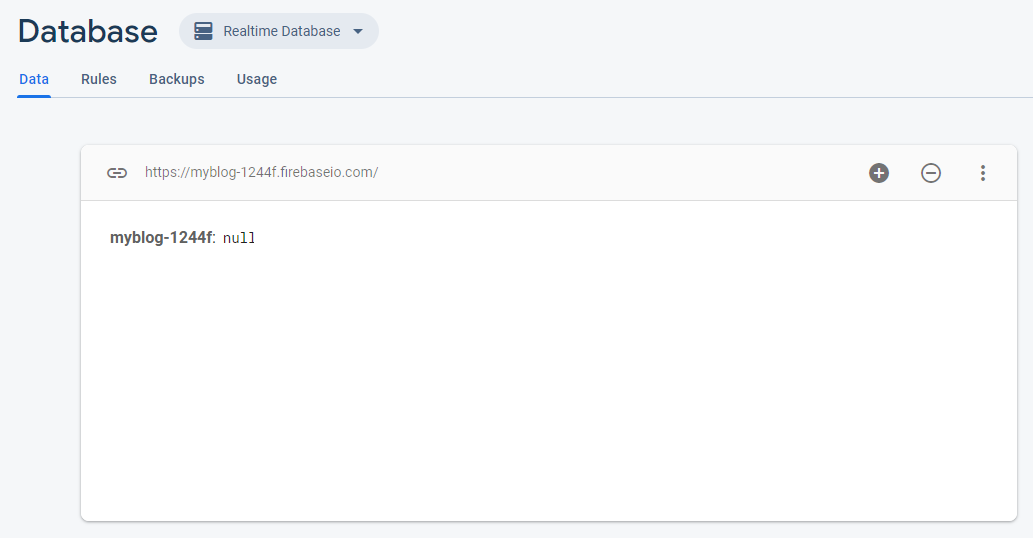
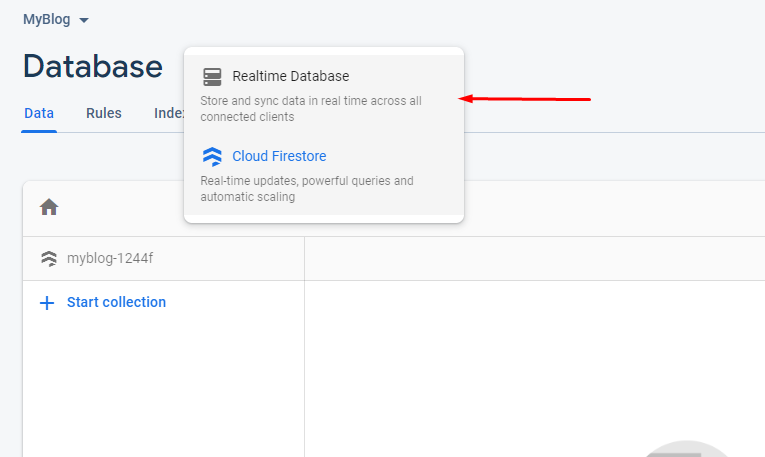
**JS Applications Exam – MyBlog SPA**

You are assigned to implement a **Web application** (SPA) using HTML5, JavaScript, AJAX, REST and JSON with cloud-based backend (Firebase or Kinvey). Using libraries like **jQuery**, **Handlebars** and **Sammy** is allowed but is **not obligatory**. The app keeps **users** and **posts**. **Users** should be able to **register** and **login**. Logged-in users should be able to view **their posts**, **create** **posts**, see **details** about an postand **logout**. Logged-in users should also be able to **edit** or **delete** the articles **they have created**. Create a Firebase REST Service

Register at **Firebase.com** and create application to keep your data in the cloud.

Create a collection called **posts.** Each **post** has a **title**, **category**, **content**.



