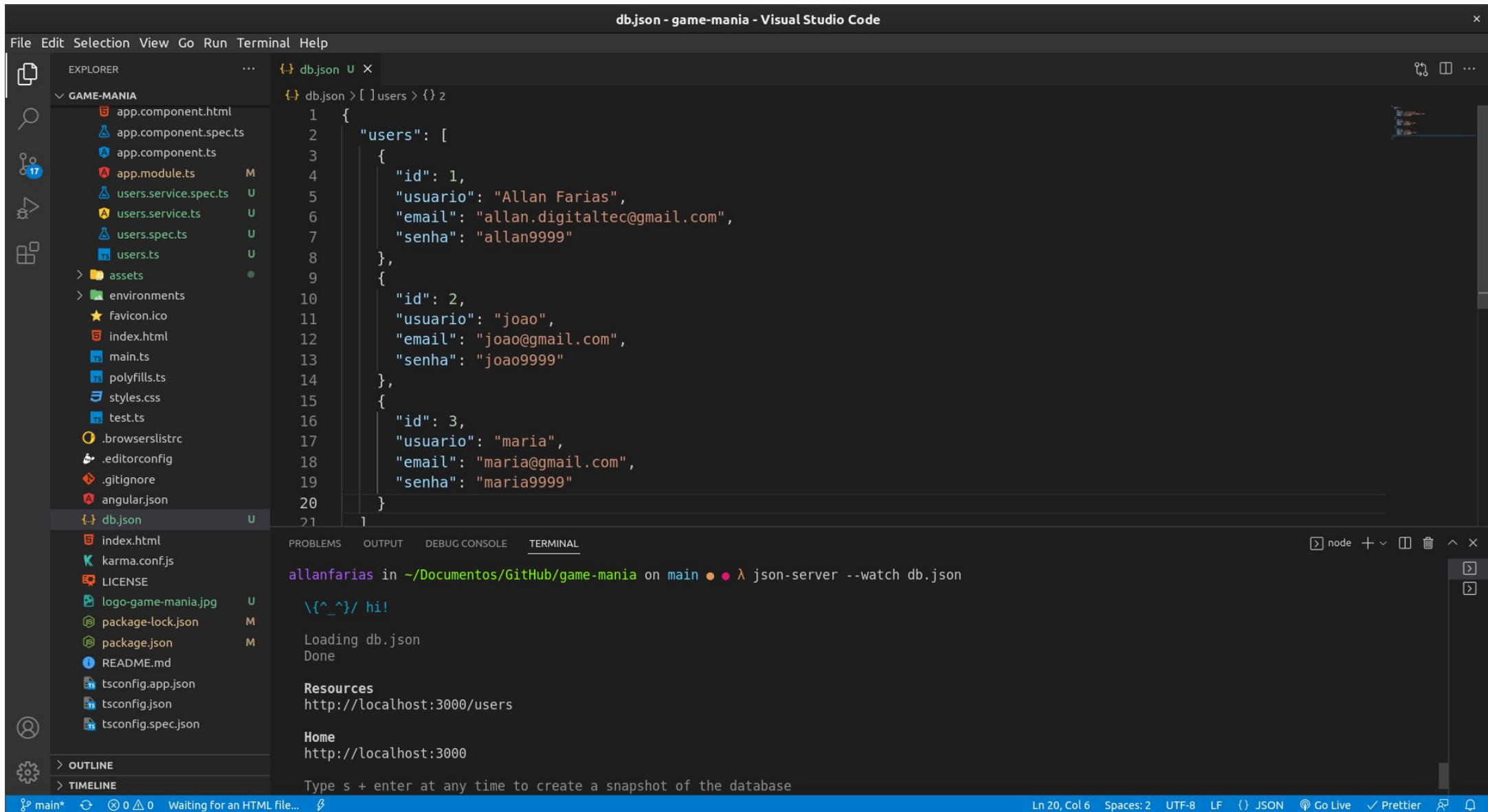


TESTE – API REST

Iniciando o fake servidor com o json-server:



