

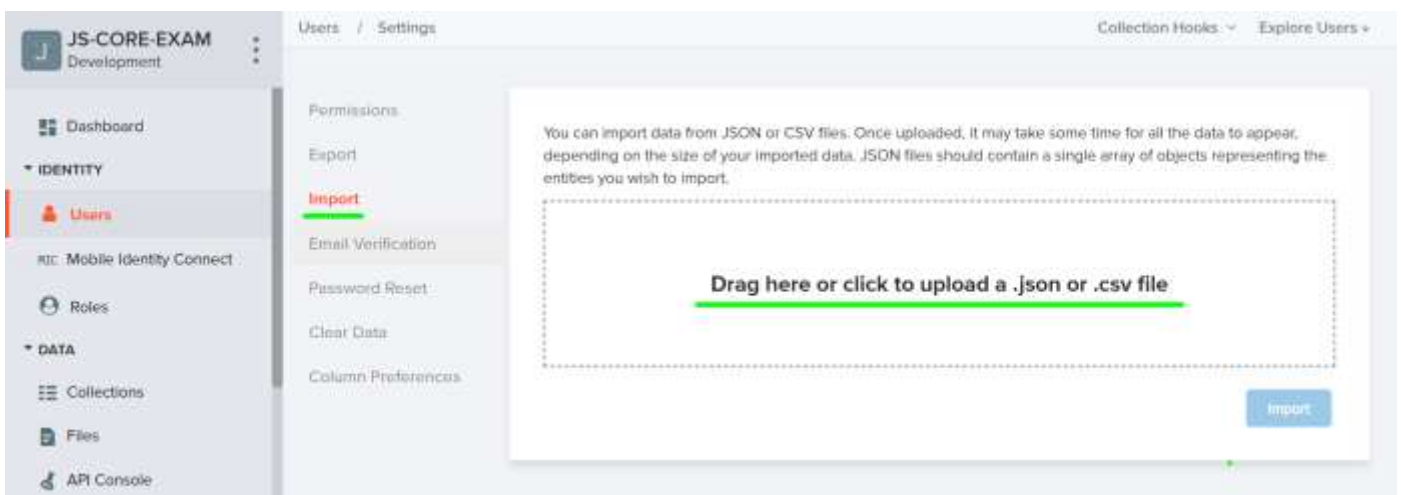
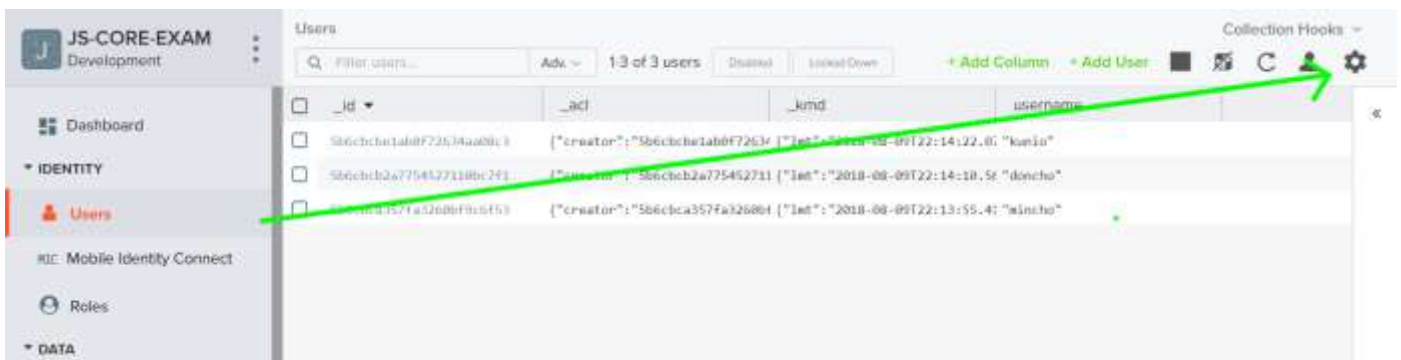
JS Apps Exam – CarTube Single Page Application

Using libraries like **jQuery**, **Handlebars** and **Sammy** is allowed but is not obligatory. You are assigned to implement a **Web application** (SPA) using HTML5, JavaScript, AJAX, REST and JSON with cloud-based backend (Kinvey). The app keeps **users** and **listings**. Users can **register**, **login**, **logout**, view a **catalog** with all **cars** sorted by **time** (a helper function will be given), **create** a car listing, **edit/delete** their **own** listings, show listing **details** and a **section** where they can view their **own** listings.