## Test the Firebase REST Services

Using **Postman** or other HTTP client tool, test the REST service end points:

### List All Posts

|  |
| --- |
| **GET** https://{apiKey}.com/{uid}/posts.json |

### Create Post

|  |  |
| --- | --- |
| **POST** https://{apiKey}.firebaseio.com/{uid}/posts.json | |
| **Request body** | {  "title": "Javascript",  "category": "Programming",  "content": "Lorem ipsum dolor sit amet, consectetur...",  } |
| **Response**  **201 Created** | {Key}:{  "title": "Javascript",  "category": "Programming",  "content": "Lorem ipsum dolor sit amet, consectetur...",  } |
| **Error response**  **401 Unauthorized** | { "error": "InvalidCredentials", "description": "Invalid credentials. Please retry your request with correct credentials", "debug": "" } |

### Edit Post

|  |  |
| --- | --- |
| **PUT** https://{apiKey}.com/{uid}/posts/{id} | |
| **Request body** | {  "title": "Arrays",  "category": "JavaScript",  "content": "Lorem ipsum dolor sit amet, consectetur...",  "creator-email": "peter.georgiev@email.com"  } |
| **Response**  **200 Ok** | {key}:{  "title": "Arrays",  "category": "JavaScript",  "content": "Lorem ipsum dolor sit amet, consectetur...",  "creator-email": "peter.georgiev@email.com"  } |
| **Error response**  **401 Unauthorized** | { "error": "InvalidCredentials", "description": "Invalid credentials. Please retry your request with correct credentials", "debug": "" } |

### Delete Post

|  |  |
| --- | --- |
| **DELETE** https://{apiKey}.com/{uid}/posts/{id} | |
| **Error response**  **404 Not Found** | { "error": "EntityNotFound", "description": "This entity not found in the collection", "debug": "" } |
| **Error response**  **401 Unauthorized** | { "error": "InvalidCredentials", "description": "Invalid credentials. Please retry your request with correct credentials", "debug": "" } |

**MyBlog – HTML and CSS**

You have been given the web design of the application as **HTML** + **CSS** files.

* Initially all views and forms are shown by the HTML. Your application may **hide**/**show elements** by CSS (display: none) or **delete**/**reattach** from and to the DOM all unneeded elements, or just display the views it needs to display.
* You may render the views/forms/components with **JavaScript** or **Handlebars**.

**Important**: Don’t change the elements’ **class names** and **ids**. Don’t rename form fields/link names/ids. You are **allowed** to add **data attributes** to any elements. You may modify href **attributes** of links and add action**/**method **attributes** to **forms**, to allow the use of a routing library.

## MyBlog - Client-Side Web Application

**Design** and **implement** a client-side front-end app (SPA) for managing **articles**. Implement the functionality described below.

**Navigation Bar (5 Pts)**

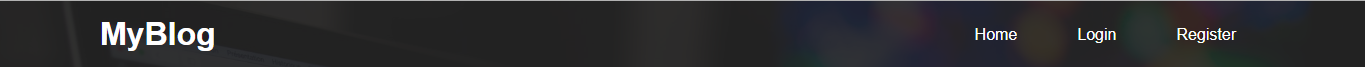
Navigation links should correctly change the current page (view).

* Clicking on the links in the **Header** should display the view behind the link (views are represented as sections in the HTML code).
* Your application may **hide**/**show elements** by CSS (display: none) or **delete**/**reattach** from and to the DOM all unneeded elements, or just display the views it needs to display.

