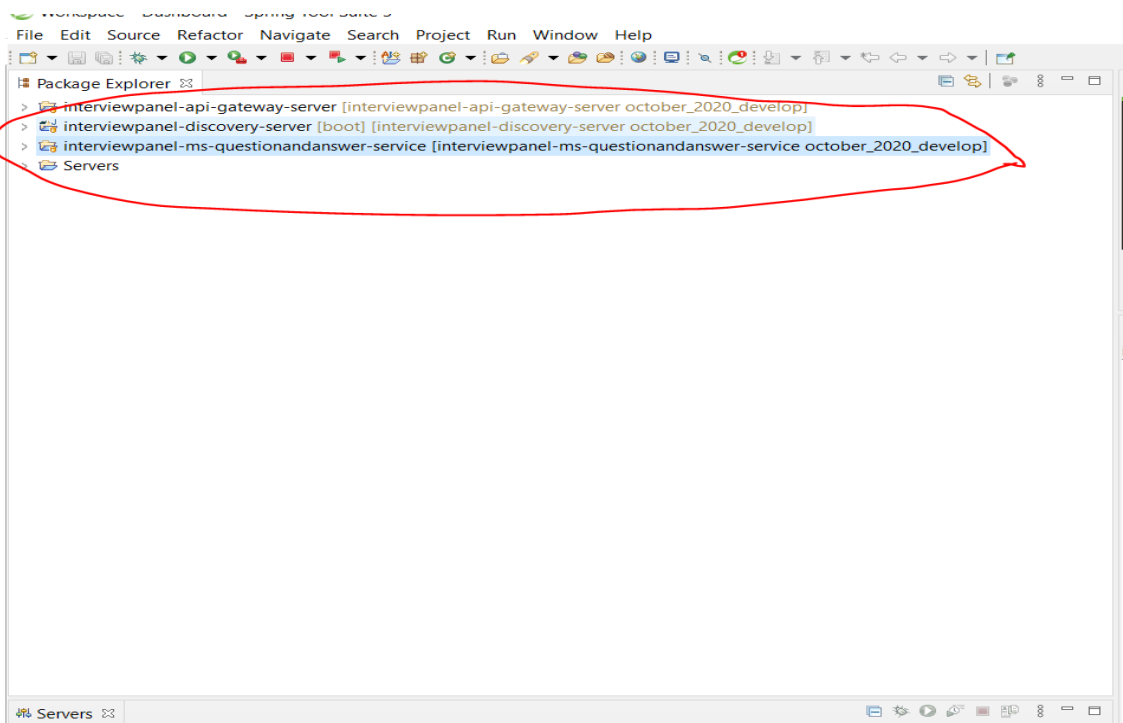
```
> Eureka Server started successfully!
```



> Now you can access <http://localhost:8761/>



Follow the **above 9 steps** for other two services (Gateway and Microservice) till Run them successful



> If all services are up and running then you will see the below screen in <http://localhost:8761/>
Eureka server:

The screenshot shows the Eureka server dashboard at localhost:8761. The browser address bar is circled in red. The page title is "DS Replicas". Below it, the text "Instances currently registered with Eureka" is displayed. A table lists the registered instances, with the first two rows circled in red:

Application	AMIs	Availability Zones	Status
INTERVIEWPANEL-API-GATEWAY-SERVER	n/a (1)	(1)	UP (1) - localhost:interviewpanel-api-gateway-server:8080
INTERVIEWPANEL-MS-QUESTIONANDANSWER-SERVICE	n/a (1)	(1)	UP (1) - localhost:interviewpanel-ms-questionandanswer-service:8081

Below the table, the "General Info" section is visible, showing various system metrics in a table:

Name	Value
total-avail-memory	405mb
environment	test
num-of-cpus	8
current-memory-usage	161mb (39%)
server-uptime	00:26
registered-replicas	
unavailable-replicas	
available-replicas	

Checking Backend service working by Restclient - POSTMAN

API Gateway URL : <http://localhost:8080/>

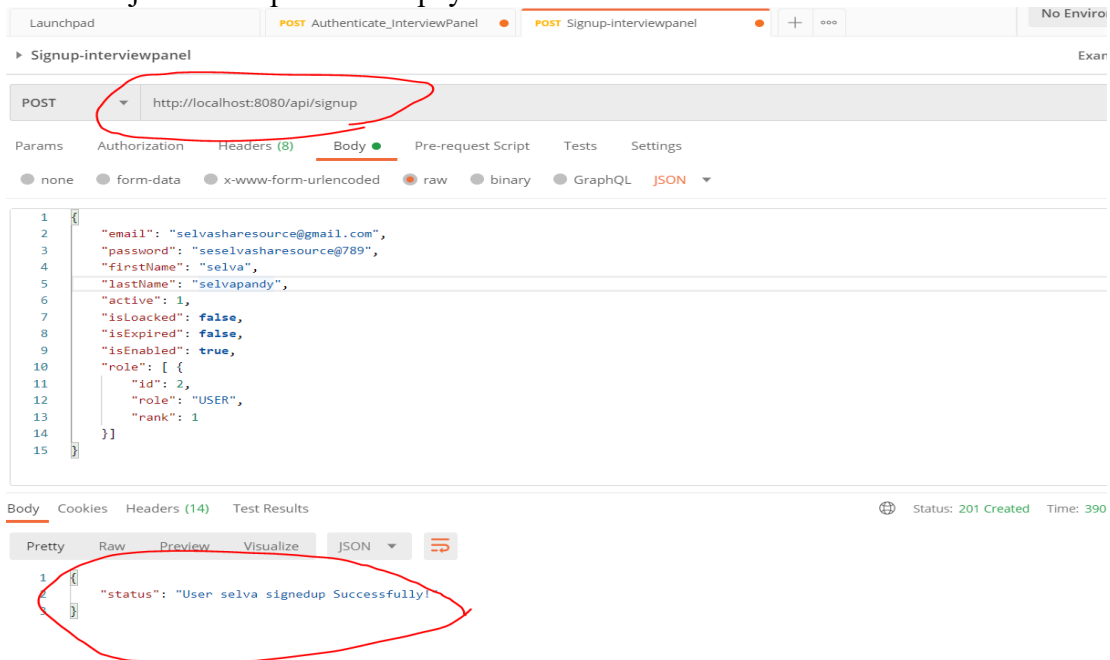
First Signup the user : <http://localhost:8080/api/signup>