Problem 1. Create a Kinvey REST Service

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

In the **Users** collection, import the provided JSON file with sample users to get started with template data. In the **Kinvey Console**, select **Users** from the navigation of the left, click **Settings** in the upper right then scroll down to the **Import** section:



Create a collection **cars**. Each **car** has a **title** ,**description** ,**imageUrl** ,**brand** ,**model** ,**fuel** ,**year** ,**price** ,**description** and an **seller** (username of the seller).



You are also given **one** more json files to **import** data for **cars**.

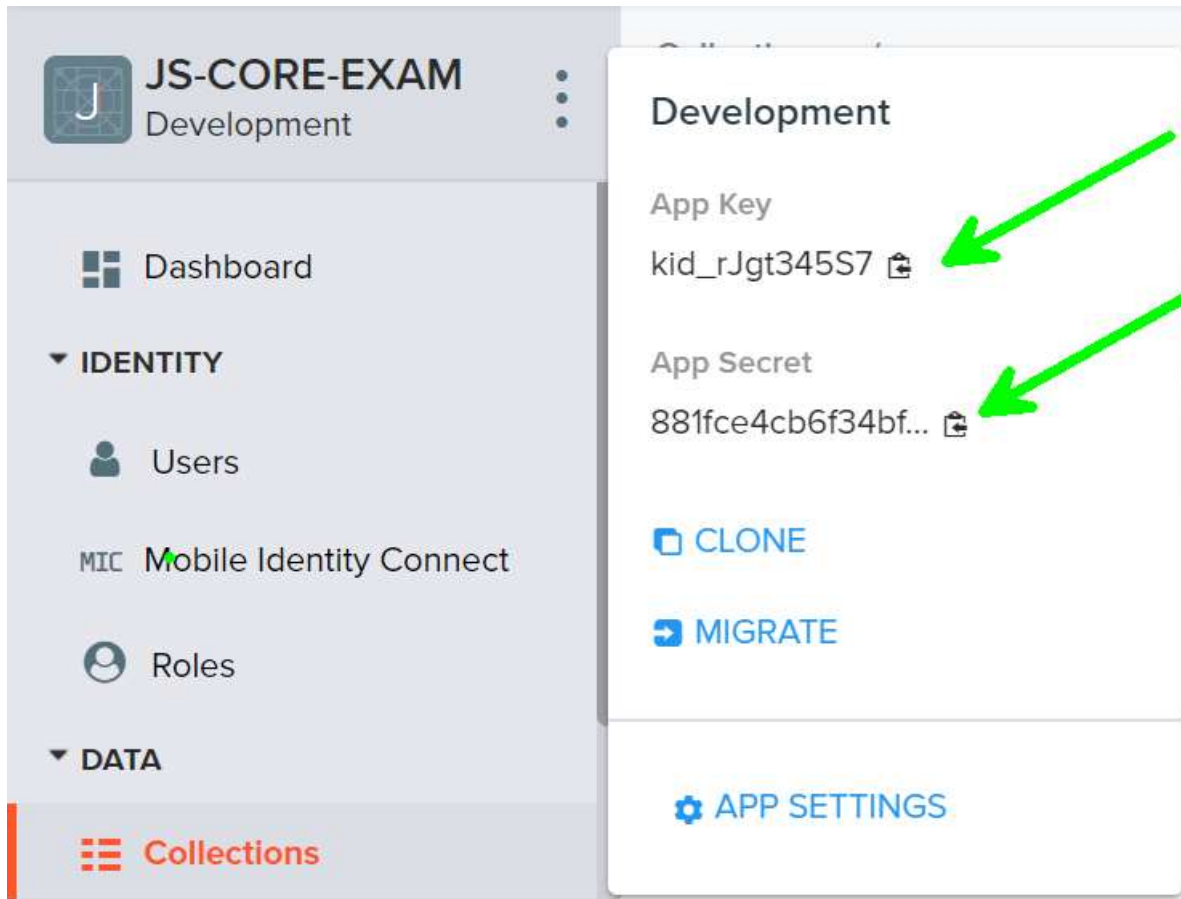
Problem 2. Test the Kinvey REST Services

Using **Postman** or other HTTP client tool (you can use Kinvey's built-in **API Console**), test the REST service endpoints:

User Registration (Sign Up)

POST https://baas.kinvey.com/user/app_id/	
Request headers	Authorization: Basic base64(app_id:app_secret) Content-Type: application/json
Request body	{ "username": "testuser", "password": "testuserpass890" }
Response 201 Created	{ "_id": "59930c78a743e20c7d3fca77", "username": "testuser", "password": "testuserpass890" }
Error response 409 Conflict	{ "error": "UserAlreadyExists", "description": "This username is already taken. Please retry your request with a different username", "debug": "" }
Error response 401 Unauthorized	{ "error": "InvalidCredentials", "description": "Invalid credentials. Please retry your request with correct credentials", "debug": "" }

The request needs “**Basic**” authentication. Use the Kinvey **App Key** and Kinvey **App Secret** as credentials.