The Logged-in user navbar should contain the following elements: **[Email] and [Logout]**



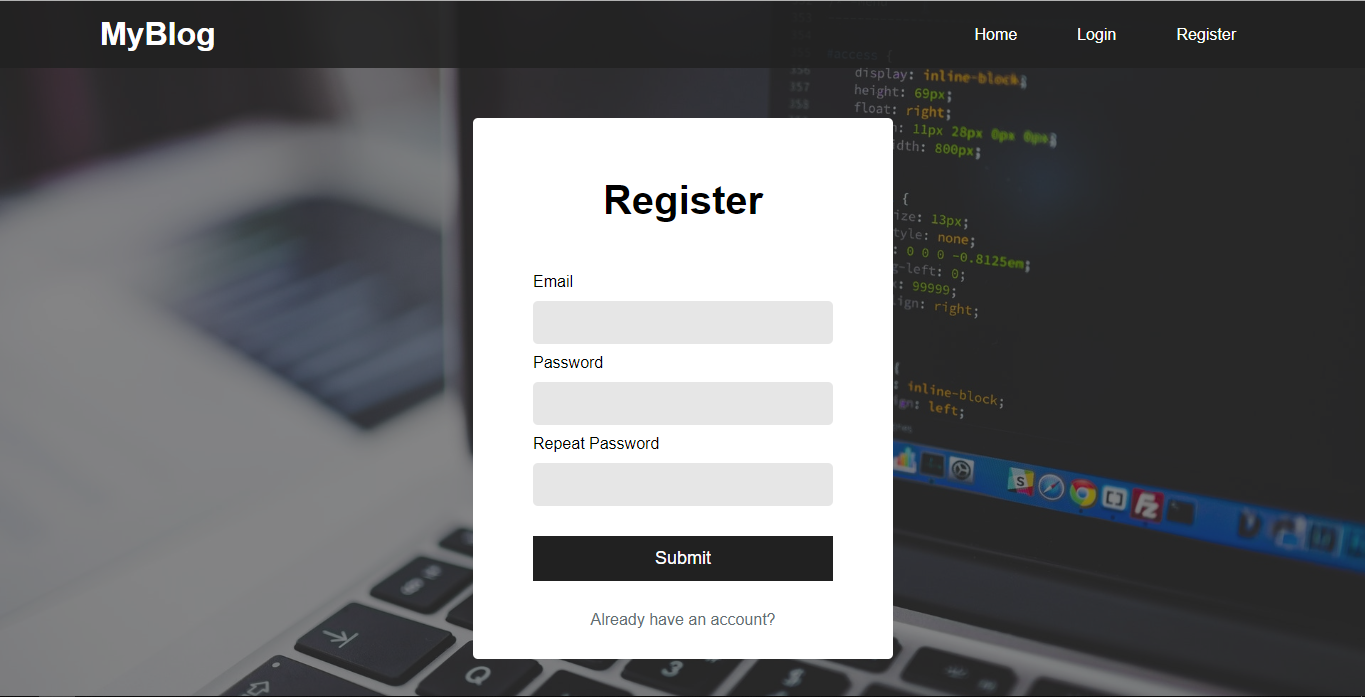
The guest users header should contain the following elements: [Home], [Login] and [Register].



**Register User (5 Pts)**

By given **email** and **password,** the app should register a new user in the system and should **redirect you to the login page.**