Below is the sample Payload, Create with your email and password

Payload:

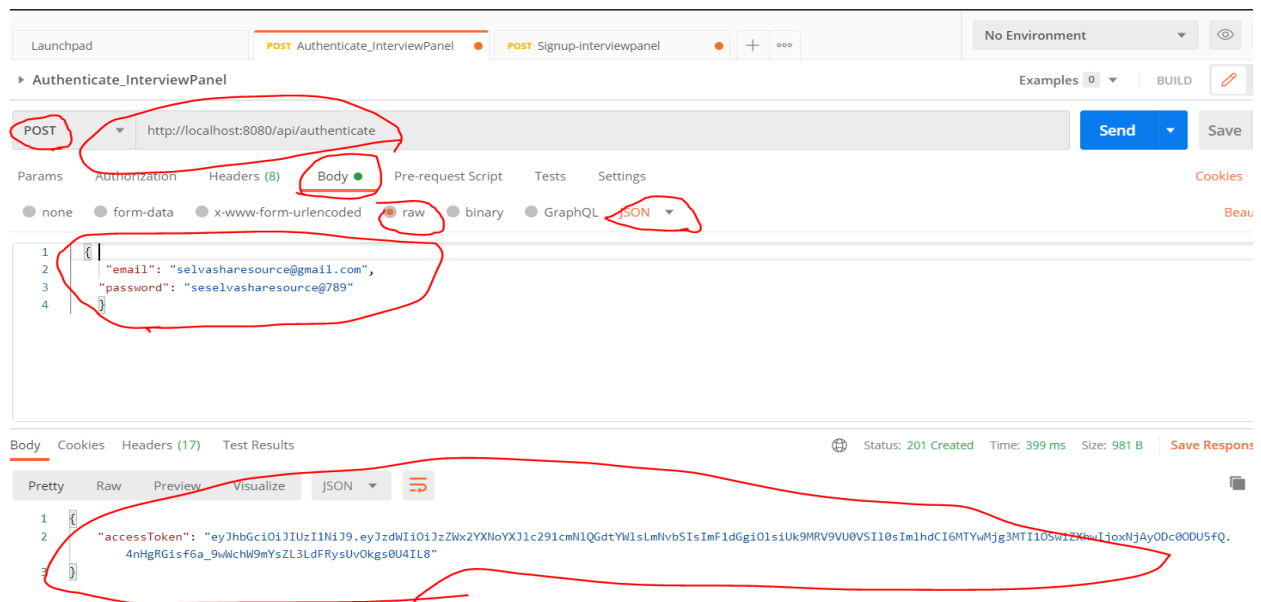
```
{
  "email": "selvasharesource@gmail.com",
  "password": "seselvasharesource@789",
  "firstName": "selva",
  "lastName": "selvapandy",
  "active": 1,
  "isLoacked": false,
  "isExpired": false,
  "isEnabled": true,
  "role": [ {
    "id": 2,
    "role": "USER",
    "rank": 1
  } ]
}
```

Choose : raw and json – and pass above payload



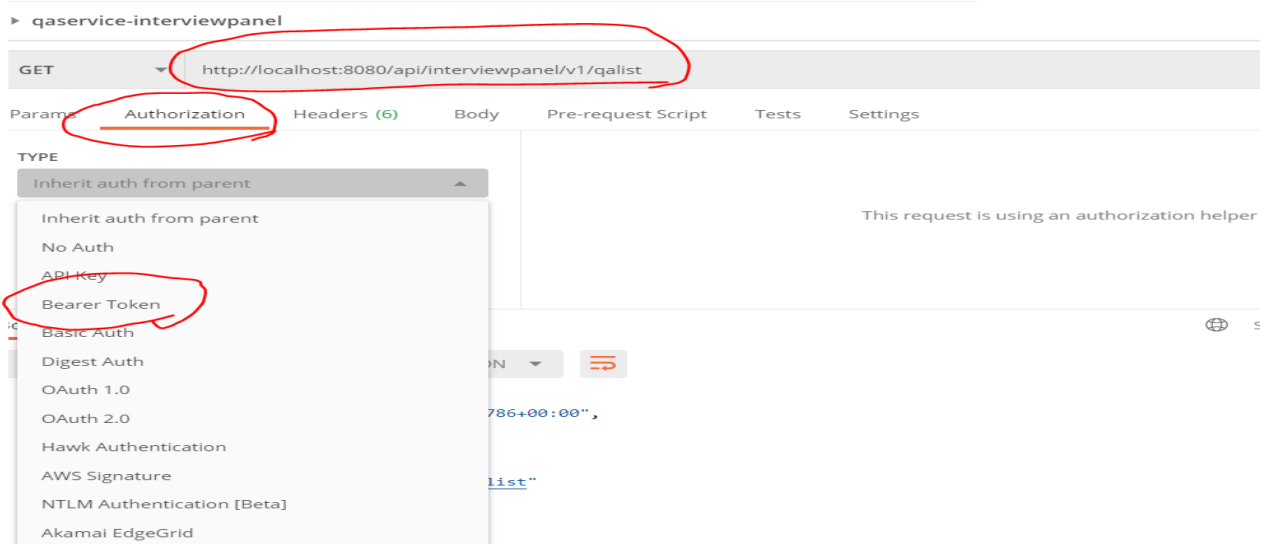
Step 2 : Call authenticate end point to get token: <http://localhost:8080/api/authenticate> with payload to call the Microservice

```
{
  "email": "selvasharesource@gmail.com",
  "password": "seselvasharesource@789"
}
```



Step 3: Call qa service by passing the Bearer token <http://localhost:8080/api/interviewpanel/v1/qalist>

Select the Bearer token:



Successfully retrieved the data

The screenshot shows the Postman interface for a GET request to `http://localhost:8080/api/interviewpanel/v1/qalist`. The request is authenticated with a Bearer Token. The response status is 200 OK, and the body is a JSON array of interview questions and answers. Red circles highlight the URL, the token, and the first element of the JSON array.

Request:

- Method: GET
- URL: `http://localhost:8080/api/interviewpanel/v1/qalist`
- Authorization: Bearer Token

Response:

```
[{"questionPostedBy": "Selvapandy Moorthy", "employerName": "Firstdata", "questionAnswers": [{"technology": "Spring", "questionAsked": "What is Autowired?", "givenAnswer": "Autowiring feature of spring framework enables you to inject the object dependency implicitly."}], [{"technology": "Spring", "questionAsked": "What is Qualifier?"}]}
```

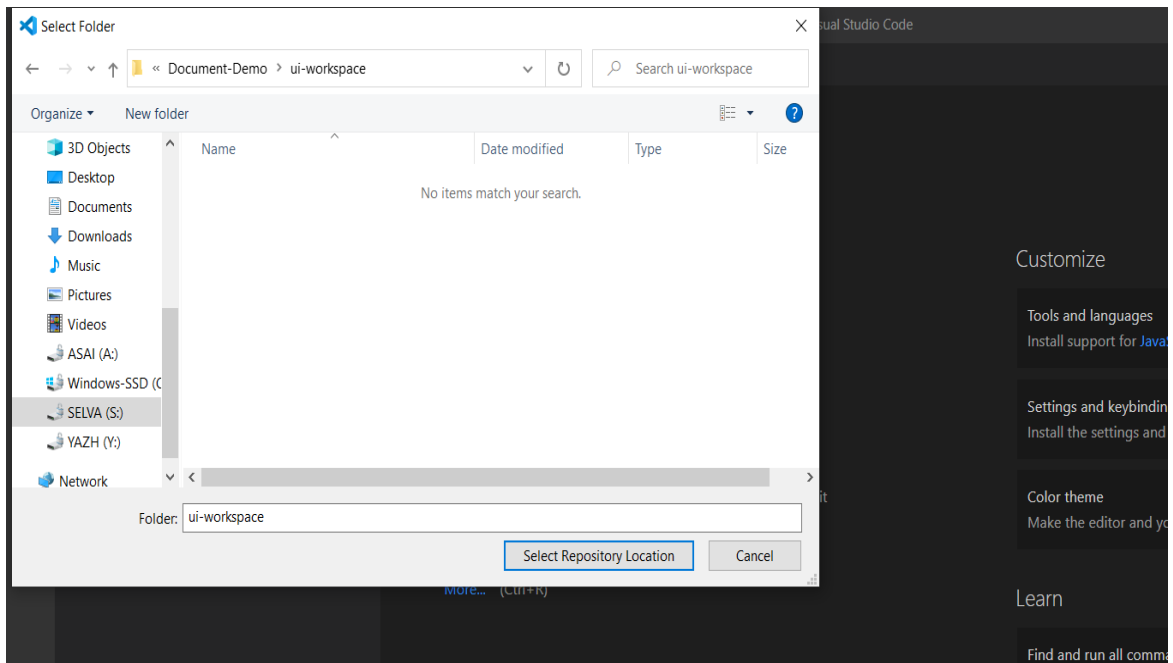
-----Backend Successful done!-----

Frontend Setup :

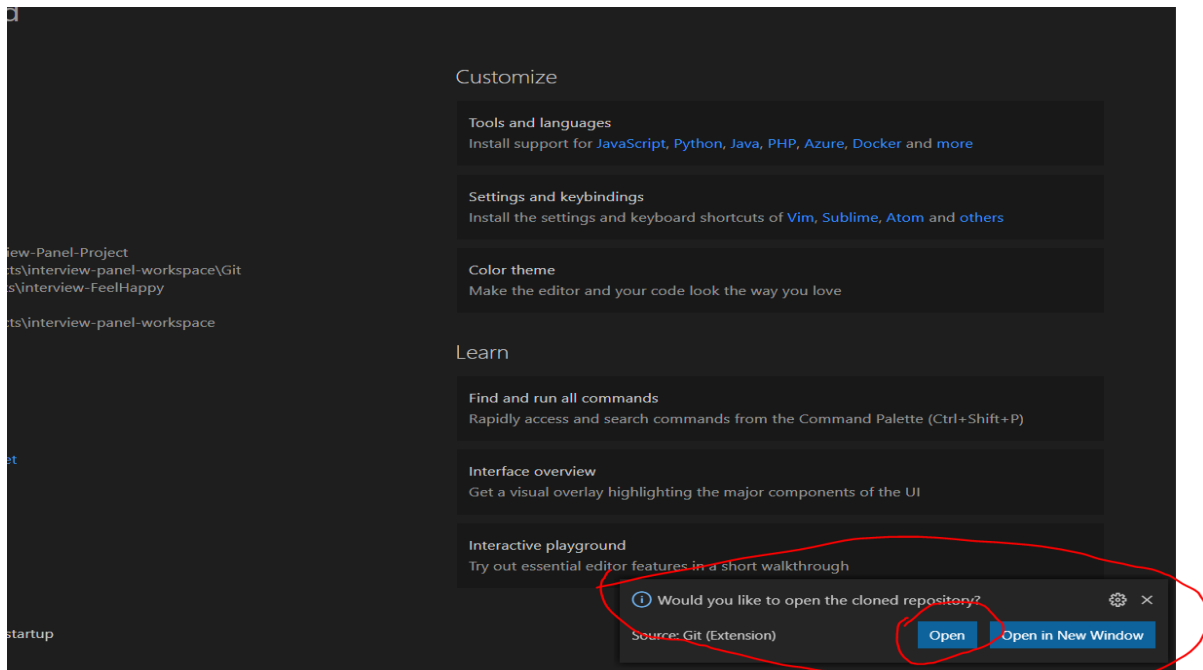
Step 1> Visual Studio Code

The screenshot shows the Visual Studio Code interface. In the Source Control panel on the left, the 'Clone Repository' button is highlighted with a red circle. In the Welcome dialog on the right, the 'Clone from URL' option is highlighted with a red circle, and the URL `https://github.com/karpomkarpom/interviewpanel-ui` is entered in the text field.

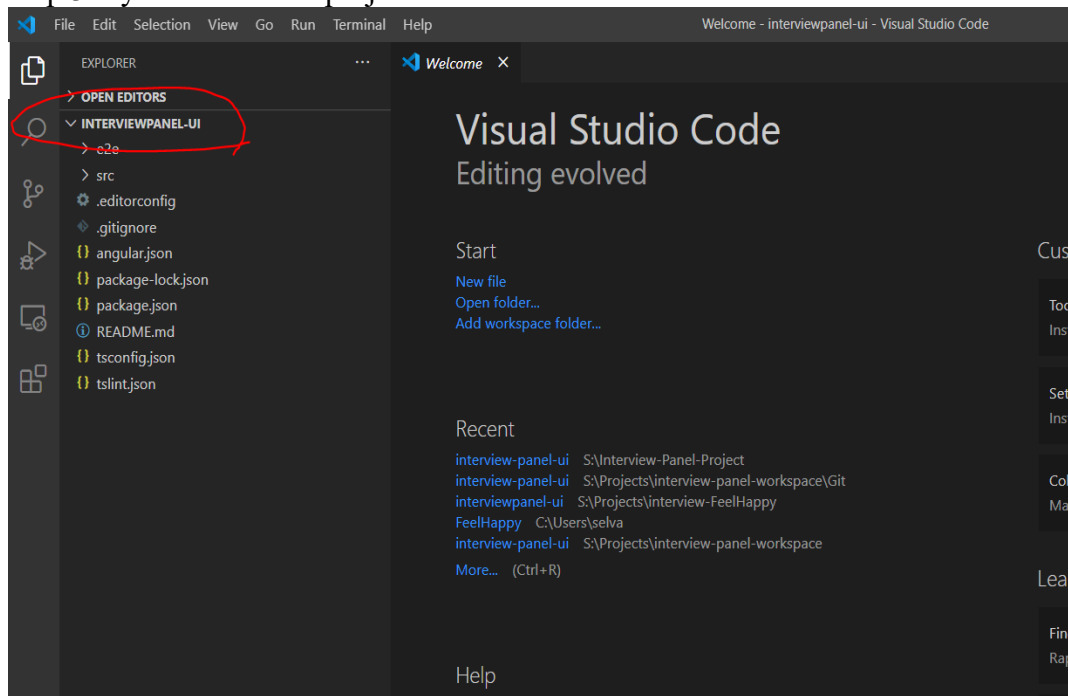
Step 2 > select some local folder



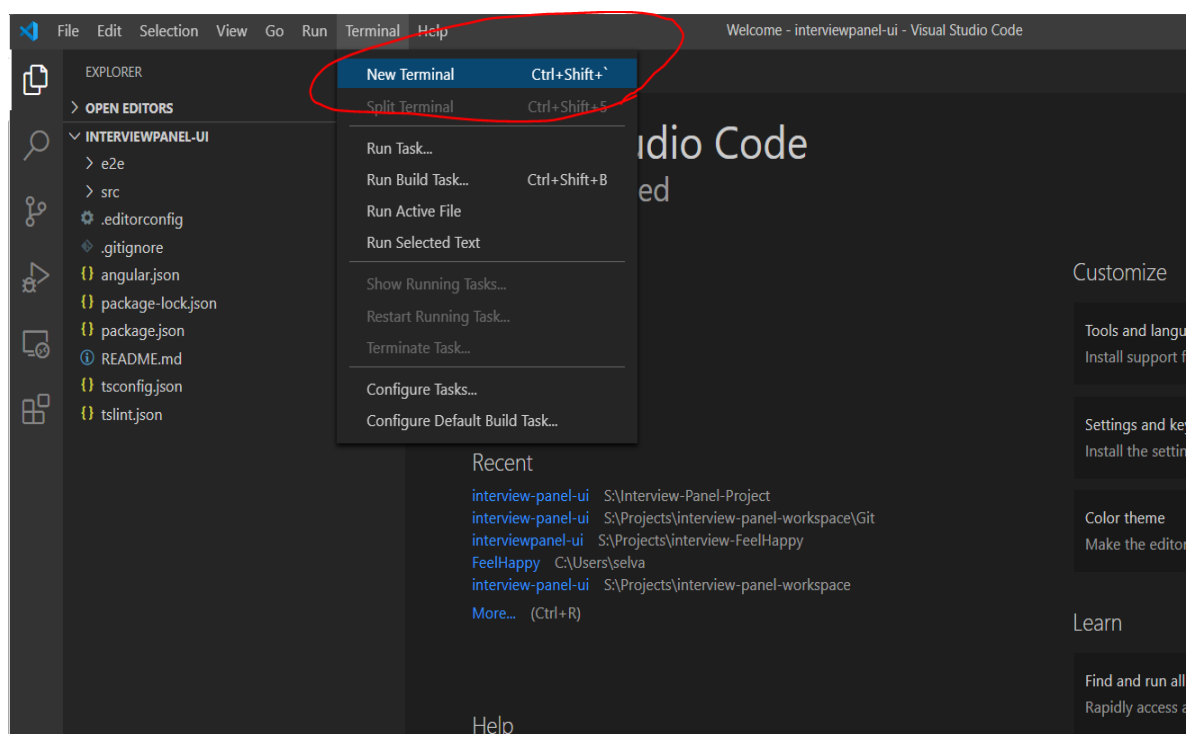
Step 3 > Open



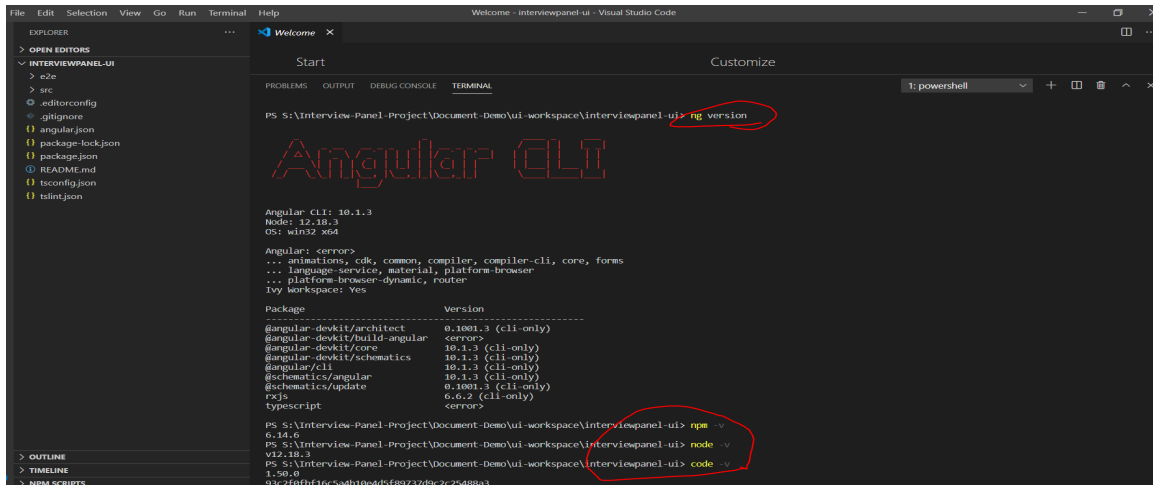
Step 3 > you can see the project structure



> Open new terminal



> Check all you have installed the version if not please follow the link [Angular 7.0 setup](#)



```
PS S:\Interview-Panel-Project\Document-Demo\ui-workspace\interviewpanel-ui> ng version

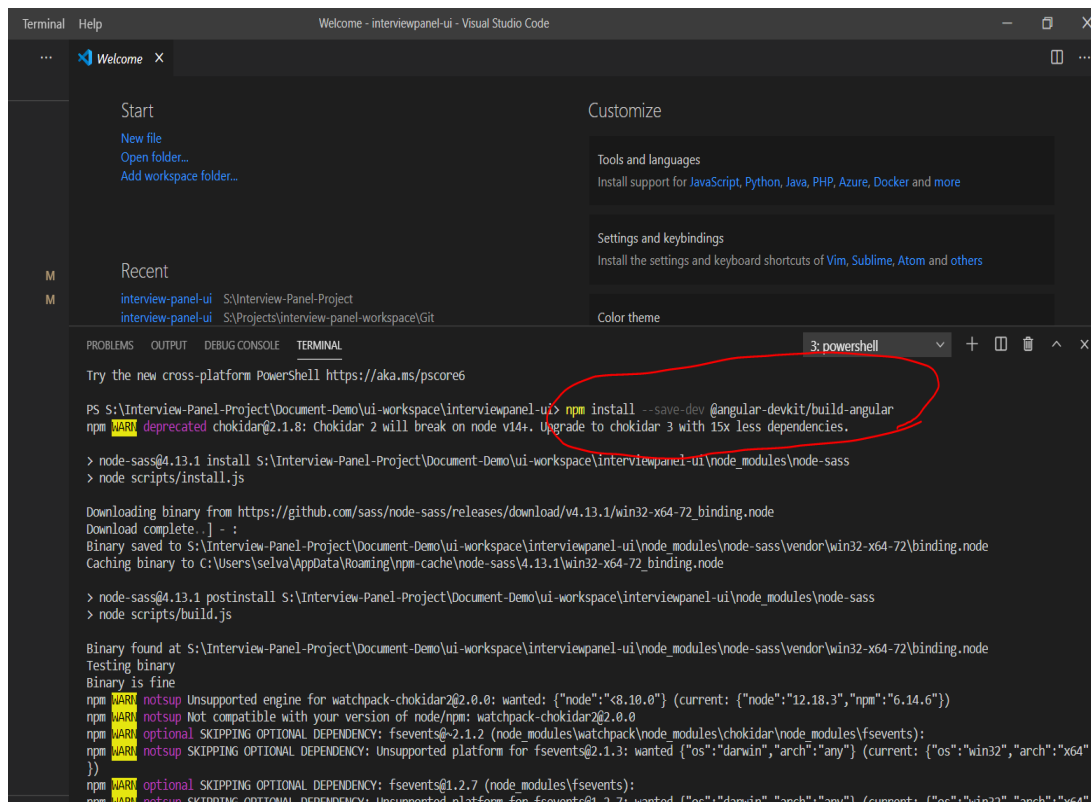
Angular CLI: 10.1.3
Node: 12.18.3
OS: win32 x64

Angular: <error>
... animations, cdk, common, compiler, compiler-cli, core, forms
... language-service, material, platform-browser
... platform-browser-dynamic, router
Ivy Workspace: Yes

Package                    Version
-----                    -
@angular-devkit/architect  0.1001.3 (cli-only)
@angular-devkit/build-angular <error>
@angular-devkit/core       10.1.3 (cli-only)
@angular-devkit/schematics 10.1.3 (cli-only)
@angular/cli               10.1.3 (cli-only)
@schematics/angular        10.1.3 (cli-only)
@schematics/update         0.1001.3 (cli-only)
rxjs                      6.6.2 (cli-only)
typescript                 <error>

PS S:\Interview-Panel-Project\Document-Demo\ui-workspace\interviewpanel-ui> npm -v
6.14.6
PS S:\Interview-Panel-Project\Document-Demo\ui-workspace\interviewpanel-ui> node -v
v12.18.3
PS S:\Interview-Panel-Project\Document-Demo\ui-workspace\interviewpanel-ui> code -v
1.50.0
03c240f0bf16c5ad810eadf5f0973709c2c25488a3
```

Must run this command: `npm install --save-dev @angular-devkit/build-angular`



```
PS S:\Interview-Panel-Project\Document-Demo\ui-workspace\interviewpanel-ui> npm install --save-dev @angular-devkit/build-angular
npm WARN deprecated chokidar@2.1.8: Chokidar 2 will break on node v14+. Upgrade to chokidar 3 with 15x less dependencies.

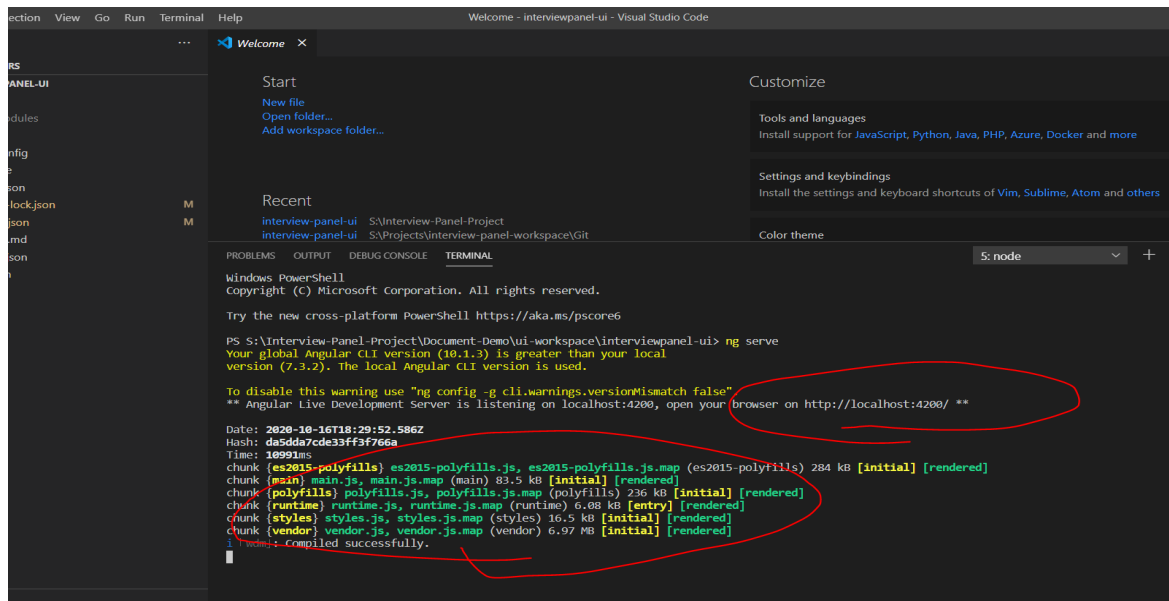
> node-sass@4.13.1 install S:\Interview-Panel-Project\Document-Demo\ui-workspace\interviewpanel-ui\node_modules\node-sass
> node scripts/install.js

Downloading binary from https://github.com/sass/node-sass/releases/download/v4.13.1/win32-x64-72_binding.node
Download complete.] - :
Binary saved to S:\Interview-Panel-Project\Document-Demo\ui-workspace\interviewpanel-ui\node_modules\node-sass\vendor\win32-x64-72\binding.node
Caching binary to C:\Users\selsva\AppData\Roaming\npm-cache\node-sass\4.13.1\win32-x64-72_binding.node

> node-sass@4.13.1 postinstall S:\Interview-Panel-Project\Document-Demo\ui-workspace\interviewpanel-ui\node_modules\node-sass
> node scripts/build.js

Binary found at S:\Interview-Panel-Project\Document-Demo\ui-workspace\interviewpanel-ui\node_modules\node-sass\vendor\win32-x64-72\binding.node
Testing binary
Binary is fine
npm WARN notsup Unsupported engine for watchpack-chokidar2@2.0.0: wanted {"node": "<8.10.0"} (current: {"node": "12.18.3", "npm": "6.14.6"})
npm WARN notsup Not compatible with your version of node/npm: watchpack-chokidar2@2.0.0
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@2.1.2 (node_modules\watchpack\node_modules\chokidar\node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@2.1.3: wanted {"os": "darwin", "arch": "any"} (current: {"os": "win32", "arch": "x64"})
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.7 (node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.7: wanted {"os": "darwin", "arch": "any"} (current: {"os": "win32", "arch": "x64"})
```

> Compiled and Run Successfully in 4200 port : <http://localhost:4200/>



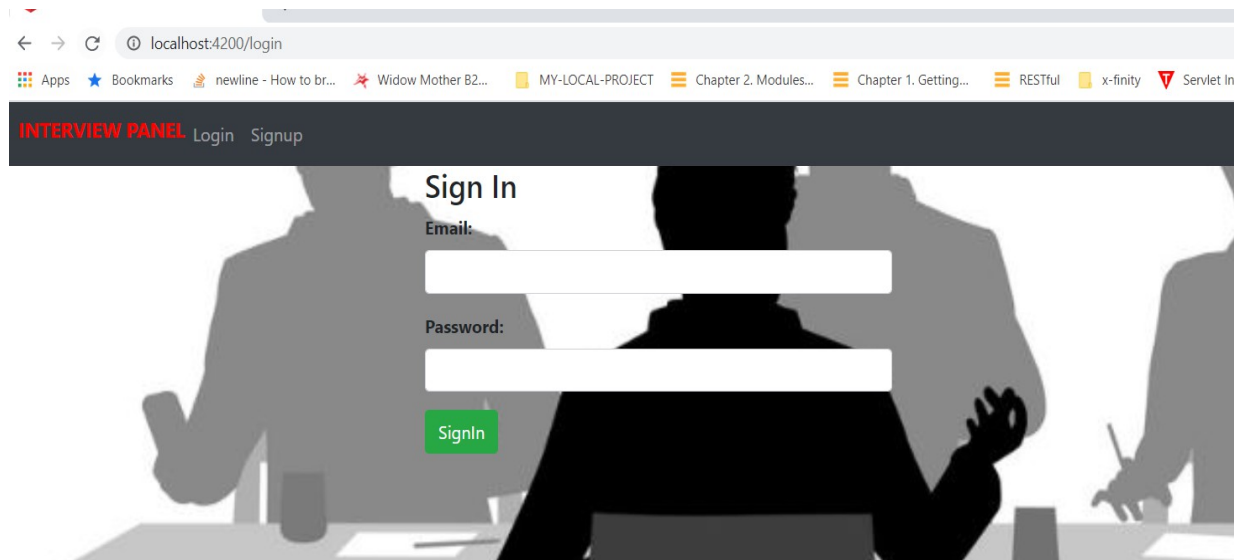
The screenshot shows the Visual Studio Code interface with the terminal window open. The terminal output indicates that the application has been compiled successfully and is now running on localhost:4200. The output includes the following text:

```
PS S:\Interview-Panel-Project\Document-Demo\ui-workspace\interviewpanel-ui> ng serve
Your global Angular CLI version (10.1.3) is greater than your local
version (7.3.2). The local Angular CLI version is used.

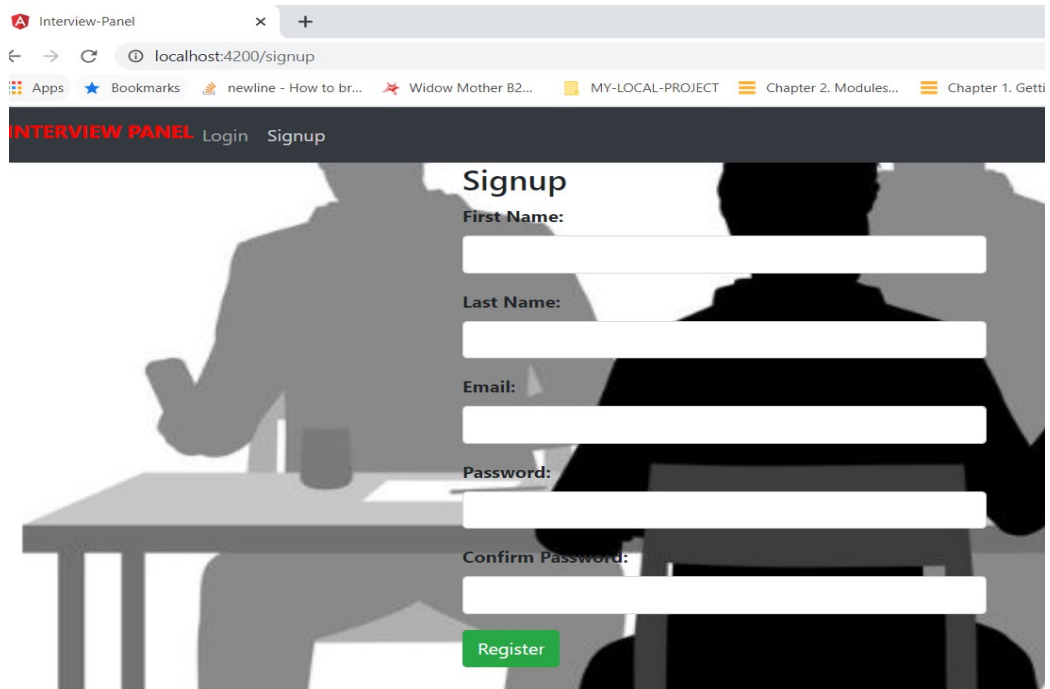
To disable this warning use "ng config -g cli.warnings.versionMismatch false"
** Angular Live Development Server is listening on localhost:4200, open your browser on http://localhost:4200/ **

Date: 2020-10-16T18:29:52.586Z
Hash: da5dda7cde33ff3f766a
Time: 10991ms
chunk (es2015-polyfills) es2015-polyfills.js, es2015-polyfills.js.map (es2015-polyfills) 284 kB [initial] [rendered]
chunk (main) main.js, main.js.map (main) 83.5 kB [initial] [rendered]
chunk (polyfills) polyfills.js, polyfills.js.map (polyfills) 236 kB [initial] [rendered]
chunk (runtime) runtime.js, runtime.js.map (runtime) 6.08 kB [entry] [rendered]
chunk (styles) styles.js, styles.js.map (styles) 16.5 kB [initial] [rendered]
chunk (vendor) vendor.js, vendor.js.map (vendor) 6.97 MB [initial] [rendered]
1 bundle compiled successfully.
```

Application run successfully:



Signup : with new user



A screenshot of a web browser showing the 'Signup' page of an application named 'Interview-Panel'. The browser's address bar shows 'localhost:4200/signup'. The page has a dark header with the application name and links for 'Login' and 'Signup'. The main content area features a background illustration of two people at a table. The 'Signup' form includes fields for 'First Name', 'Last Name', 'Email', 'Password', and 'Confirm Password', each with a white input box. A green 'Register' button is positioned at the bottom of the form.

Interview-Panel

localhost:4200/signup

Apps Bookmarks newline - How to br... Widow Mother B2... MY-LOCAL-PROJECT Chapter 2. Modules... Chapter 1. Getti

INTERVIEW PANEL Login Signup

Signup

First Name:

Last Name:

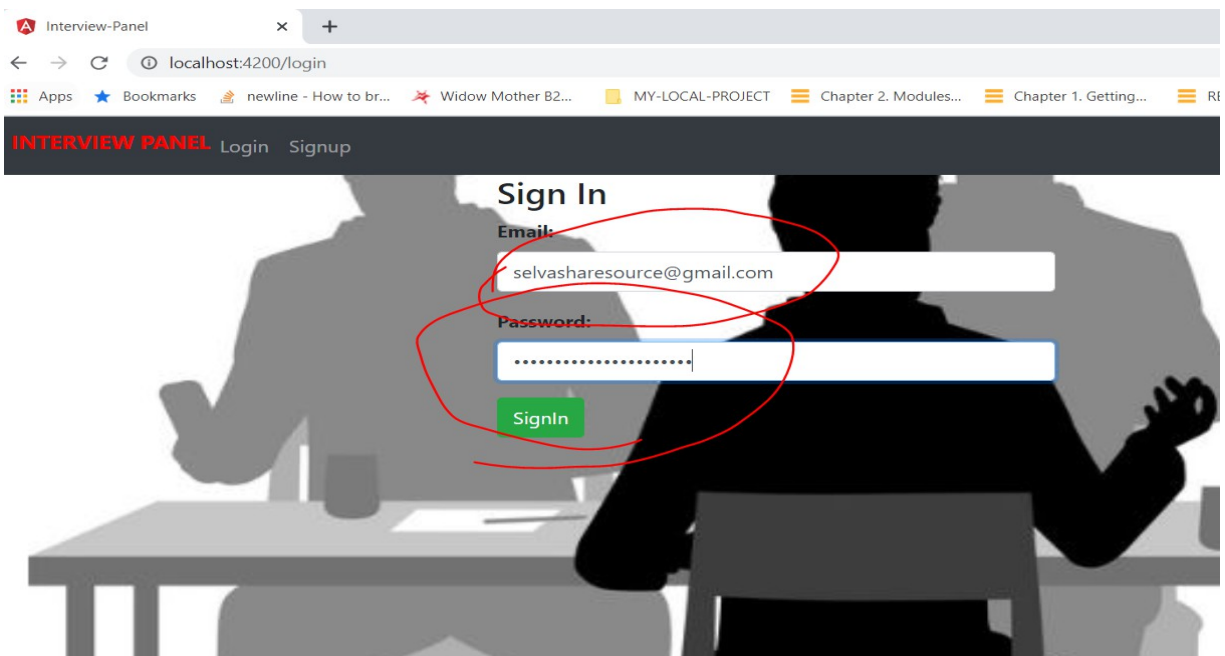
Email:

Password:

Confirm Password:

Register

SignIn:



A screenshot of a web browser showing the 'Sign In' page of the 'Interview-Panel' application. The browser's address bar shows 'localhost:4200/login'. The page has a dark header with the application name and links for 'Login' and 'Signup'. The main content area features a background illustration of two people at a table. The 'Sign In' form includes fields for 'Email' and 'Password', each with a white input box. A green 'SignIn' button is positioned at the bottom of the form. A red circle highlights the email and password fields and the 'SignIn' button.

Interview-Panel

localhost:4200/login

Apps Bookmarks newline - How to br... Widow Mother B2... MY-LOCAL-PROJECT Chapter 2. Modules... Chapter 1. Getting... RT

INTERVIEW PANEL Login Signup

Sign In

Email:

Password:

SignIn

Successfull page:

Interview-Panel x +

localhost:4200

Apps Bookmarks newline - How to br... Widow Mother B2... MY-LOCAL-PROJECT Chapter 2. Modules... Chapter 1. Getting... RESTful x-finity Servlet Interface - j... Other bookma

INTERVIEW PANEL View InterviewQuestionAnswers Add New InterviewQuestionAnswers Feedback Contact Us LogOut

Filter

Employer Name	Question And Answers			QuestionPosted By	Interview Date
Firstdata	Technology	Question Asked	Given Answer	Selvapandy Moorthy	10/10/2020
	Spring	What is Autowired?	Autowiring feature of spring framework enables you to inject the object dependency implicitly.		
	Spring	What is Qualifier?	The @Qualifier annotation is used to resolve the autowiring conflict, when there are multiple beans of same type.		
Apple	Angular	What is Typescript?	Superset of javascript	Paul	10/10/2020
	Technology	Question Asked	Given Answer		
	Node JS	What is NodeJS	NodeJS UI development		
Microsoft	Microservice	What is Mircoservice?	Microservice granular service	Selvapandy Moorthy	10/10/2020
	Technology	Question Asked	Given Answer		
	Java	java is	java		
	networking	what is ccna	ccna		

Items per page: 9 1 - 3 of 3

Add Interview Question and Answer Page:

localhost:4200/addinterviewqa

Apps Bookmarks newline - How to br... Widow Mother B2... MY-LOCAL-PROJECT Chapter 2. Modules... Chapter 1. Getting... RESTful x-finity Servlet Interface - j... Other bookma

INTERVIEW PANEL View InterviewQuestionAnswers Add New InterviewQuestionAnswers Feedback Contact Us LogOut

Add Interview Question And Answer

***Employer Name:**
Firstdata/Virtusa/Apple/Microsoft

***Your Name:**
Selvapandy Moorthy

Your contact:
Mobile/Email

***Question and Answers:** Add More

Technology :	Question Asked :	Given Answer:	Remove
Angular	What is typescript?	Superset of Javascript	

***Interview Date:**
10/10/2020

Create