The screenshot shows the Visual Studio Code interface with a file explorer on the left, a code editor in the center, and a terminal at the bottom. The file explorer shows a project named 'GAME-MANIA' with various files including 'app.component.html', 'app.component.spec.ts', 'app.component.ts', 'app.module.ts', 'users.service.spec.ts', 'users.service.ts', 'users.spec.ts', 'users.ts', 'assets', 'environments', 'favicon.ico', 'index.html', 'main.ts', 'polyfills.ts', 'styles.css', 'test.ts', '.browserslistrc', '.editorconfig', '.gitignore', 'angular.json', 'db.json', 'index.html', 'karma.conf.js', 'LICENSE', 'logo-game-mania.jpg', 'package-lock.json', 'package.json', 'README.md', 'tsconfig.app.json', 'tsconfig.json', and 'tsconfig.spec.json'. The code editor shows the content of 'db.json', which is a JSON array of three user objects. The terminal shows the command 'json-server --watch db.json' being executed, and the output indicates that the server is running on 'http://localhost:3000/users'.

```
db.json - game-mania - Visual Studio Code
```

File Edit Selection View Go Run Terminal Help

EXPLORER

GAME-MANIA

- app.component.html
- app.component.spec.ts
- app.component.ts
- app.module.ts
- users.service.spec.ts
- users.service.ts
- users.spec.ts
- users.ts
- assets
- environments
- favicon.ico
- index.html
- main.ts
- polyfills.ts
- styles.css
- test.ts
- .browserslistrc
- .editorconfig
- .gitignore
- angular.json
- db.json
- index.html
- karma.conf.js
- LICENSE
- logo-game-mania.jpg
- package-lock.json
- package.json
- README.md
- tsconfig.app.json
- tsconfig.json
- tsconfig.spec.json

db.json

```
{
  "users": [
    {
      "id": 1,
      "usuario": "Allan Farias",
      "email": "allan.digitaltec@gmail.com",
      "senha": "allan9999"
    },
    {
      "id": 2,
      "usuario": "joao",
      "email": "joao@gmail.com",
      "senha": "joao9999"
    },
    {
      "id": 3,
      "usuario": "maria",
      "email": "maria@gmail.com",
      "senha": "maria9999"
    }
  ]
}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

node

```
allanfarias in ~/Documentos/GitHub/game-mania on main ● λ json-server --watch db.json
\{^_^\}/ hi!
Loading db.json
Done
Resources
http://localhost:3000/users
Home
http://localhost:3000
Type s + enter at any time to create a snapshot of the database
```

Ln 20, Col 6 Spaces: 2 UTF-8 LF {} JSON Go Live ✓ Prettier

Realizando os testes (CRUD) com o Postman

GET

The screenshot shows the Postman application interface. The top bar includes the Postman logo, a search bar, and buttons for 'Invite', 'Upgrade', and 'Send'. The left sidebar shows the 'My Workspace' section with a collection named 'Game Mania' containing a request named 'GET Teste de API - Json Server'. The main panel shows the request details for the 'GET' method to 'localhost:3000/users'. The 'Body' tab is selected, showing a JSON response with three user objects. The status bar at the bottom indicates 'Status: 200 OK', 'Time: 28 ms', and 'Size: 723 B'.

Request Details:

- Method: GET
- URL: localhost:3000/users
- Body Type: x-www-form-urlencoded

Response Body (JSON):

```
1 {
2   {
3     "id": 1,
4     "usuario": "Allan Farias",
5     "email": "allan.digitaltec@gmail.com",
6     "senha": "allan9999"
7   },
8   {
9     "usuario": "joao",
10    "email": "joao@gmail.com",
11    "senha": "joao9999",
12    "id": 2
13  },
14  {
15    "id": 3,
16    "usuario": "maria",
17    "email": "maria@gmail.com",
18    "senha": "maria9999"
19  }
20 }
```

POST – INSERIR NOVO USUARIO

The screenshot displays the Postman application interface. The top navigation bar includes 'File', 'Edit', 'View', and 'Help'. Below it, the 'My Workspace' section shows a collection named 'Game Mania' containing a request named 'Teste de API - Json Server'. The request is a POST method to the URL 'localhost:3000/users'. The 'Body' tab is selected, showing a JSON body with the following data:

KEY	VALUE	DESCRIPTION
id		
usuario	Novo Usuario	
senha	usuario9999	

The response is displayed in the 'Body' tab, showing a JSON object with the following data:

```
1 {
2   "id": 4,
3   "usuario": "Novo Usuario",
4   "senha": "usuario9999"
5 }
```

The status bar at the bottom indicates the request was successful with a status of 201 Created, a time of 40 ms, and a size of 574 B. The 'Save Response' button is visible.

Resultado no Json Server

The screenshot shows the Visual Studio Code interface with the file `db.json` open in the editor. The file contains a JSON array of user objects. The Explorer sidebar on the left shows the project structure for `GAME-MANIA`, including files like `app.component.html`, `app.component.spec.ts`, `app.component.ts`, `app.module.ts`, `users.service.spec.ts`, `users.service.ts`, `users.spec.ts`, `users.ts`, `assets`, `environments`, `favicon.ico`, `index.html`, `main.ts`, `polyfills.ts`, `styles.css`, `test.ts`, `.browserslistrc`, `.editorconfig`, `.gitignore`, `angular.json`, `db.json`, `index.html`, `karma.conf.js`, `LICENSE`, `logo-game-mania.jpg`, `package-lock.json`, `package.json`, `README.md`, `tsconfig.app.json`, `tsconfig.json`, and `tsconfig.spec.json`.

The `db.json` file content is as follows:

```
{
  "usuarios": [
    {
      "id": 1,
      "usuario": "allan",
      "email": "allan.digitaltec@gmail.com",
      "senha": "allan9999"
    },
    {
      "id": 2,
      "usuario": "joao",
      "email": "joao@gmail.com",
      "senha": "joao9999"
    },
    {
      "id": 3,
      "usuario": "maria",
      "email": "maria@gmail.com",
      "senha": "maria9999"
    },
    {
      "id": 4,
      "usuario": "Novo Usuario",
      "senha": "usuario9999"
    }
  ]
}
```

The Terminal at the bottom shows the command `json-server --watch db.json` being executed. The output indicates that the server is running and listening on `http://localhost:3000/users`. The status bar at the bottom shows the current file is `main*` and the editor is waiting for an HTML file.

PUT – ATUALIZAR USUÁRIO 4

The screenshot displays the Postman interface for a PUT request. The workspace is named "My Workspace" and the collection is "Game Mania". The request is titled "Teste de API - Json Server" and is directed to "localhost:3000/users/4". The request body is a JSON object with the following fields:

KEY	VALUE	DESCRIPTION
id	4	
usuario	Update Usuario	
senha	usuario9999	

The response status is 200 OK, with a time of 14 ms and a size of 465 B. The response body is a JSON object with the following fields:

```
1 {
2   "id": 4,
3   "usuario": "Update Usuario",
4   "senha": "usuario9999"
5 }
```

Resultado no Json Server

The screenshot shows the Visual Studio Code interface with the file explorer on the left, the editor in the center, and the terminal at the bottom. The file explorer shows the project structure for 'GAME-MANIA'. The editor displays the content of 'db.json', which is a JSON array of user objects. The terminal shows the command 'json-server -w db.json' being executed, and the output indicates that the server is running and listening on port 3000.

```
db.json - game-mania - Visual Studio Code
```

File Edit Selection View Go Run Terminal Help

EXPLORER

- GAME-MANIA
 - app.component.html
 - app.component.spec.ts
 - app.component.ts
 - app.module.ts
 - users.service.spec.ts
 - users.service.ts
 - users.spec.ts
 - users.ts
 - assets
 - environments
 - favicon.ico
 - index.html
 - main.ts
 - polyfills.ts
 - styles.css
 - test.ts
 - .browserslistrc
 - .editorconfig
 - .gitignore
 - angular.json
 - db.json
 - index.html
 - karma.conf.js
 - LICENSE
 - logo-game-mania.jpg
 - package-lock.json
 - package.json
 - README.md
 - tsconfig.app.json
 - tsconfig.json
 - tsconfig.spec.json

db.json

```
{
  "usuarios": [
    {
      "id": 1,
      "usuario": "allan",
      "email": "allan.digitaltec@gmail.com",
      "senha": "allan9999"
    },
    {
      "id": 2,
      "usuario": "joao",
      "email": "joao@gmail.com",
      "senha": "joao9999"
    },
    {
      "id": 3,
      "usuario": "maria",
      "email": "maria@gmail.com",
      "senha": "maria9999"
    },
    {
      "id": 4,
      "usuario": "Update Usuario",
      "senha": "usuario9999"
    }
  ]
}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
allanfarias in ~/Documentos/GitHub/game-mania on main ●●λ json-server -w db.json
{^_^}/ hi!
Loading db.json
Done
Resources
http://localhost:3000/users
Home
http://localhost:3000
Type s + enter at any time to create a snapshot of the database
```

Ln 27, Col 2 Spaces: 2 UTF-8 LF {} JSON Go Live ✓ Prettier

PATCH – ATUALIZAR SENHA DO USUÁRIO 4

The screenshot shows the Postman application interface. The top bar includes the menu (File, Edit, View, Help), navigation tabs (Home, Workspaces, API Network, Reports, Explore), a search bar, and utility buttons (Invite, Upgrade). The left sidebar displays the workspace structure: My Workspace (New, Import), Collections (Game Mania), APIs, Environments, Mock Servers, Monitors, Flows, and History.

The main panel shows a PATCH request to `localhost:3000/users/4`. The request is configured with the following details:

- Method:** PATCH
- URL:** localhost:3000/users/4
- Body Type:** x-www-form-urlencoded
- Body Content:**

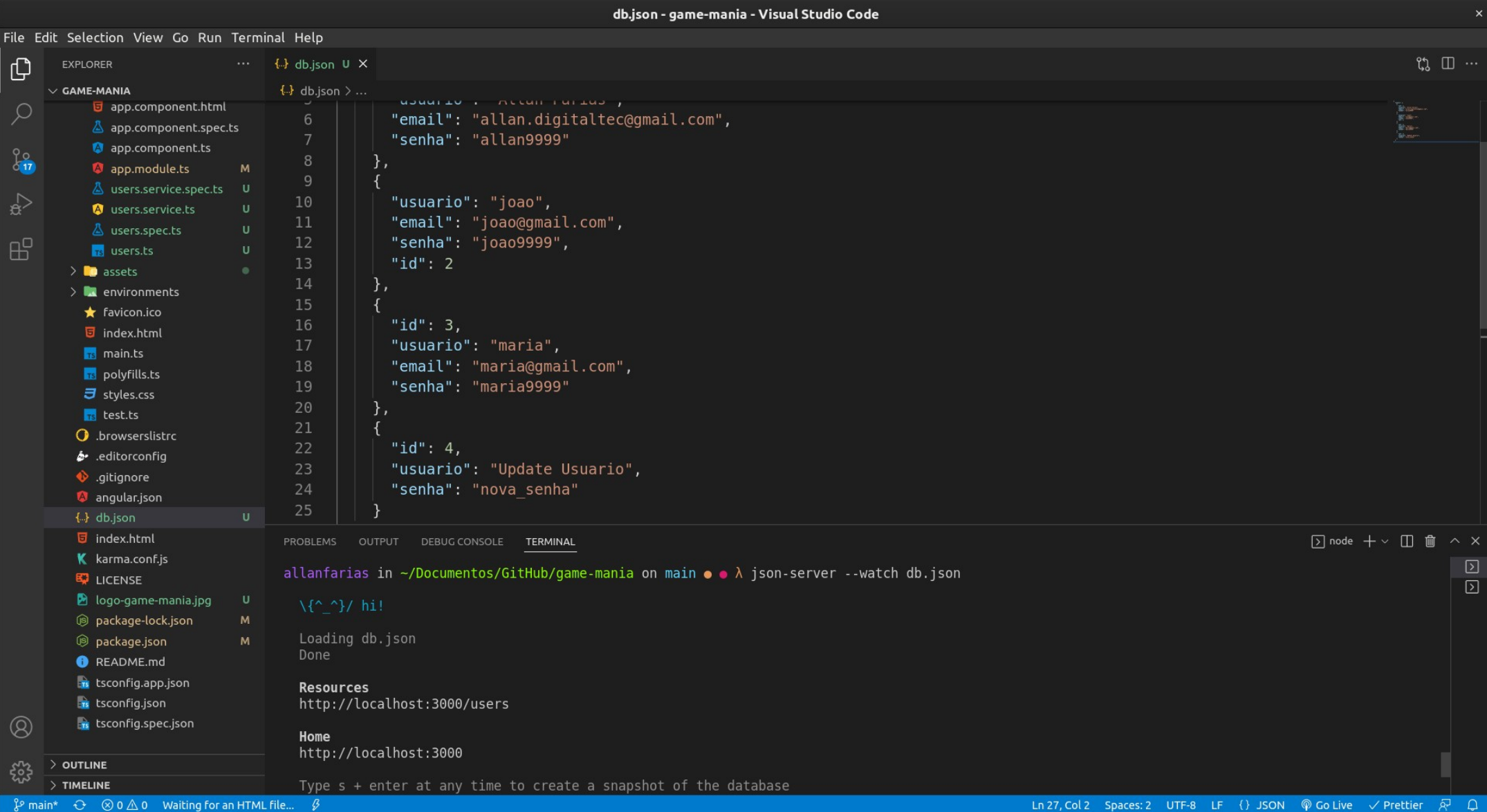
KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> id	4	
<input checked="" type="checkbox"/> usuario	Update Usuario	
<input checked="" type="checkbox"/> senha	nova_senha	
Key	Value	Description

The response section shows the following details:

- Status:** 200 OK
- Time:** 8 ms
- Size:** 464 B
- Body Type:** JSON

```
1 {
2   "id": 4,
3   "usuario": "Update Usuario",
4   "senha": "nova_senha"
5 }
```

Resultado no Json Server



The screenshot shows the Visual Studio Code interface with the file `db.json` open in the editor. The file contains a JSON array of user objects. The terminal at the bottom shows the command `json-server --watch db.json` being executed, and the output displays the JSON data and the server's status.

```
db.json - game-mania - Visual Studio Code
```

File Edit Selection View Go Run Terminal Help

EXPLORER

- GAME-MANIA
 - app.component.html
 - app.component.spec.ts
 - app.component.ts
 - app.module.ts
 - users.service.spec.ts
 - users.service.ts
 - users.spec.ts
 - users.ts
 - assets
 - environments
 - favicon.ico
 - index.html
 - main.ts
 - polyfills.ts
 - styles.css
 - test.ts
 - .browserslistrc
 - .editorconfig
 - .gitignore
 - angular.json
 - db.json
 - index.html
 - karma.conf.js
 - LICENSE
 - logo-game-mania.jpg
 - package-lock.json
 - package.json
 - README.md
 - tsconfig.app.json
 - tsconfig.json
 - tsconfig.spec.json

db.json

```
{
  "usuarios": [
    {
      "id": 1,
      "usuario": "allanfarias",
      "email": "allan.digitaltec@gmail.com",
      "senha": "allan9999"
    },
    {
      "id": 2,
      "usuario": "joao",
      "email": "joao@gmail.com",
      "senha": "joao9999"
    },
    {
      "id": 3,
      "usuario": "maria",
      "email": "maria@gmail.com",
      "senha": "maria9999"
    },
    {
      "id": 4,
      "usuario": "Update Usuario",
      "senha": "nova_senha"
    }
  ]
}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
allanfarias in ~/Documentos/GitHub/game-mania on main ●●λ json-server --watch db.json

\{^_^}/ hi!

Loading db.json
Done

Resources
http://localhost:3000/users

Home
http://localhost:3000

Type s + enter at any time to create a snapshot of the database
```

main* 0 0 Waiting for an HTML file... Ln 27, Col 2 Spaces: 2 UTF-8 LF {} JSON Go Live ✓ Prettier

DELETE – DELETAR USUÁRIO 4

The screenshot displays the Postman application interface. The top navigation bar includes 'File', 'Edit', 'View', and 'Help'. Below this, a secondary bar contains 'Home', 'Workspaces', 'API Network', 'Reports', and 'Explore', along with a search bar and utility buttons like 'Invite' and 'Upgrade'. The left sidebar, titled 'My Workspace', lists various collections and environments, with 'Game Mania' expanded to show 'Teste de API - Json Server'. The main workspace shows a 'DELETE' request to 'localhost:3000/users/4'. The 'Body' tab is selected, indicating no body is present. The bottom section shows the response status as '200 OK' with a size of '395 B'. The bottom status bar includes options for 'Find and Replace', 'Console', 'Cookies', 'Capture requests', 'Bootcamp', 'Runner', 'Trash', and a help icon.

Postman

File Edit View Help

Home Workspaces API Network Reports Explore

Search Postman

My Workspace New Import

Game Mania / Teste de API - Json Server

Save Send

DELETE localhost:3000/users/4

Params Authorization Headers (6) Body Pre-request Script Tests Settings

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

This request does not have a body

Body Cookies Headers (13) Test Results

Status: 200 OK Time: 6 ms Size: 395 B Save Response

Pretty Raw Preview Visualize JSON

1

Find and Replace Console Cookies Capture requests Bootcamp Runner Trash

Resultado no Json Server

The screenshot shows the Visual Studio Code interface with the file `db.json` open. The Explorer sidebar on the left shows the project structure for `GAME-MANIA`, including files like `app.component.html`, `app.component.spec.ts`, `app.component.ts`, `app.module.ts`, `users.service.spec.ts`, `users.service.ts`, `users.spec.ts`, `users.ts`, `assets`, `environments`, `favicon.ico`, `index.html`, `main.ts`, `polyfills.ts`, `styles.css`, `test.ts`, `.browserslistrc`, `.editorconfig`, `.gitignore`, `angular.json`, `db.json`, `index.html`, `karma.conf.js`, `LICENSE`, `logo-game-mania.jpg`, `package-lock.json`, `package.json`, `README.md`, `tsconfig.app.json`, `tsconfig.json`, and `tsconfig.spec.json`.

The `db.json` file contains the following JSON data:

```
5  "usuario": "allanfarias",
6  "email": "allan.digitaltec@gmail.com",
7  "senha": "allan9999"
8  },
9  {
10   "usuario": "joao",
11   "email": "joao@gmail.com",
12   "senha": "joao9999",
13   "id": 2
14 },
15 {
16   "id": 3,
17   "usuario": "maria",
18   "email": "maria@gmail.com",
19   "senha": "maria9999"
20 }
21 ]
22 }
```

The TERMINAL panel at the bottom shows the command `json-server --watch db.json` being executed. The output indicates that the server is running and listening on `http://localhost:3000/users`. The status bar at the bottom shows the file is at `Ln 22, Col 2` in `UTF-8` encoding, with `JSON` format and `Prettier` formatter.

Resultados dos comandos no Console

The image shows a Visual Studio Code editor window titled "db.json - game-mania - Visual Studio Code". The Explorer sidebar on the left shows a project named "GAME-MANIA" with various files including "app.component.html", "app.component.spec.ts", "app.component.ts", "app.module.ts", "users.service.spec.ts", "users.service.ts", "users.spec.ts", "users.ts", "assets", "environments", "favicon.ico", "index.html", "main.ts", "polyfills.ts", "styles.css", "test.ts", ".browserslistrc", ".editorconfig", ".gitignore", "angular.json", "db.json", "index.html", "karma.conf.js", "LICENSE", "logo-game-mania.jpg", "package-lock.json", "package.json", "README.md", "tsconfig.app.json", "tsconfig.json", and "tsconfig.spec.json".

The main editor area displays the content of "db.json", which is a JSON array of user objects:

```
6  {  
7    "email": "allan.digitaltec@gmail.com",  
8    "senha": "allan9999"  
9  },  
10 {  
11   "usuario": "joao",  
12   "email": "joao@gmail.com",  
13   "senha": "joao9999",  
14   "id": 2  
15 },  
16 {  
17   "id": 3,  
18   "usuario": "maria",  
19   "email": "maria@gmail.com",  
20   "senha": "maria9999"  
21 }  
22 ]
```

The bottom panel shows the "TERMINAL" tab with the following output:

```
node + v  
http://localhost:3000  
  
Type s + enter at any time to create a snapshot of the database  
Watching...  
  
GET /users 200 4.909 ms - 326  
POST /users 201 12.728 ms - 68  
PUT /users/4 200 5.092 ms - 69  
DELETE /users/4 200 3.755 ms - 2  
POST /users 201 3.788 ms - 68  
PUT /users/4 200 3.057 ms - 70  
PATCH /users/4 200 3.404 ms - 69  
DELETE /users/4 200 1.925 ms - 2
```

The status bar at the bottom indicates the current file is "main*", the encoding is "UTF-8", and the language is "JSON".