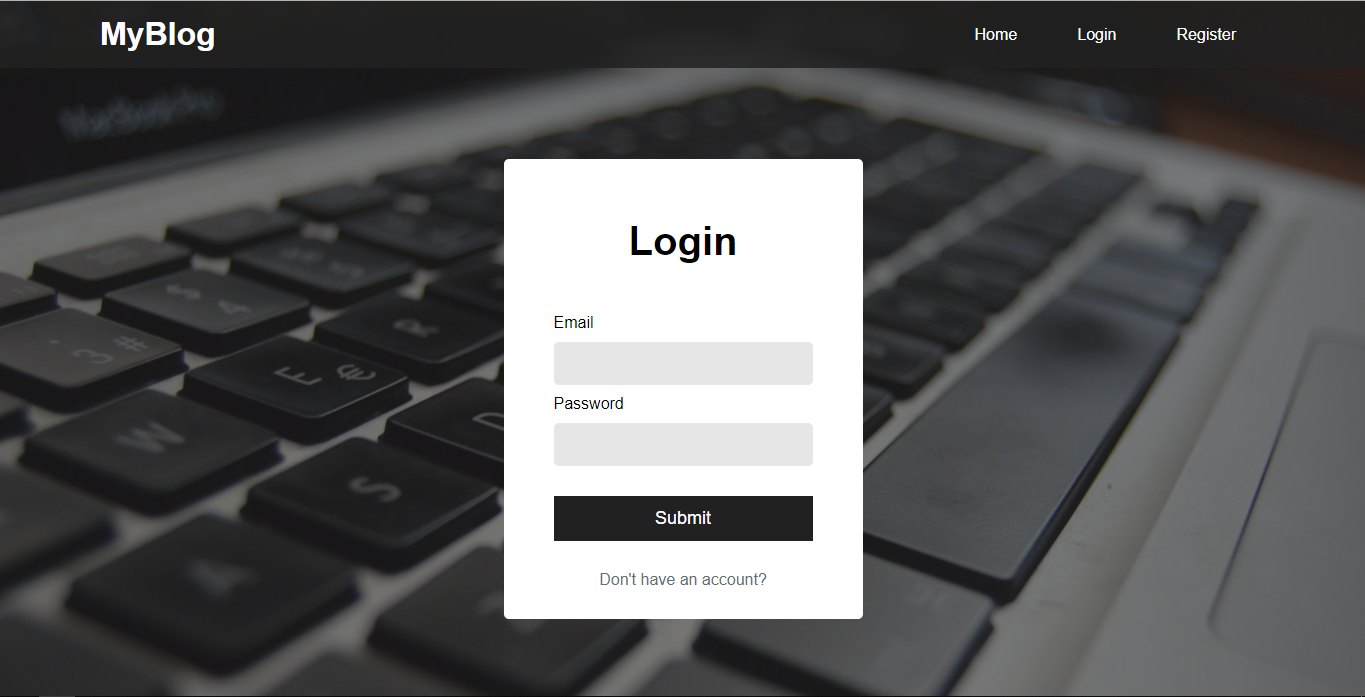
* Keep the user local data in the browser's **local/session storage**.



**Login User (5 Pts)**

By given **email** and **password,** the app should login an existing user.

* Keep the user local data in the browser's **local storage**.
* After a successful login **redirect to the home page**.



**Logout (5 Pts)**

Successfully logged in users should be able to **logout** from the app.

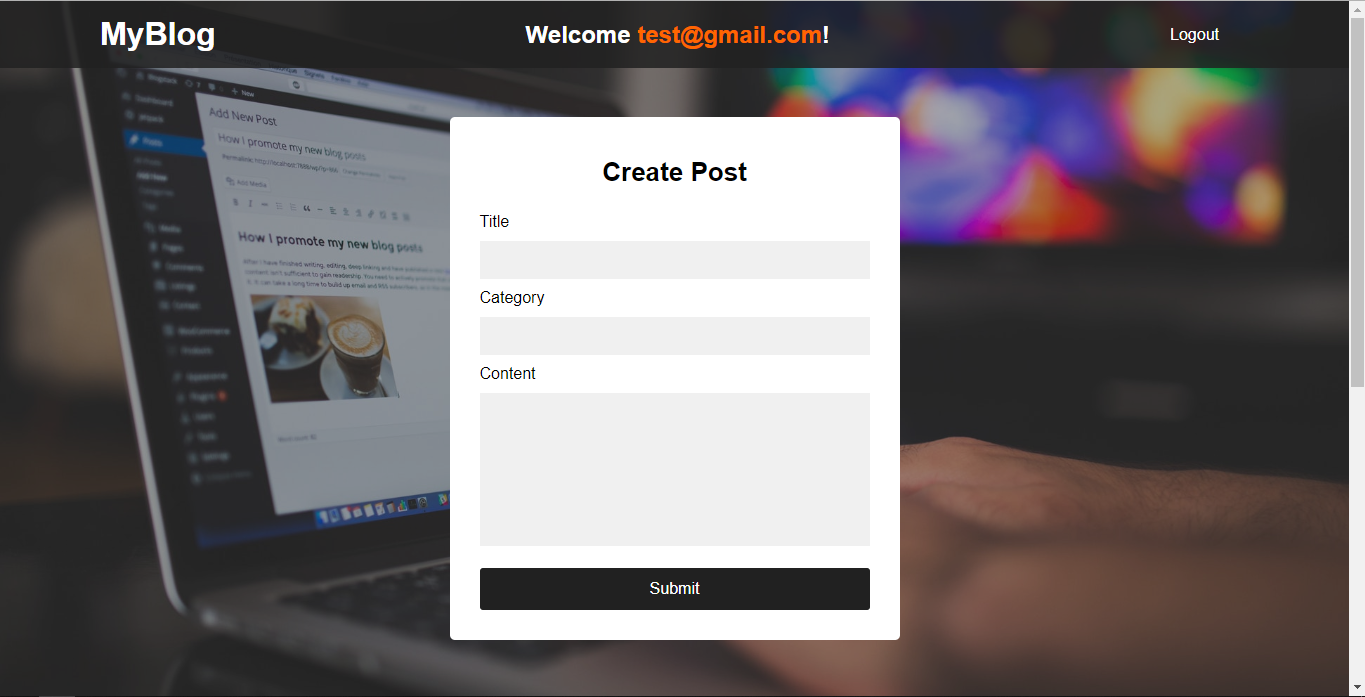
* The **"logout" REST service** at the back-end **must** be called at logout
* All local/session information in the browser (**user local data**) about the current user should be deleted
* After a successful logout **redirect to the home page**.

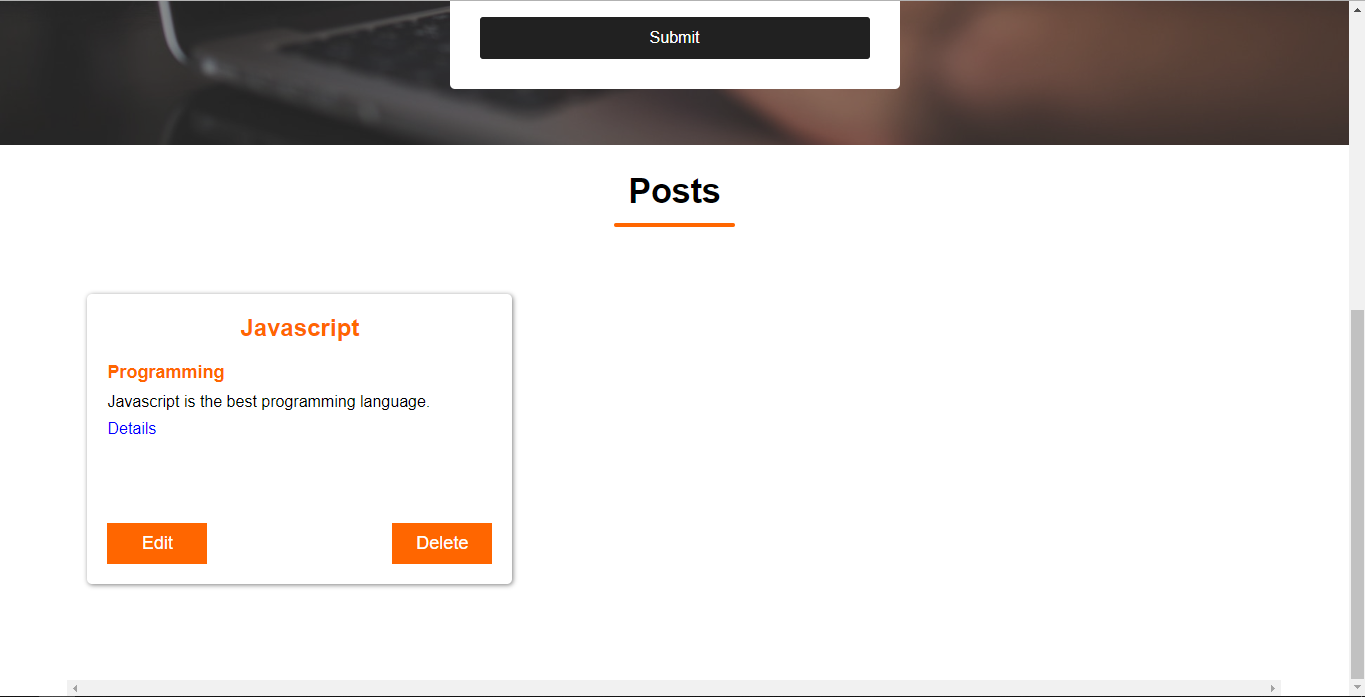
**Home Page (40 Pts)**

Guest users should be welcomed by the Homepage. They should be able to navigate to register or login.



Successfully logged-in users should be welcomed by the Homepage. They should be able to create and see all their posts.



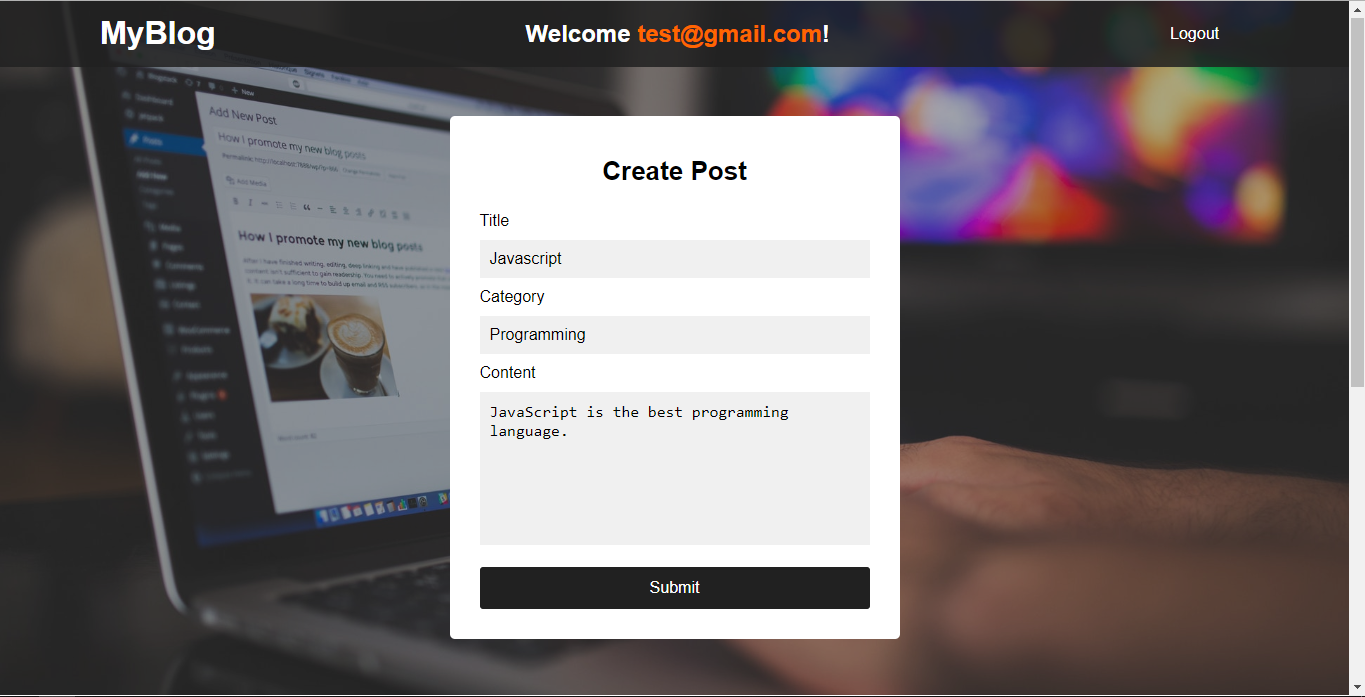


If there are no posts, posts section should be empty.

**Create Post (10 Pts)**

Logged-in users should be able to **Create posts**.

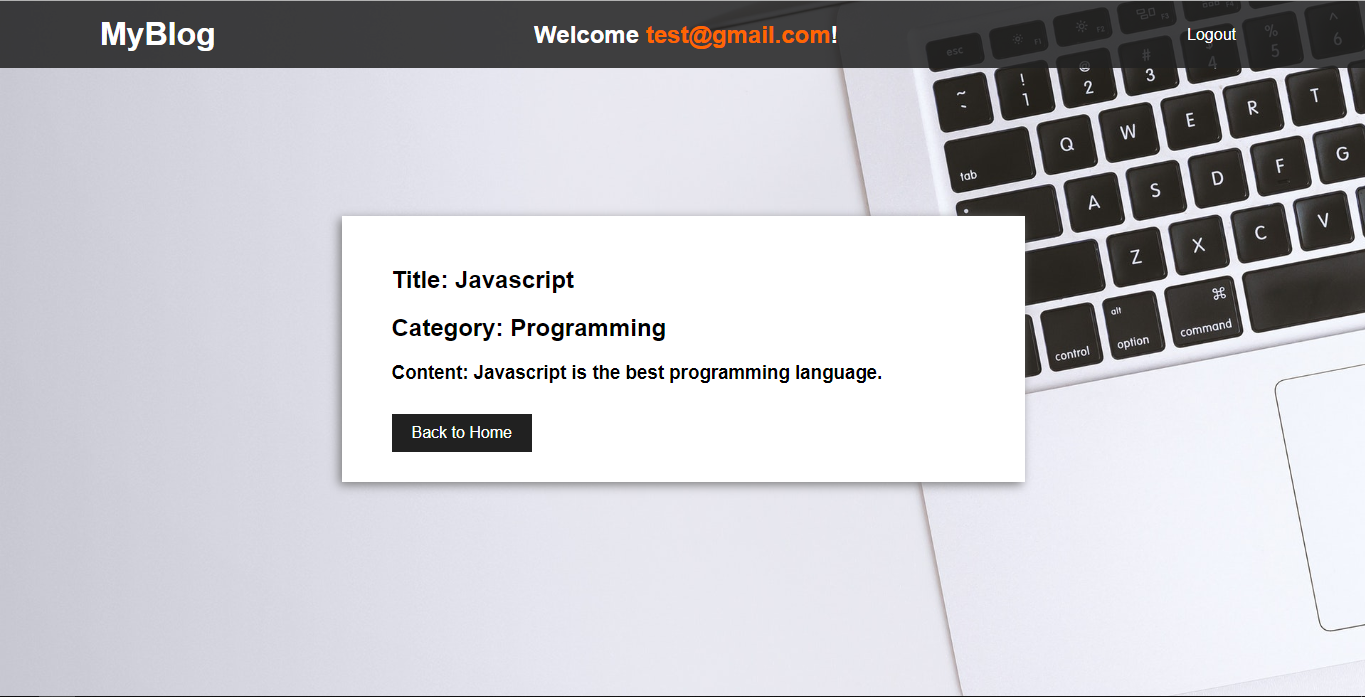
* The newly added post should be stored in the Firebase\Kinvey collection "posts".
* After a post is created successfully you should be redirected to the **Home Page**.



**Details (15 Pts)**

Logged-in users should be able to **view details** abouttheir posts.

Clicking the [Details] **link** in of a **particular post** should **display** the Posts Details page.

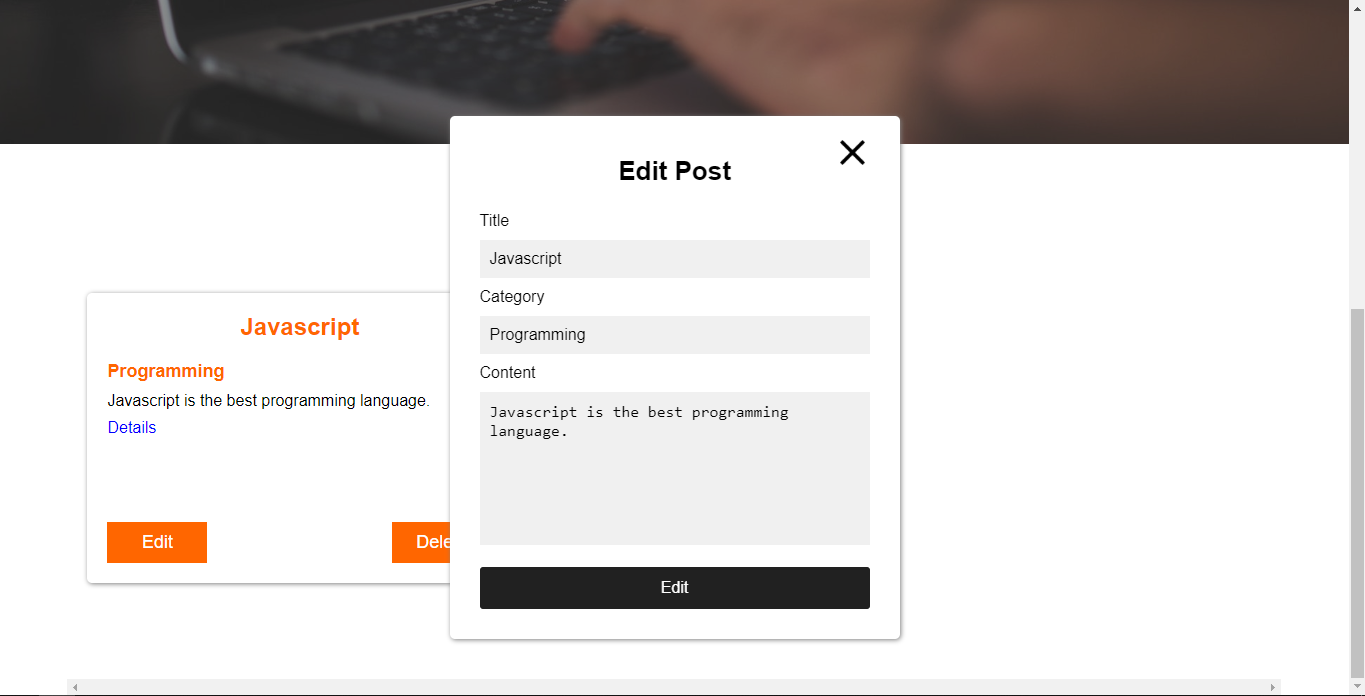


**Edit Article (10 Pts)**

Logged-in users should be able to **edit** their **own** posts.

Clicking the [Edit] **button** of a **particular post** on the Posts section should **display** the Edit page:

* After a **successful edit** the user should be redirected to the Home page.



**Delete Post (5 Pts)**

Logged-in users should be able to **delete their** articles.

Clicking the [Delete] **button** of an **particular post** should **delete** the **post**.

* After **successful** **post delete** you should redirect to the Home page**.**

**(BONUS) Notifications (5 Pts)**

The application should notify the users about the result of their actions.

* In case of a **successful** action, a **notification message** **(green)** should be shown, which disappears automatically after 5 seconds.



* In case of **error**, an **error notification message (red)** should be shown, which disappears automatically after 5 seconds.



* **NOTE**: You get all the points if **all** the notifications have the **exact content** as described in each section above.

## Submitting Your Solution

Exclude the node\_modules folder and ZIP your project. Upload the archive to Judge system.