User Login

POST https://baas.kinvey.com/user/app_id/login	
Request headers	Authorization: Basic base64(app_id:app_secret) Content-Type: application/json
Request body	{ "username": "testuser", "password": "testuserpass890" }
Response 200 OK	{ "_id": "59930c78a743e20c7d3fca77", "username": "testuser" "_kmd": { "authtoken": "8e6471bc-3712-4cfb-b92e-50e62a0c80....Duj5fHdM /7XHle6KdY="
Error response 401 Unauthorized	{ "error": "InvalidCredentials", "description": "Invalid credentials. Please retry your request with correct credentials", "debug": "" }

Successful login returns an **"authtoken"** which is later used to authenticate the CRUD operations.

User Logout

POST https://baas.kinvey.com/user/app_id/logout
--

Request headers	Authorization: Kinvey authtoken
Response 204 No Content	
Error response 401 Unauthorized	{ "error": "InvalidCredentials", "description": "Invalid credentials. Please retry your request with correct credentials", "debug": "" }

To logout, you need to provide the “**authtoken**” given by login / register as “**Kinvey**” authorization header.

List all Cars (Catalog – sorted by car Listing time, descending)

GET <a -1}"="" _kmd.ect":="" href="https://baas.kinvey.com/appdata/app_id/cars?query={}&sort={">https://baas.kinvey.com/appdata/app_id/cars?query={}&sort={"_kmd.ect": -1}	
Request headers	Authorization: Kinvey authtoken
Response 200 OK	[{ "brand": "BMW", "description": "Нов внос, изобщо не е бълскана никога! Качваш се и караш!!!", "fuel": "Gasoline", "imageUrl": "url-link", "isAuthor": true, "model": "E92", "price": "14000", "seller": "mincho", "title": "3er Coupe (E92) Нов внос!", "year": "2007", "_acl": { "creator": "kid_B1zfi_e_Z" }, "_kmd": { "lmt": "2017-08-15T14:40:19.182Z", "ect": "2017-08-15T14:40:19.182Z" } }, ...]
Error response 401 Unauthorized	{ "error": "InvalidCredentials", "description": "Invalid credentials. Please retry your request with correct credentials", "debug": "" }

Create Car Listing

POST https://baas.kinvey.com/appdata/app_id/cars	
Request headers	Authorization: Kinvey authtoken Content-Type: application/json
Request body	{ "brand": "BMW", "description": "Нов внос, изобщо не е бълскана никога! Качваш се и караш!!!", "fuel": "Gasoline", "imageUrl": "url-link", "isAuthor": true, "model": "E92", "price": "14000", "seller": "mincho", "title": "3er Coupe (E92) Нов внос!", }

	<pre>"year": "2007", }</pre>
Response 201 Created	<pre>{ "_id": "59931398996ab5127d2a84d1", "brand": "BMW", "description": "Нов внос, изобщо не е бълскана никога! Качваш се и караш!!!", "fuel": "Gasoline", "imageUrl": "url-link", "isAuthor": true, "model": "E92", "price": "14000", "seller": "mincho", "title": "3er Coupe (E92) Нов внос!", "year": "2007", }</pre>
Error response 401 Unauthorized	<pre>{ "error": "InvalidCredentials", "description": "Invalid credentials. Please retry your request with correct credentials", "debug": "" }</pre>

Edit Car Listing

PUT https://baas.kinvey.com/appdata/app_id/cars/car_id	
Request headers	Authorization: Kinvey authtoken Content-Type: application/json
Request body	<pre>{ "brand": "BMW", "description": "Нов внос, изобщо не е бълскана никога! Качваш се и караш!!!", "fuel": "Gasoline", "imageUrl": "url-link", "isAuthor": true, "model": "E92", "price": "14000", "seller": "mincho", "title": "3er Coupe (E92) Нов внос!", "year": "2007", }</pre>
Response 200 Ok	<pre>{ "_id": "59931398996ab5127d2a84d1", "brand": "BMW", "description": "Нов внос, изобщо не е бълскана никога! Качваш се и караш!!!", "fuel": "Gasoline", "imageUrl": "url-link", "isAuthor": true, "model": "E92", "price": "14000", "seller": "mincho", "title": "3er Coupe (E92) Нов внос!", "year": "2007", }</pre>
Error response 401 Unauthorized	<pre>{ "error": "InvalidCredentials", "description": "Invalid credentials. Please retry your request with correct credentials", "debug": "" }</pre>

Delete Car Listing

DELETE https://baas.kinvey.com/appdata/app_id/cars/car_id	
Request headers	Authorization: Kinvey authtoken
Response 200 OK	{ "count": 1 }
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": "" }

My Car Listings (View all Car Listings by logged in user, sorted by post time, descending)

GET <a -1}"="" href="https://baas.kinvey.com/appdata/app_id/cars?query={" seller":"username"}&sort='{"_kmd.ect":'>https://baas.kinvey.com/appdata/app_id/cars?query={"seller":"username"}&sort={"_kmd.ect": -1}	
Request headers	Authorization: Kinvey authtoken
Response 200 OK	[{ "_id": "599307d36af42b495f87bad2", "brand": "BMW", "description": "Нов внос, изобщо не е бълскана никога! Качваш се и караш!!!", "fuel": "Gasoline", "imageUrl": "url-link", "isAuthor": true, "model": "E92", "price": "14000", "seller": "mincho", "title": "3er Coupe (E92) Нов внос!", "year": "2007", "_acl": { "creator": "kid_B1zfi_e_Z" }, "_kmd": { "lmt": "2017-08-15T14:40:19.182Z", "ect": "2017-08-15T14:40:19.182Z" } }, { "_id": "5993178dbce993cf532bbe29", "brand": "BMW", "description": "Нов внос, изобщо не е бълскана никога! Качваш се и караш!!!", "fuel": "Gasoline", "imageUrl": "url-link", "isAuthor": true, "model": "E92", "price": "14000", "seller": "mincho", "title": "3er Coupe (E92) Нов внос!", "year": "2007", "_acl": { "creator": "kid_B1zfi_e_Z" }, "_kmd": { "lmt": "2017-08-15T14:40:19.182Z", "ect": "2017-08-15T14:40:19.182Z" } }]

	<pre> "creator": "59930c78a743e20c7d3fca77" }, "_kmd": { "lmt": "2017-08-15T15:47:25.376Z", "ect": "2017-08-15T15:47:25.376Z" } } }] </pre>
Error response 401 Unauthorized	<pre> { "error": "InvalidCredentials", "description": "Invalid credentials. Please retry your request with correct credentials", "debug": "" } </pre>

Problem 3. CarTube – HTML and CSS

You are 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** by CSS (display: none) or **delete** from the DOM all unneeded elements or just display the views it needs to display.
- You may render the views / forms / components with **jQuery** or **Handlebars**.

Important: don't change the elements' **class name** and **id**. 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.

Problem 4. CarTube Client-Side Web Application

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

Notifications

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

- In case of successful action an **informational (green) notification message** should be shown, which disappears automatically after 3 seconds or manually when the user clicks it.

Logout successful.

- In case of **error**, an **error notification message** (red) should be shown which disappears on user click.

Error: Invalid credentials. Please retry your request with correct credentials

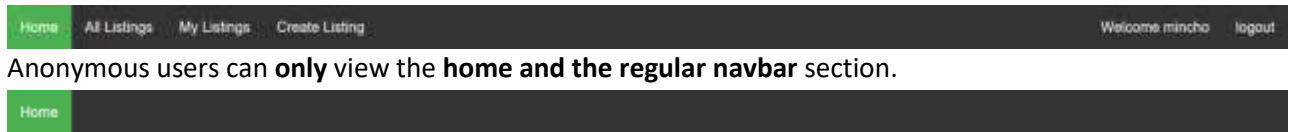
- During the AJAX calls a **loading notification message (blue)** should be shown. It should disappear automatically as soon as the AJAX call is completed.

Loading ...

NavBar (5 pts)

Implement a **NavBar** for the app: navigation links should correctly change the current screen (view).

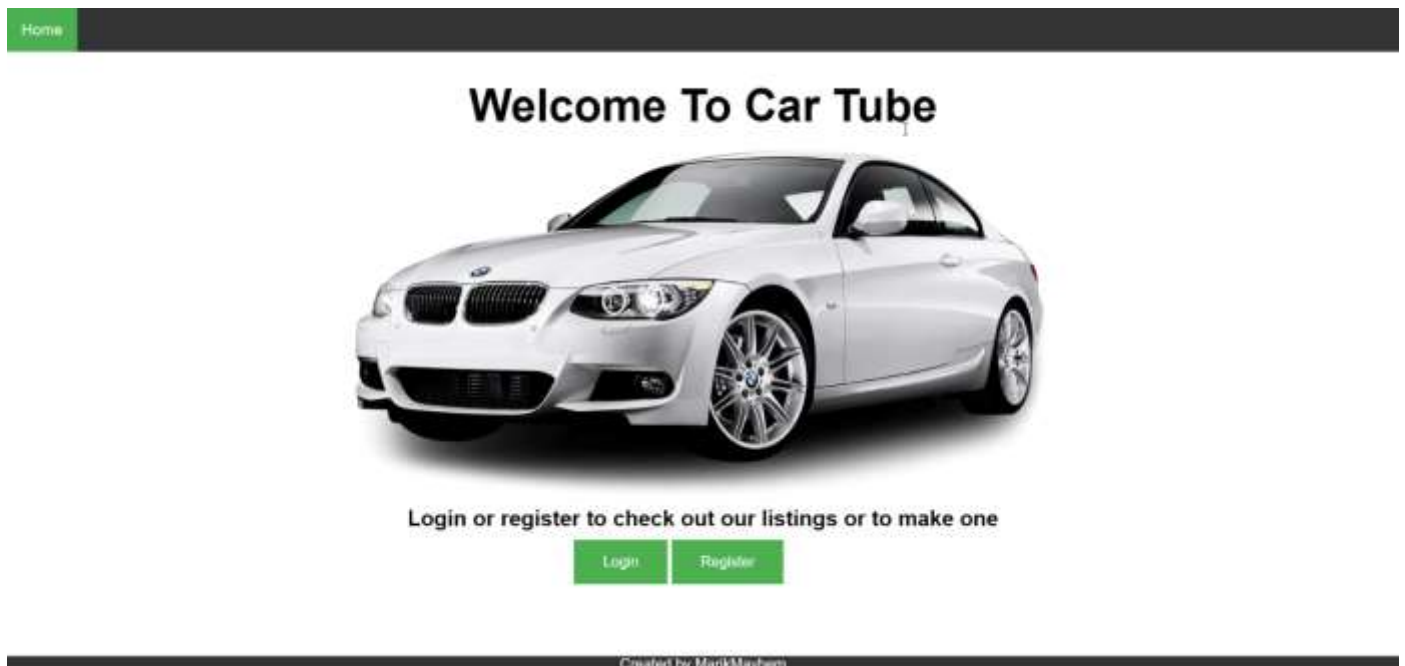
- Clicking on the links in the **NavBar** or the links on each **individual** listing should display the view behind the link (views are sections in the HTML code).
- Your application may **hide** by CSS (display: none) or **delete** from the DOM all unneeded elements or just display the views it needs to display.
- The given „**All Listings, My Listings, Create Listings, Welcome and logout**“ should be visible **only** for logged in users.



Anonymous users can **only** view the **home and the regular navbar** section.

Home Screen (10 pts)

The initial screen should display the navigation, register, login and the initial image + footer.



Register User (10 pts)

By given **username, password and repeat password** the app should register a new user in the system.

- After a **successful registration**, a notification message “User registration successful.” should be displayed and the **catalog** with all **listings** should be shown.
- You **need** to validate the **input**. A username **should** be at **least** 3 characters **long** and should contain only english alphabet **letters**. A user’s password should be at **least** 6 characters **long** and should contain only english alphabet **letters** and **digits**. Both passwords **must** match.

- In case of **error** (eg. invalid username/password), an appropriate error **message** should be displayed and the user should be able to **try** to register again.
- Keep the user session data in the browser's **session storage**.

[Home](#)

Register

Please fill in this form to create an account.

Username

Password

Repeat Password

Register

Already have an account? [Sign in](#)

Created by MarkMayhem

Login User (5 pts)

By given **username** and **password** the app should be able to login an existing user.

- After a **successful login**, a notification message "Login successful." should be displayed and the user home screen should be displayed.
- In case of **error**, an appropriate error message should be displayed and the user should be able to fill the login form again.
- **Form validation** should be the **same** as register.
- Keep the user session data in the browser's **session storage**.
- Clear **all** input fields after **successful** login.

[Home](#)

Invalid credentials. Please retry your request with correct credentials.

Login

Please enter your credentials.

Username

Password

Login

Dont have an account? [Sign up](#)

Created by MarkMayhem

Logout (5 pts)

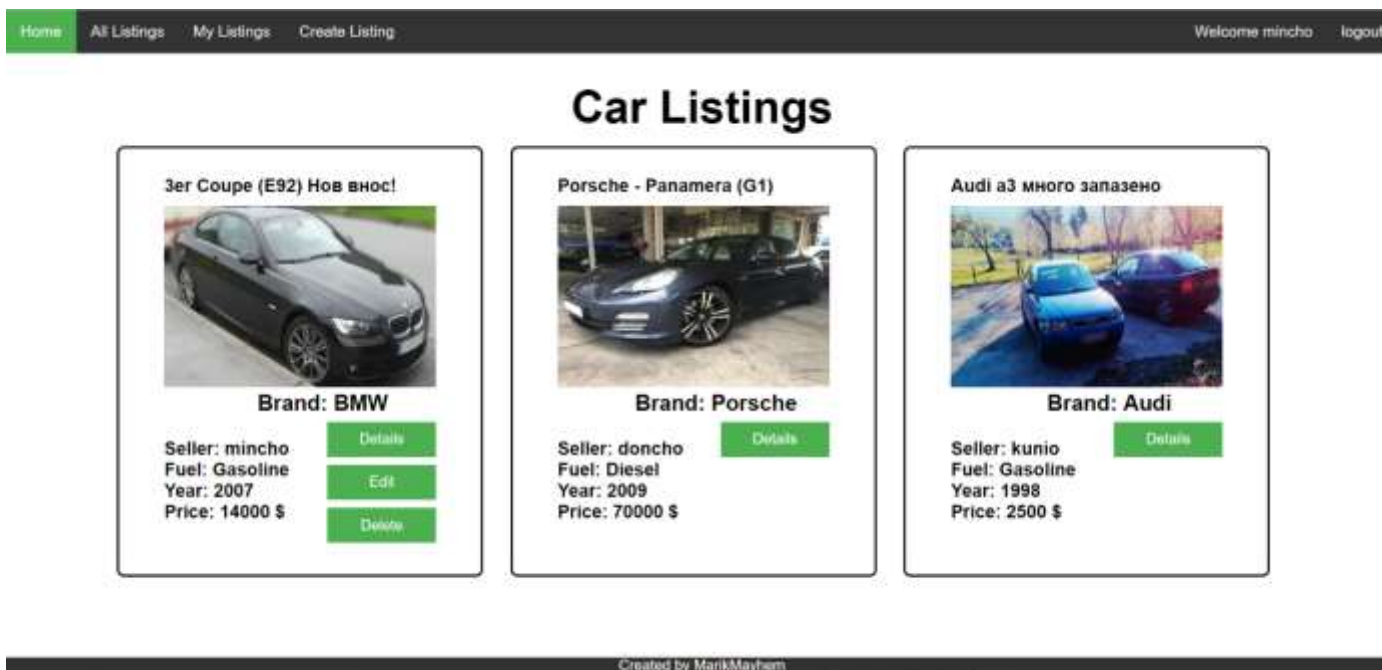
Successfully logged in user should be able to **logout** from the app.

- After a **successful** logout, a **notification** message “Logout successful.” should be displayed.
- After successful logout, the **Sign In screen** should be shown.
- The “**logout**” **REST service** at the back-end should be obligatory called at logout.
- All local information in the browser (**user session data**) about the current user should be deleted.

Logged In Home Screen (15 pts)

Successfully logged users should be welcomed by the **catalog home** screen which can be accessed **after** clicking the **All Listings** link in the **NavBar**. The catalog contains **all** listings on the site.

- The **listings** should be listed in the **format** as shown in the Web design (see the screenshot below).
- Each **post** has **title, image, brand, seller, fuel, year, price**.
- When the **details/edit** link is clicked, your app should **redirect** to the details/edit **section** of the listing.
- Listing **authors** can edit/delete their **own** listings.
- In case of **error** (e.g. Internet connection lost), an error message should be displayed.
- In case of **no listings**, display ‘No listings in database’.



Create Listing Screen (10 pts)

Logged in users can **create** listings. