

ACTIVITY1

Requirements:

You are required to deliver a REST service, that provides an interface for submitting a set of GitHub repositories (identified as a list of strings of the form “username/repositoryname”). Your service should identify the set of GitHub developers who have contributed to any of these repositories in 2018, and for this user set, provide an ordering that ranks these users according to one of the following criteria:

1. Total number of commit contributions to any project to which a user has contributed.
2. Total number of commit contributions as above, but restricted to projects that are members of the original submitted set.
3. The number of known programming languages for each user (presuming that the languages of any repository committed to are known to the user)
4. The weekly commit rate of users (provide a weekly rank ordering) for the submitted project set, for 2018.
5. The average commit rate of each user to any project, for 2018.
6. The total number of collaborators in 2018 (ie. a count of other users who have contributed to any project that the user has contributed to).

Your rest service should email the results of calculation to the submitter once complete. You may design the REST API of your service as you see fit. You must describe it in your report. Your service must be capable of accepting concurrent requests.

You should provide a report detailing the architecture of your solution, and detailing results of the execution of your solution over a set of at least 5 repositories.

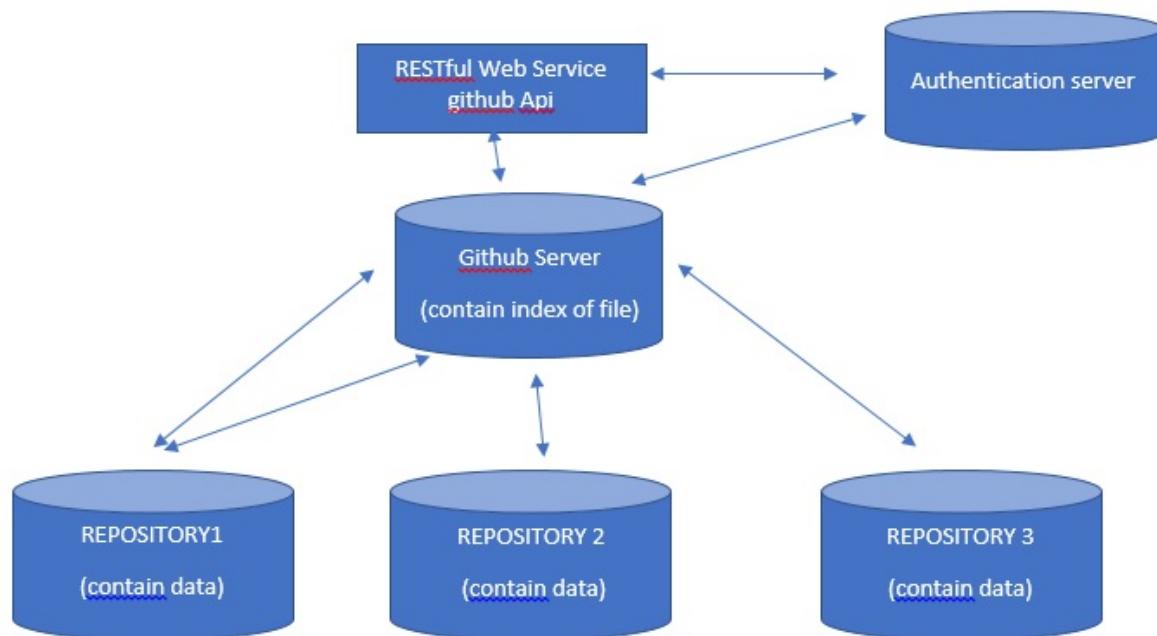
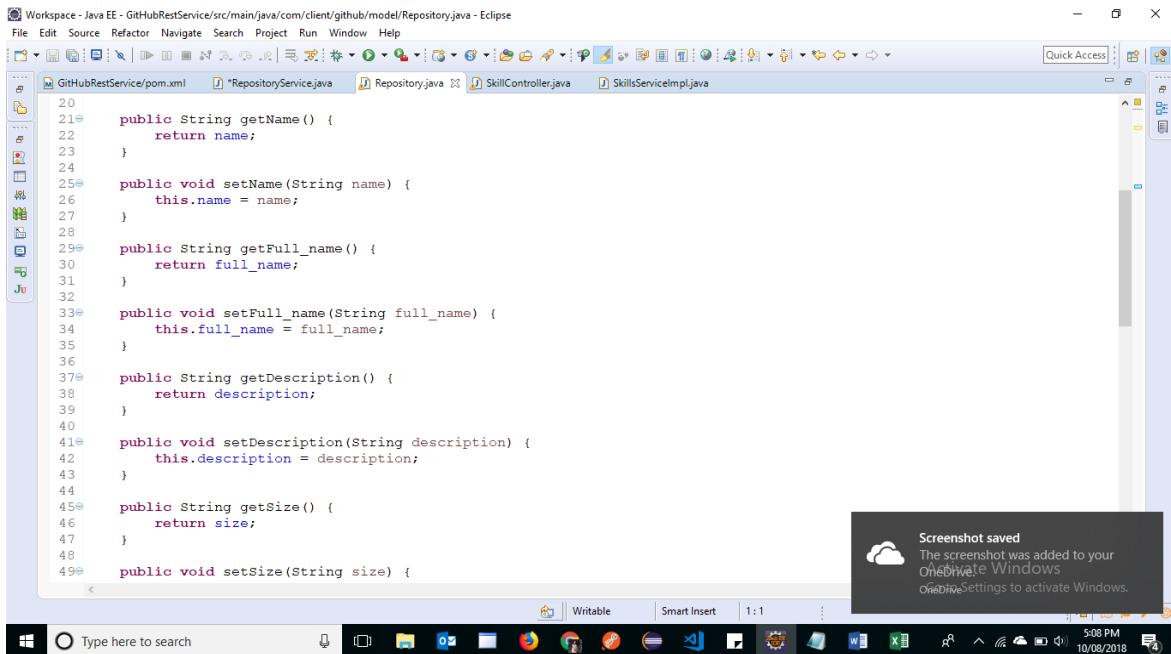


Fig 1. Block diagram for Rest Services

The report shows how code executes and how the required outputs are achieved. The screenshots attached below will give a clear picture of how the Rest services work keeping all the requirements in the mind and stepwise display of input and output for the same.

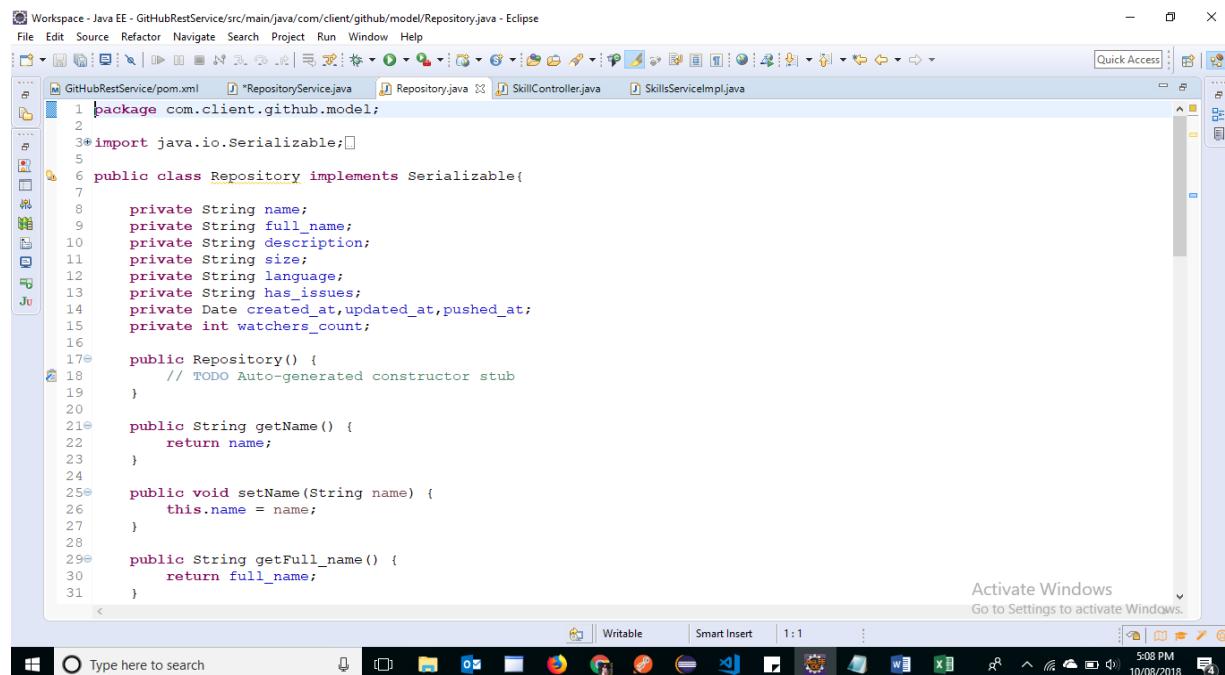
Step 1

Creating Model class to display the required values



The screenshot shows the Eclipse IDE interface with the 'Repository.java' file open in the editor. The code defines a simple model class with getters and setters for name, full_name, description, and size. A tooltip 'Screenshot saved' is visible in the bottom right corner.

```
20 public String getName() {
21     return name;
22 }
23
24 public void setName(String name) {
25     this.name = name;
26 }
27
28 public String getFull_name() {
29     return full_name;
30 }
31
32 public void setFull_name(String full_name) {
33     this.full_name = full_name;
34 }
35
36 public String getDescription() {
37     return description;
38 }
39
40 public void setDescription(String description) {
41     this.description = description;
42 }
43
44 public String getSize() {
45     return size;
46 }
47
48 public void setSize(String size) {
```



The screenshot shows the Eclipse IDE interface with the 'Repository.java' file open in the editor. The code has been updated to include additional private fields: language, has_issues, created_at, updated_at, pushed_at, and watchers_count. A tooltip 'Activate Windows' is visible in the bottom right corner.

```
1 package com.client.github.model;
2
3* import java.io.Serializable;□
4
5 public class Repository implements Serializable{
6
7     private String name;
8     private String full_name;
9     private String description;
10    private String size;
11    private String language;
12    private String has_issues;
13    private Date created_at,updated_at,pushed_at;
14    private int watchers_count;
15
16    public Repository() {
17        // TODO Auto-generated constructor stub
18    }
19
20    public String getName() {
21        return name;
22    }
23
24    public void setName(String name) {
25        this.name = name;
26    }
27
28    public String getFull_name() {
29        return full_name;
30    }
31}
```

Workspace - Java EE - GitHubRestService/src/main/java/com/client/github/model/Repository.java - Eclipse

```
38     }
39 }
40
41     public void setDescription(String description) {
42         this.description = description;
43     }
44
45     public String getSize() {
46         return size;
47     }
48
49     public void setSize(String size) {
50         this.size = size;
51     }
52
53     public String getLanguage() {
54         return language;
55     }
56
57     public void setLanguage(String language) {
58         this.language = language;
59     }
60
61     public String getHas_issues() {
62         return has_issues;
63     }
64
65     public void setHas_issues(String has_issues) {
66         this.has_issues = has_issues;
67     }
```

Activate Windows
Go to Settings to activate Windows.

Workspace - Java EE - GitHubRestService/src/main/java/com/client/github/model/Repository.java - Eclipse

```
62     return has_issues;
63 }
64
65     public void setHas_issues(String has_issues) {
66         this.has_issues = has_issues;
67     }
68
69     public Date getCreated_at() {
70         return created_at;
71     }
72
73     public void setCreated_at(Date created_at) {
74         this.created_at = created_at;
75     }
76
77     public Date getUpdated_at() {
78         return updated_at;
79     }
80
81     public void setUpdated_at(Date updated_at) {
82         this.updated_at = updated_at;
83     }
84
85     public Date getPushed_at() {
86         return pushed_at;
87     }
88
89     public void setPushed_at(Date pushed_at) {
90         this.pushed_at = pushed_at;
91     }
```

Screenshot saved
The screenshot was added to your OneDrive Windows
OneDrive Settings to activate Windows.

The screenshot shows the Eclipse IDE interface with the title "Workspace - Java EE - GitHubRestService/src/main/java/com/client/github/model/Repository.java - Eclipse". The code editor displays the following Java code:

```
74     this.created_at = created_at;
75   }
76 
77   public Date getUpdated_at() {
78     return updated_at;
79   }
80 
81   public void setUpdated_at(Date updated_at) {
82     this.updated_at = updated_at;
83   }
84 
85   public Date getPushed_at() {
86     return pushed_at;
87   }
88 
89   public void setPushed_at(Date pushed_at) {
90     this.pushed_at = pushed_at;
91   }
92 
93   public int getWatchers_count() {
94     return watchers_count;
95   }
96 
97   public void setWatchers_count(int watchers_count) {
98     this.watchers_count = watchers_count;
99   }
100 
101 
102 }
```

A tooltip message "Screenshot saved" is visible in the bottom right corner of the screen.

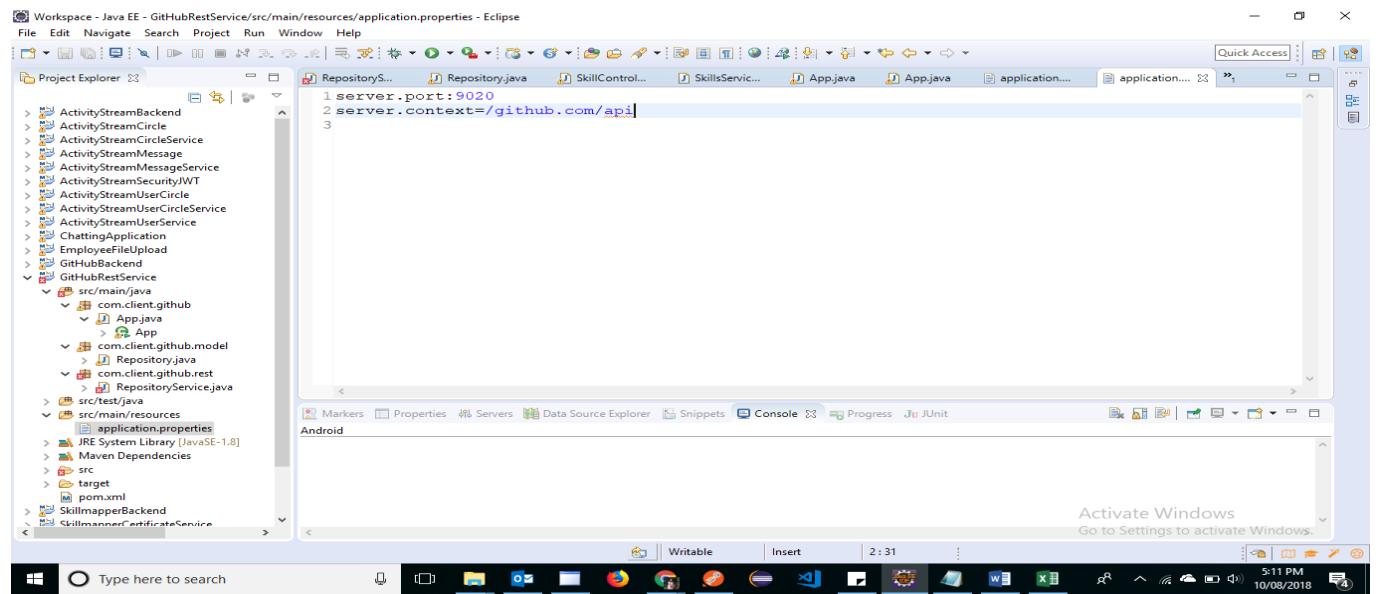
Step 2: Creating REST API

The screenshot shows the Eclipse IDE interface with the title "Workspace - Java EE - GitHubRestService/src/main/java/com/client/github/App.java - Eclipse". The code editor displays the following Java code:

```
14 /**
15  * Hello world!
16  */
17 
18 @SpringBootApplication(scanBasePackages={"com.stackroute"})
19 @EntityScan(basePackageClasses=Repository.class)
20 public class App
21 {
22   public static void main( String[] args )
23   {
24     SpringApplication.run(App.class, args);
25   }
26 
27   @Bean
28   public RestTemplate restTemplate() {
29     return new RestTemplate();
30   }
31 }
```

A tooltip message "Activate Windows Go to Settings to activate Windows" is visible in the bottom right corner of the screen.

Step 3: Setting up server id and port no



Step 3: Verifying the output via Postman

The screenshot shows the Postman application interface. A GET request is made to <https://api.github.com/users/rakesh13/repos>. The response body is displayed in JSON format, showing details about the user's repositories. The status is 200 OK and the time taken is 1725 ms.

```
5
6   "name": "ActivityStream",
7   "full_name": "rakesh13/ActivityStream",
8   "owner": {
9     "login": "rakesh13",
10    "id": 4477805,
11    "node_id": "MDQ6VXNlcjQ0Nzc4ODUu",
12    "avatar_url": "https://avatars0.githubusercontent.com/u/4477805?v=4",
13    "gravatar_id": "",
14    "url": "https://api.github.com/users/rakesh13",
15    "html_url": "https://github.com/rakesh13",
16    "followers_url": "https://api.github.com/users/rakesh13/followers",
17    "following_url": "https://api.github.com/users/rakesh13/following{/other_user}",
18    "gists_url": "https://api.github.com/users/rakesh13/gists{/gist_id}",
19    "starred_url": "https://api.github.com/users/rakesh13/starred{/owner}{/repo}",
20    "subscriptions_url": "https://api.github.com/users/rakesh13/subscriptions",
21    "organization_url": "https://api.github.com/users/rakesh13/orgs",
22    "repos_url": "https://api.github.com/users/rakesh13/repos",
23    "events_url": "https://api.github.com/users/rakesh13/events{/privacy}",
24    "received_events_url": "https://api.github.com/users/rakesh13/received_events",
25    "type": "User",
26    "site_admin": false
},
```

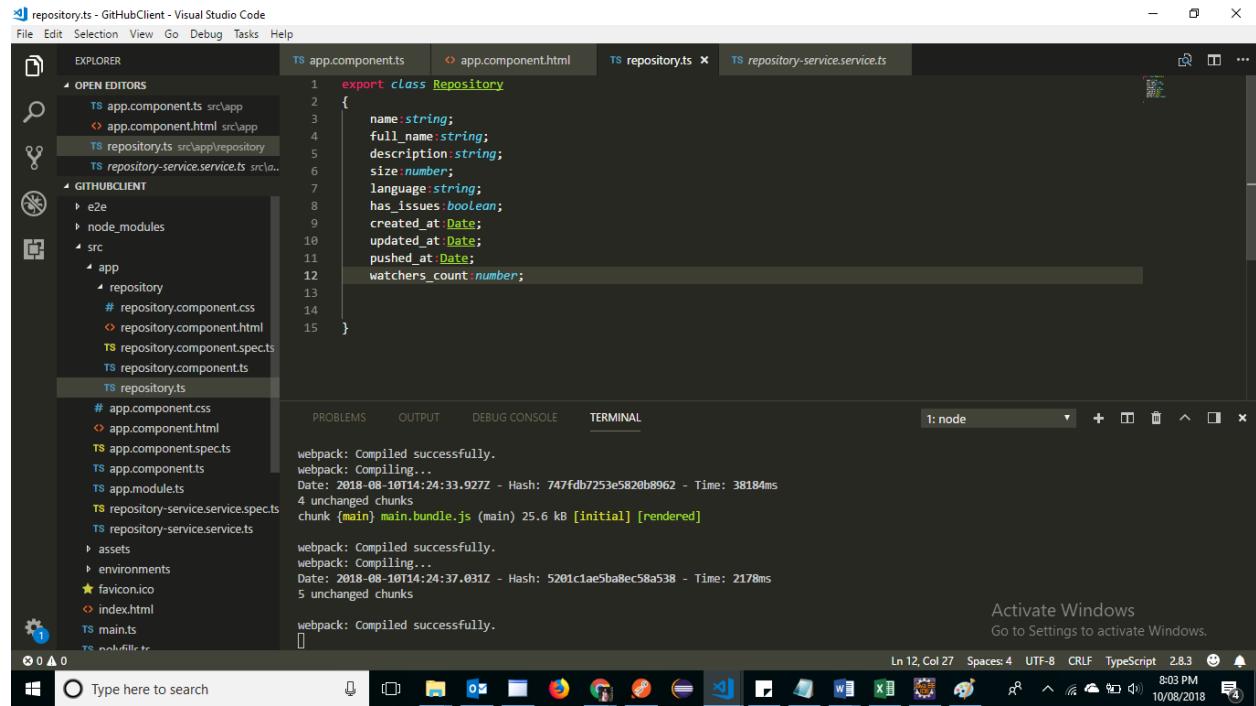
The 'History' tab on the left shows previous API calls made from Postman. The status bar at the bottom right indicates 'Activate Windows Go to Settings to activate Windows.'

Step 5: Creating the UI using Angular 6 and Node server

The screenshot shows the Visual Studio Code interface with the following details:

- Title Bar:** app.component.html - GitHubClient - Visual Studio Code
- File Menu:** File Edit Selection View Go Debug Tasks Help
- Explorer:** Shows the project structure under 'OPEN EDITORS' and 'GITHUBCLIENT'. The 'src/app/app.component.html' file is selected.
- Code Editor:** Displays the content of 'app.component.html'. The code includes placeholder text, a central heading, and a list of links to help start the application.
- Terminal:** Shows the output of the build process, indicating that the live development server is running on port 4200 and provides details about the build time, hash, and chunks.
- Bottom Status Bar:** Shows the current file (app.component.html), line count (15), column count (Col 8), spaces (Spaces: 2), encoding (UTF-8), line feed (LF), and HTML mode. It also shows the date and time (10/08/2018, 7:53 PM).

Step 6: Creating the model class in Angular



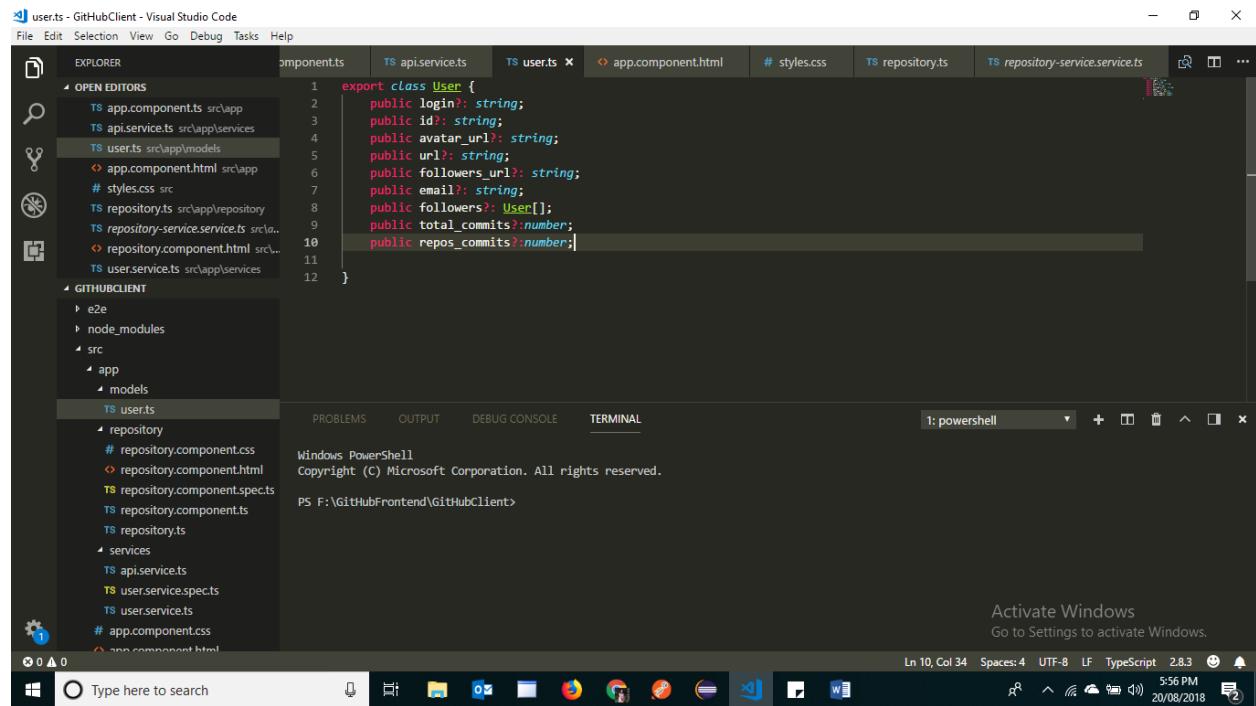
The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows the project structure under the `GITHUBCLIENT` folder, including `e2e`, `node_modules`, and `src` (which contains `app`, `assets`, `environments`, `favicon.ico`, `index.html`, `main.ts`, and `polyfills.ts`).
- Editor:** The `repository.ts` file is open in the editor tab, showing the following code:export class Repository {
 name:string;
 full_name:string;
 description:string;
 size:number;
 language:string;
 has_issues:boolean;
 created_at:Date;
 updated_at:Date;
 pushed_at:Date;
 watchers_count:number;
}

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
- Terminal:** The terminal shows the output of a webpack build process:webpack: Compiled successfully.
webpack: Compiling...
Date: 2018-08-10T14:24:33.927Z - Hash: 747fdb7253e5820b8962 - Time: 38184ms
4 unchanged chunks
chunk {main} main.bundle.js (main) 25.6 kB [initial] [rendered]

webpack: Compiled successfully.
webpack: Compiling...
Date: 2018-08-10T14:24:37.031Z - Hash: 5201c1ae5ba8ec58a538 - Time: 2178ms
5 unchanged chunks

webpack: Compiled successfully.
- Bottom Bar:** Shows the Windows taskbar with various pinned icons like File Explorer, Edge, and File Explorer again.



The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows the project structure under the `GITHUBCLIENT` folder, including `e2e`, `node_modules`, and `src` (which contains `app`, `models`, `repository`, `services`, and `styles.css`).
- Editor:** The `user.ts` file is open in the editor tab, showing the following code:export class User {
 public login?: string;
 public id?: string;
 public avatar_url?: string;
 public url?: string;
 public followers_url?: string;
 public email?: string;
 public followers?: User[];
 public total_commits?: number;
 public repos_commits?: number[];
}

1 2 3 4 5 6 7 8 9 10 11 12
- Terminal:** The terminal shows the output of a PowerShell session:Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
PS F:\GitHubFrontend\GitHubClient>
- Bottom Bar:** Shows the Windows taskbar with various pinned icons like File Explorer, Edge, and File Explorer again.

Step 7: Creating user.service.ts file

```

1 import {Injectable} from '@angular/core';
2 import {Observable} from "rxjs";
3 import {User} from "../models/user";
4 import {ApiService} from "./api.service";
5
6 @Injectable()
7 export class UserService {
8
9   constructor(private api: ApiService) {
10 }
11
12
13   getUsers(filter?: string): Observable<User[]> {
14
15     let endPoint = '/users';
16
17     return this.api.get(endPoint).map(res => res.json() as User[]).catch(err => Observable.throw(err));
18   }
19
20   search(q: string): Observable<any> {
21
22     let endPoint = '/search/users?q=' + q;
23
24     return this.api.get(endPoint).map(res => res.json()).catch(err => Observable.throw(err));
25   }
26
27   getUserFollowers(user: string): Observable<any> {
28
29     let endPoint = '/users/' + user + '/followers';
30
31     return this.api.get(endPoint).map(res => res.json()).catch(err => Observable.throw(err));
32   }
33
34   getUserCommits(user: string): Observable<any> {
35
36     let endPoint = '/users/' + user + '/repos/commits';
37
38     return this.api.get(endPoint).map(res => res.json()).catch(err => Observable.throw(err));
39   }
40
41   getUserCommits2018(user: string): Observable<any> {
42
43     let endPoint = '/users/' + user + '/repos/commits/(year)';
44
45     return this.api.get(endPoint).map(res => res.json()).catch(err => Observable.throw(err));
46   }
47
48 }

```

File Edit Selection View Go Debug Tasks Help

EXPLORER app.component.html # styles.css TS repository.ts TS repository-service.service.ts repository.component.html TS user.service.ts

GITHUBCLIENT

- # repository.component.css
- repository.component.html
- repository.component.specs.ts
- repository.component.ts
- services
- api.service.ts
- user.service.specs.ts
- user.service.ts**
- # app.component.css
- app.component.html
- app.component.specs.ts
- app.components.ts
- app.module.ts
- index.ts
- repository-service.service.specs.ts
- repository-service.service.ts

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

t: powershell

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
PS F:\GitHubFrontend\GitHubClient>

Ln 41, Col 65 Spaces: 4 UTF-8 LF TypeScript 2.8.3 5:58 PM 20/08/2018

```

1 import {Injectable} from '@angular/core';
2 import {Observable} from "rxjs";
3 import {User} from "../models/user";
4 import {ApiService} from "./api.service";
5
6 @Injectable()
7 export class UserService {
8
9   constructor(private api: ApiService) {
10 }
11
12
13   getUsers(filter?: string): Observable<User[]> {
14
15     let endPoint = '/users';
16
17     return this.api.get(endPoint).map(res => res.json() as User[]).catch(err => Observable.throw(err));
18   }
19
20   search(q: string): Observable<any> {
21
22     let endPoint = '/search/users?q=' + q;
23
24     return this.api.get(endPoint).map(res => res.json()).catch(err => Observable.throw(err));
25   }
26
27   getUserFollowers(user: string): Observable<any> {
28
29     let endPoint = '/users/' + user + '/followers';
30
31     return this.api.get(endPoint).map(res => res.json()).catch(err => Observable.throw(err));
32   }
33
34   getUserCommits(user: string): Observable<any> {
35
36     let endPoint = '/users/' + user + '/repos/commits';
37
38     return this.api.get(endPoint).map(res => res.json()).catch(err => Observable.throw(err));
39   }
40
41   getUserCommits2018(user: string): Observable<any> {
42
43     let endPoint = '/users/' + user + '/repos/commits/(year)';
44
45     return this.api.get(endPoint).map(res => res.json()).catch(err => Observable.throw(err));
46   }
47
48 }

```

File Edit Selection View Go Debug Tasks Help

EXPLORER app.component.html # styles.css TS repository.ts TS repository-service.service.ts repository.component.html TS user.service.ts

GITHUBCLIENT

- # repository.component.css
- repository.component.html
- repository.component.specs.ts
- repository.component.ts
- services
- api.service.ts
- user.service.specs.ts
- user.service.ts**
- # app.component.css
- app.component.html
- app.component.specs.ts
- app.components.ts
- app.module.ts
- index.ts
- repository-service.service.specs.ts
- repository-service.service.ts

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

t: powershell

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
PS F:\GitHubFrontend\GitHubClient>

Ln 41, Col 65 Spaces: 4 UTF-8 LF TypeScript 2.8.3 5:58 PM 20/08/2018

Step 8: Creating api.service.ts to communicate with github api and then the result is directed to user.service.ts

```

1 import {Injectable} from '@angular/core';
2 import {Headers, Http, Response, RequestOptionsArgs} from '@angular/http';
3 import {Observable} from 'rxjs';
4
5
6 @Injectable()
7 export class ApiService {
8
9
10    private headers: Headers = new Headers();
11    private requestOptions: RequestOptionsArgs = {};
12    private apiServer: string = "https://api.github.com";
13
14    constructor(private http: Http) {
15
16        this.headers.set("Content-Type", "application/json");
17
18        this.requestOptions.headers = this.headers;
19
20    }
21
22
23
24
25    getHeaders(): Headers {
26
27        return this.headers;
28
29    }
30
31    setHeader(key: string, value: string) {
32
33        this.headers.set(key, value);
34
35    }
36
37    removeHeader(key: string) {
38
39        this.headers.delete(key);
40
41    }
42
43    get(endPoint: string, options?: RequestOptionsArgs): Observable<Response> {
44        return this.http.get(this.createUrl(endPoint), this.getRequestOptions(options));
45
46    }
47
48    post(endPoint: string, body: any, options?: RequestOptionsArgs): Observable<Response> {
49
50        return this.http.post(this.createUrl(endPoint), body, this.getRequestOptions(options));
51
52    }
53
54
55}

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
PS F:\GitHubFrontend\GitHubClient>

Ln 100, Col 6 Spaces: 4 UTF-8 LF TypeScript 2.8.3 6:00 PM 20/08/2018

```

24
25    getHeaders(): Headers {
26
27        return this.headers;
28
29    }
30
31    setHeader(key: string, value: string) {
32
33        this.headers.set(key, value);
34
35    }
36
37    removeHeader(key: string) {
38
39        this.headers.delete(key);
40
41    }
42
43    get(endPoint: string, options?: RequestOptionsArgs): Observable<Response> {
44        return this.http.get(this.createUrl(endPoint), this.getRequestOptions(options));
45
46    }
47
48    post(endPoint: string, body: any, options?: RequestOptionsArgs): Observable<Response> {
49
50        return this.http.post(this.createUrl(endPoint), body, this.getRequestOptions(options));
51
52    }
53
54
55}

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
PS F:\GitHubFrontend\GitHubClient>

Ln 100, Col 6 Spaces: 4 UTF-8 LF TypeScript 2.8.3 6:00 PM 20/08/2018

api.service.ts - GitHubClient - Visual Studio Code

File Edit Selection View Go Debug Tasks Help

EXPLORER

OPEN EDITORS

- TS app.component.ts src\app
- TS api.service.ts src\app\services
- TS user.ts src\app\models
- ▷ app.component.html src\app
- # styles.css src
- TS repository.ts src\app\repository
- TS repository-service.service.ts src\app\services
- ▷ repository.component.html src\app
- TS user.service.ts src\app\services

GITHUBCLIENT

- # repository.component.css
- ▷ repository.component.html
- TS repository.component.spec.ts
- TS repository.component.ts
- services
- TS api.service.ts
- TS user.service.specs
- TS user.service.ts
- # app.component.css
- ▷ app.component.html
- TS app.component.spec.ts
- TS app.component.ts
- TS app.module.ts
- TS index.ts
- TS repository-service.service.spec.ts
- TS repository-service.service.ts

api.service.ts

```
45
46
47 post(endPoint: string, body: any, options?: RequestOptionsArgs): Observable<Response> {
48
49     return this.http.post(this.createUrl(endPoint), body, this.getRequestOptions(options));
50 }
51
52 put(endPoint: string, body: any, options?: RequestOptionsArgs): Observable<Response> {
53
54     return this.http.put(this.createUrl(endPoint), body, this.getRequestOptions(options));
55 }
56
57 delete(endPoint: string, options?: RequestOptionsArgs): Observable<Response> {
58
59     return this.http.delete(this.createUrl(endPoint), this.getRequestOptions(options));
60 }
61
62 patch(endPoint: string, body: any, options?: RequestOptionsArgs): Observable<Response> {
63
64     return this.http.patch(this.createUrl(endPoint), body, this.getRequestOptions(options));
65 }
66
67 head(endPoint: string, options?: RequestOptionsArgs): Observable<Response> {
68
69 }
70
71 
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS F:\GitHubFrontend\GitHubClient>

1: powershell

Activate Windows
Go to Settings to activate Windows.

Ln 100, Col 6 Spaces: 4 UTF-8 LF TypeScript 2.8.3 6:01 PM 20/08/2018

Type here to search

api.service.ts - GitHubClient - Visual Studio Code

File Edit Selection View Go Debug Tasks Help

EXPLORER

OPEN EDITORS

- TS app.component.ts src\app
- TS api.service.ts src\app\services
- TS user.ts src\app\models
- ▷ app.component.html src\app
- # styles.css src
- TS repository.ts src\app\repository
- TS repository-service.service.ts src\app\services
- ▷ repository.component.html src\app
- TS user.service.ts src\app\services

GITHUBCLIENT

- # repository.component.css
- ▷ repository.component.html
- TS repository.component.spec.ts
- TS repository.component.ts
- TS repository.ts
- services
- TS api.service.ts
- TS user.service.specs
- TS user.service.ts
- # app.component.css
- ▷ app.component.html
- TS app.component.spec.ts
- TS app.component.ts
- TS app.module.ts
- TS index.ts
- TS repository-service.service.spec.ts
- TS repository-service.service.ts

api.service.ts

```
68
69
70 head(endPoint: string, options?: RequestOptionsArgs): Observable<Response> {
71
72     return this.http.head(this.createUrl(endPoint), this.getRequestOptions(options));
73 }
74
75 options(endPoint: string, options?: RequestOptionsArgs): Observable<Response> {
76
77     return this.http.post(this.createUrl(endPoint), this.getRequestOptions(options));
78 }
79
80createUrl(endPoint: string) {
81
82     let url = this.apiServer + endPoint;
83     if (!endPoint.startsWith('/')) {
84         url = this.apiServer + '/' + endPoint;
85     }
86
87     return url;
88 }
89
90 get RequestOptions(options?: RequestOptionsArgs): RequestOptionsArgs {
91
92     this.requestOptions.headers = this.headers;
93
94     if (options) {
95         this.requestOptions.headers = this.headers;
96     }
97 }
98
99 
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS F:\GitHubFrontend\GitHubClient>

1: powershell

Activate Windows
Go to Settings to activate Windows.

Ln 100, Col 6 Spaces: 4 UTF-8 LF TypeScript 2.8.3 6:01 PM 20/08/2018

Type here to search

api.service.ts - GitHubClient - Visual Studio Code

```

EXPLORER          TS api.service.ts ×   TS user.ts      app.component.html    # styles.css   TS repository.ts   TS repository-service.service.ts   repository.c
OPEN EDITORS
  TS app.components.ts src\app
  TS api.service.ts src\app\services
  TS user.ts src\app\models
  app.component.html src\app
  # styles.css src
  TS repository.ts src\app\repository
  TS repository-service.service.ts src\...
  repository.component.html src\...
  TS user.service.ts src\app\services
GITHUBCLIENT
  # repository.component.css
  repository.component.html
  TS repository.component.specs
  TS repository.component.ts
  TS repository.ts
  services
    TS api.service.ts
    TS user.service.specs
    TS user.service.ts
  # app.component.css
  app.component.html
  TS app.component.specs
  TS app.component.ts
  TS app.module.ts
  TS index.ts
  TS repository-service.service.specs
  TS repository-service.service.ts
  TS repository-service.service.ts
  Type here to search
  PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
PS F:\GitHubFrontend\GitHubClient>
  1: powershell
  Activate Windows
  Go to Settings to activate Windows.
Ln 100, Col 6  Spaces:4  UTF-8  LF  TypeScript  2.8.3  6:03 PM  20/08/2018

```

Step 8: app.module.ts for registering userservice and api service class

app.module.ts - GitHubClient - Visual Studio Code

```

EXPLORER          TS api.service.ts   TS app.module.ts ×   TS user.ts      app.component.html    # styles.css   TS repository.ts   TS repository-service.service.ts   repository.c
OPEN EDITORS
  TS app.components.ts src\app
  TS api.service.ts src\app\services
  TS app.module.ts src\app
  TS user.ts src\app\models
  app.component.html src\app
  # styles.css src
  TS repository.ts src\app\repository
  repository.component.html src\...
  TS user.service.ts src\app\services
GITHUBCLIENT
  TS repository.component.specs
  TS repository.component.ts
  TS repository.ts
  services
    TS api.service.ts
    TS user.service.specs
    TS user.service.ts
  # app.component.css
  app.component.html
  TS app.component.specs
  TS app.component.ts
  TS app.module.ts
  TS index.ts
  TS repository-service.service.specs
  TS repository-service.service.ts
  TS repository-service.service.ts
  Type here to search
  PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
PS F:\GitHubFrontend\GitHubClient>
  1: powershell
  Activate Windows
  Go to Settings to activate Windows.
Ln 1, Col 1  Spaces:2  UTF-8  LF  TypeScript  2.8.3  6:03 PM  20/08/2018

```

Step 9 : App.component.ts for fetching value from service and storing and then it will be displayed to user by app.componet.html

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer (Left):** Shows the project structure under "OPEN EDITORS".
 - GITHUBCLIENT:** Contains files like repository.component.ts, repository.ts, services, api.service.ts, user.service.ts, app.component.ts, and index.ts.
 - app:** Contains files like app.module.ts, user.ts, component.html, styles.css, repository.ts, repository.component.html, user.service.ts, and component.spec.ts.
- Code Editor (Center):** Displays the content of `app.component.ts`. The code implements the `OnInit` interface and uses dependency injection for `UserService`.
- Bottom Bar:** Includes tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, and TERMINAL. The TERMINAL tab shows a powershell window with the command `PS F:\GitHubFrontend\GitHubClient>`.
- Bottom Status Bar:** Shows file information (Line 9, Column 26), file encoding (UTF-8), and other settings.
- Bottom Icons:** Standard Windows taskbar icons for File, Copy, Paste, etc.

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows the project structure under "OPEN EDITORS" and "GITHUBCLIENT".
- Code Editor:** The main editor window displays `app.component.ts` with TypeScript code for a user search component.
- Bottom Bar:** Includes tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, and TERMINAL, along with a terminal window showing "powershell".
- Status Bar:** Displays file path "F:\GitHubFrontend\GitHubClient", line "1: powershell", and other system information.
- System Taskbar:** Shows icons for File Explorer, Task View, Start, and other system functions.

```
app.component.ts - GitHubClient - Visual Studio Code
File Edit Selection View Go Debug Tasks Help

EXPLORER
OPEN EDITORS
TS app.components.ts src\app
TS api.service.ts src\app\services
TS app.module.ts src\app
TS user.ts src\app\models
TS app.component.html src\app
# styles.css src
TS repository.ts src\app\repository
TS repository.component.html src\app
TS user.service.ts src\app\services

GITHUBCLIENT
TS repository.component.spec.ts
TS repository.component.ts
TS repository.ts
services
TS api.service.ts
TS user.service.ts
TS user.service.spec.ts
TS user.service.ts
# app.component.css
TS app.component.html
TS app.component.spec.ts
TS app.components.ts
TS app.module.ts
TS index.ts
TS repository-service.service.spec.ts
TS repository-service.service.ts

assets
environments
favicon.ico

app.component.ts
1: powershell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS F:\GitHubFrontend\GitHubClient>
Line 1, Col 1
In 9, Col 26 Spaces: 4 UTF-8 LF TypeScript 2.8.3
6:04 PM 20/08/2018
```

app.component.ts - GitHubClient - Visual Studio Code

File Edit Selection View Go Debug Tasks Help

EXPLORER

- OPEN EDITORS
- TS app.component.ts src\app
- TS api.service.ts src\app\services
- TS app.module.ts src\app
- TS user.ts src\app\models
- ▷ app.component.html src\app
- # styles.css src
- TS repository.ts src\app\repository
- ▷ repository.component.html src\...
- TS user.service.ts src\app\services

GITHUBCLIENT

- TS repository.component.spec.ts
- TS repository.component.ts
- TS repository.ts
- services
- TS api.service.ts
- TS user.service.spec.ts
- TS user.service.ts
- # app.component.css
- ▷ app.component.html
- TS app.component.spec.ts
- TS app.component.ts
- TS app.module.ts
- TS index.ts
- TS repository-service.service.spec.ts
- TS repository-service.service.ts

assets environments favicon.ico

Type here to search

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
PS F:\GitHubFrontend\GitHubClient>

1: powershell

Ln 9, Col 26 Spaces:4 UTF-8 LF TypeScript 2.8.3 6:05 PM 20/08/2018

app.component.ts - GitHubClient - Visual Studio Code

File Edit Selection View Go Debug Tasks Help

EXPLORER

- OPEN EDITORS
- TS app.component.ts src\app
- TS api.service.ts src\app\services
- TS app.module.ts src\app
- TS user.ts src\app\models
- ▷ app.component.html src\app
- # styles.css src
- TS repository.ts src\app\repository
- ▷ repository.component.html src\...
- TS user.service.ts src\app\services

GITHUBCLIENT

- TS repository.component.spec.ts
- TS repository.component.ts
- TS repository.ts
- services
- TS api.service.ts
- TS user.service.spec.ts
- TS user.service.ts
- # app.component.css
- ▷ app.component.html
- TS app.component.spec.ts
- TS app.component.ts
- TS app.module.ts
- TS index.ts
- TS repository-service.service.spec.ts
- TS repository-service.service.ts

assets environments favicon.ico

Type here to search

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
PS F:\GitHubFrontend\GitHubClient>

1: powershell

Ln 9, Col 26 Spaces:4 UTF-8 LF TypeScript 2.8.3 6:05 PM 20/08/2018

app.component.ts - GitHubClient - Visual Studio Code

File Edit Selection View Go Debug Tasks Help

EXPLORER TS app.component.ts x TS api.service.ts TS app.module.ts TS user.ts app.component.html # styles.css TS repository.ts

```
OPEN EDITORS
TS app.component.ts src\app
TS api.service.ts src\app\services
TS app.module.ts src\app
TS user.ts src\app\models
app.component.html src\app
# styles.css src
TS repository.ts src\app\repository
repository.component.html src\app
TS user.service.ts src\app\services

GITHUBCLIENT
TS repository.component.specs
TS repository.component.ts
TS repository.ts
services
TS api.service.ts
TS user.service.specs
TS user.service.ts
# app.component.css
app.component.html
TS app.component.specs
TS app.component.ts
TS app.module.ts
TS index.ts
TS repository-service.service.specs
TS repository-service.service.ts
assets
environments
favicon.ico
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
PS F:\GitHubFrontend\GitHubClient>

Activate Windows
Go to Settings to activate Windows.

Ln 9, Col 26 Spaces: 4 UTF-8 LF TypeScript 2.8.3 6:05 PM 20/08/2018

Type here to search

```
88     this.userService.getUserFollowers(user.login).subscribe(res => {
89         this.selectedUser.followers = res as User[];
90 
91         this.cacheSelectUser(this.selectedUser);
92 
93         this.loadingFollowers = false;
94     }, err => {
95         console.log(err);
96         this.loadingFollowers = false;
97     });
98 }
99 }

/**
 * we storage selected user and dont again to api. just simply function for now.
 */

cacheSelectUser(user: User) {
    if (!this.findUserInCache(user)) {
        this.cache.selectedUser.push(user);
    }
}
```

app.component.ts - GitHubClient - Visual Studio Code

File Edit Selection View Go Debug Tasks Help

EXPLORER TS app.component.ts x TS api.service.ts TS app.module.ts TS user.ts app.component.html # styles.css TS repository.ts

```
OPEN EDITORS
TS app.component.ts src\app
TS api.service.ts src\app\services
TS app.module.ts src\app
TS user.ts src\app\models
app.component.html src\app
# styles.css src
TS repository.ts src\app\repository
repository.component.html src\app
TS user.service.ts src\app\services

GITHUBCLIENT
TS repository.component.specs
TS repository.component.ts
TS repository.ts
services
TS api.service.ts
TS user.service.specs
TS user.service.ts
# app.component.css
app.component.html
TS app.component.specs
TS app.component.ts
TS app.module.ts
TS index.ts
TS repository-service.service.specs
TS repository-service.service.ts
assets
environments
favicon.ico
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
PS F:\GitHubFrontend\GitHubClient>

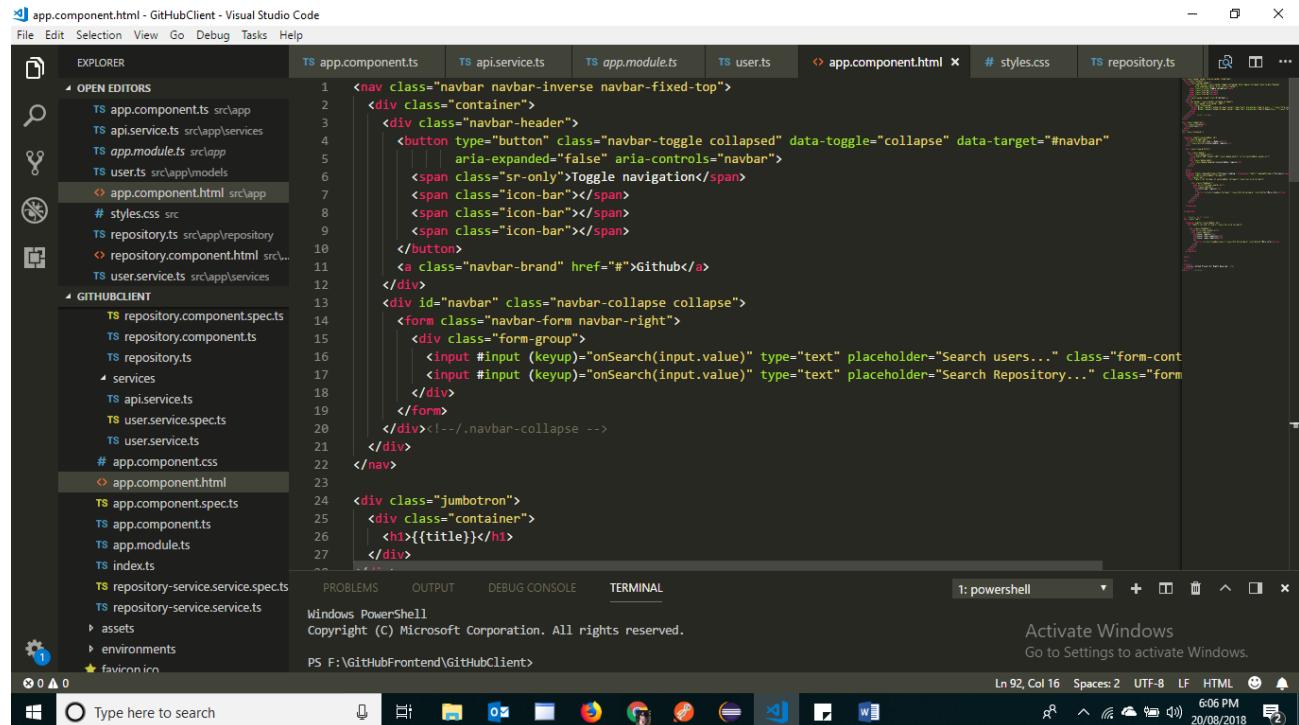
Activate Windows
Go to Settings to activate Windows.

Ln 9, Col 26 Spaces: 4 UTF-8 LF TypeScript 2.8.3 6:06 PM 20/08/2018

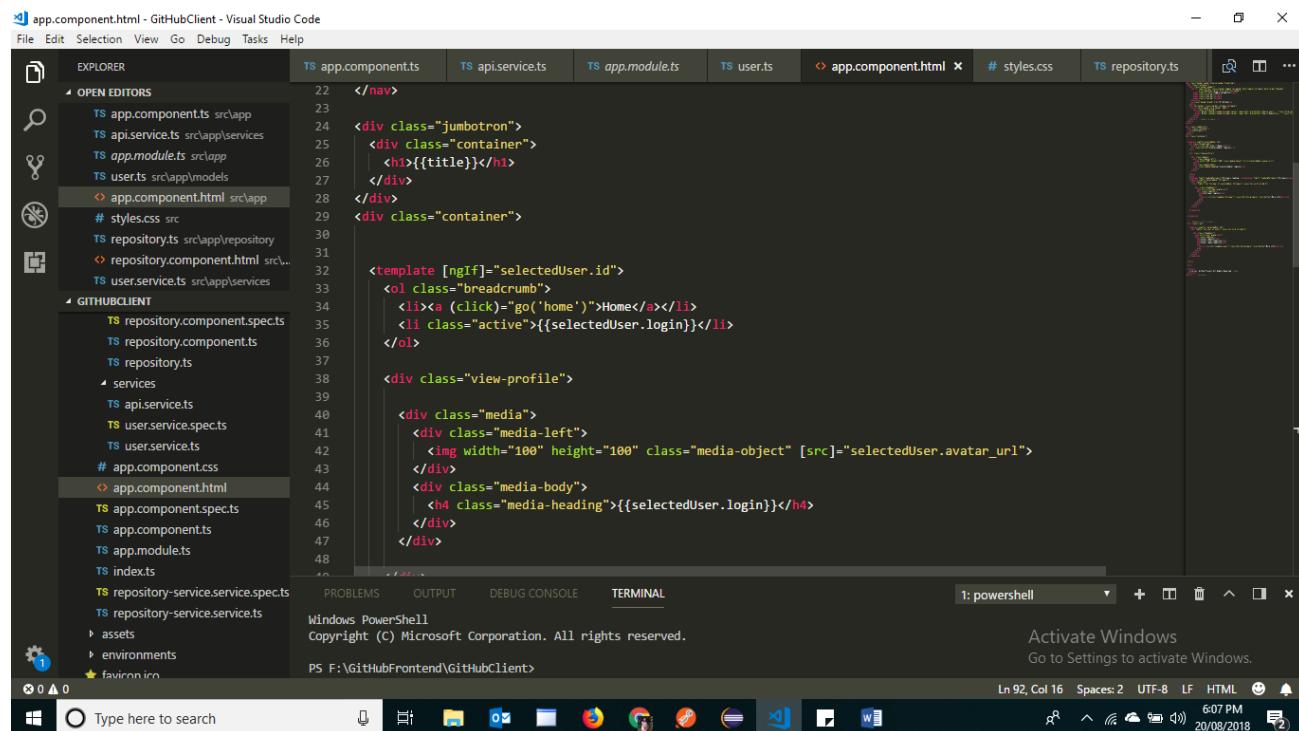
Type here to search

```
106 /**
107 * we storage selected user and dont again to api. just simply function for now.
108 */
109
110 cacheSelectUser(user: User) {
111     if (!this.findUserInCache(user)) {
112         this.cache.selectedUser.push(user);
113     }
114 }
115
116 /**
117 * Find user if exist in cache we return user object
118 * @param user
119 * @returns {boolean}
120 */
121 findUserInCache(user: User): User {
122     for (var i = 0; i < this.cache.selectedUser.length; i++) {
123         if (this.cache.selectedUser[i].login == user.login) {
124             return this.cache.selectedUser[i];
125         }
126     }
127     return null;
128 }
129
130
131
132
133 }
```

Step 10: App.component.html page to display the values to user.



```
<nav class="navbar navbar-inverse navbar-fixed-top">
  <div class="container">
    <button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-target="#navbar"
      aria-expanded="false" aria-controls="navbar">
      <span class="sr-only">Toggle navigation</span>
      <span class="icon-bar"></span>
      <span class="icon-bar"></span>
      <span class="icon-bar"></span>
    </button>
    <a class="navbar-brand" href="#">Github</a>
  </div>
  <div id="navbar" class="navbar-collapse collapse">
    <form class="navbar-form navbar-right">
      <div class="form-group">
        <input #input (keyup)="onSearch(input.value)" type="text" placeholder="Search users..." class="form-control" />
        <input #input (keyup)="onSearch(input.value)" type="text" placeholder="Search Repository..." class="form-control" />
      </div>
    </form>
  </div><!--.navbar-collapse -->
</div>
<div class="jumbotron">
  <div class="container">
    <h1>{{title}}</h1>
  </div>
</div>
```



```
</nav>
<div class="jumbotron">
  <div class="container">
    <h1>{{title}}</h1>
  </div>
</div>
<div class="container">
  <template [ngIf]="selectedUser.id">
    <ol class="breadcrumb">
      <li><a (click)=>go('home')>Home</a></li>
      <li class="active">{{selectedUser.login}}</li>
    </ol>
    <div class="view-profile">
      <div class="media">
        <div class="media-left">
          <img width="100" height="100" class="media-object" [src]="selectedUser.avatar_url">
        </div>
        <div class="media-body">
          <h4 class="media-heading">{{selectedUser.login}}</h4>
        </div>
      </div>
    </div>
  </template>
</div>
```

app.component.html - GitHubClient - Visual Studio Code

File Edit Selection View Go Debug Tasks Help

EXPLORER

- OPEN EDITORS
- TS app.component.ts src\app
- TS api.service.ts src\app\services
- TS app.module.ts src\app
- TS user.ts src\app\models
- ▷ app.component.html src\app
- # styles.css src
- TS repository.ts src\app\repository
- ▷ repository.component.html src\...
- TS user.service.ts src\app\services

GITHUBCLIENT

- TS repository.component.spec.ts
- TS repository.component.ts
- TS repository.ts
- services
- TS api.service.ts
- TS user.service.spec.ts
- TS user.service.ts
- # app.component.css
- ▷ app.component.html
- TS app.component.spec.ts
- TS app.component.ts
- TS app.module.ts
- TS index.ts
- TS repository-service.service.spec.ts
- TS repository-service.service.ts

assets

environments

favicon.ico

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
PS F:\GitHubFrontend\GitHubClient>

Activate Windows
Go to Settings to activate Windows.

Ln 1: powershell + 1: powershell 607 PM 20/08/2018

Type here to search

The screenshot shows the Visual Studio Code interface with the GitHubClient project open. The Explorer sidebar on the left lists files and folders, including app.component.html, repository component files, and service files. The code editor displays the template for the app component's HTML file. The terminal at the bottom shows a PowerShell session running on the project directory. The status bar at the bottom right indicates the line number (1), column (powershell), and date (20/08/2018).

app.component.html - GitHubClient - Visual Studio Code

File Edit Selection View Go Debug Tasks Help

EXPLORER

- OPEN EDITORS
- TS app.component.ts src\app
- TS api.service.ts src\app\services
- TS app.module.ts src\app
- TS user.ts src\app\models
- ▷ app.component.html src\app
- # styles.css src
- TS repository.ts src\app\repository
- ▷ repository.component.html src\...
- TS user.service.ts src\app\services

GITHUBCLIENT

- TS repository.component.spec.ts
- TS repository.component.ts
- TS repository.ts
- services
- TS api.service.ts
- TS user.service.spec.ts
- TS user.service.ts
- # app.component.css
- ▷ app.component.html
- TS app.component.spec.ts
- TS app.component.ts
- TS app.module.ts
- TS index.ts
- TS repository-service.service.spec.ts
- TS repository-service.service.ts

assets

environments

favicon.ico

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
PS F:\GitHubFrontend\GitHubClient>

Activate Windows
Go to Settings to activate Windows.

Ln 1: powershell + 1: powershell 608 PM 20/08/2018

Type here to search

This screenshot is nearly identical to the first one, showing the same project structure and code in the editor. The terminal and status bar also show the same information, indicating no significant changes have occurred between these two moments in time.

The screenshot shows the Visual Studio Code interface with the following details:

- Title Bar:** app.component.html - GitHubClient - Visual Studio Code
- File Menu:** File Edit Selection View Go Debug Tasks Help
- Explorer:** Shows the project structure under "OPEN EDITORS" and "GITHUBCLIENT".
- Code Editor:** Displays the content of `app.component.html`. The code includes Angular components like `<app-header>`, `<app-body>`, and `<app-footer>`, along with user data binding and a button click handler.
- Terminal:** Shows a Windows PowerShell session with the command `PS F:\GitHubFrontend\GitHubClient>`.
- Bottom Bar:** Includes a search bar, pinned icons for various tools, and system status indicators.

Step 11: Style.css for defining the styles:

The screenshot shows the Visual Studio Code interface with the following details:

- Title Bar:** styles.css - GitHubClient - Visual Studio Code
- File Menu:** File Edit Selection View Go Debug Tasks Help
- Explorer:** Shows the project structure under "OPEN EDITORS" and "GITHUBCLIENT".
- Code Editor:** Displays the content of `styles.css`. It contains a single CSS rule: `.breadcrumb { cursor: pointer; }`.
- Terminal:** Shows a Windows PowerShell session with the command `PS F:\GitHubFrontend\GitHubClient>`.
- Bottom Bar:** Includes a search bar, pinned icons for various tools, and system status indicators.

Step 12: Starting the node server and deploying the application by using npm start command

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer (Left):** Shows the project structure under "OPEN EDITORS" and "GITHUBCLIENT".
- Code Editor (Top Center):** Displays a file named "app.component.ts" with the following code:

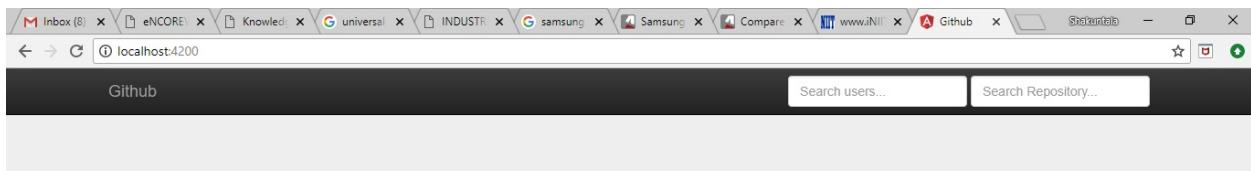
```
1  /* You can add global styles to this file, and also import other style files */
2  .breadcrumb {
3      cursor: pointer;
4 }
```
- Terminal (Bottom):** Shows the command line output for running the application:

```
PS F:\GitHubFrontend\GitHubClient> npm start
> git-hub-client@0.0.0 start F:\GitHubFrontend\GitHubClient
> ng serve

** NG Live Development Server is listening on localhost:4200, open your browser on http://localhost:4200/ **
Date: 2018-08-20T12:42:26.636Z
Hash: 4414ea5082e2ff5358e
Time: 5068ms
chunk {inline} inline.bundle.js (inline) 5.79 kB [entry] [rendered]
chunk {main} main.bundle.js (main) 57.2 kB [initial] [rendered]
chunk {polyfills} polyfills.bundle.js (polyfills) 558 kB [initial] [rendered]
chunk {styles} styles.bundle.js (styles) 33.9 kB [initial] [rendered]
chunk {vendor} vendor.bundle.js (vendor) 11.3 MB [initial] [rendered]

webpack: Compiled successfully.
```
- System Tray (Bottom Right):** Shows system icons and the date/time: 6:12 PM, 20/08/2018.

Step 13: Verifying the output:



A screenshot of a GitHub user profile page for 'PankajSre'. The profile picture is a red and black grid pattern. The name 'PankajSre' is displayed prominently. Below the profile picture, there are sections for 'Followers (6)' and 'Commits (29)'. The followers listed are Jatin1697, anshuman101, BalanVignesh482, tshep159, Aj1402, and anujiji1999. The commits section shows 29 commits by the user. At the bottom right of the page, there is a message from Microsoft: 'Activate Windows Go to Settings to activate Windows.' The browser address bar shows the URL as localhost:4200. The Windows taskbar at the bottom is identical to the one in the previous screenshot.

localhost:4200

Github rakesh13 MusicHubFinal

rakesh13

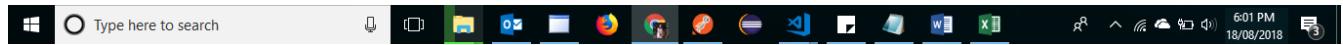
Date : 18-08-2018

Home / rakesh13



[Followers \(0\)](#) [Commits \(21\)](#)

Activate Windows
Go to Settings to activate Windows.



localhost:4200

Github rakesh13 MusicHubFinal

rakesh13

Date : 18-08-2018

rakesh13 [More info](#)

rakesh1313 [More info](#)

rakesh1310 [More info](#)

rakesh1312 [More info](#)

Rakesh1307 [More info](#)

Rakesh1325 [More info](#)

Activate Windows
Go to Settings to activate Windows.

The screenshot shows a web browser window with the URL [@ localhost:4200](http://localhost:4200). The title bar says "Github". The main content area displays the GitHub profile for "rakesh13". It includes a profile picture of a man with a mustache, a bio section, and a "Followers (0)" count.



The screenshot shows a web browser window with the URL [@ localhost:4200](http://localhost:4200). The title bar says "Github". The main content area displays the GitHub profile for "PankajSre". It includes a profile picture of a red "DEV" logo, a bio section, and a "Followers (6)" count. Below this, there is a grid of six follower profiles, each with a small profile picture and a name: Jatin1697, anshuman101, BalanViqnesh482, tshep159, Aj1402, and anujji1999.

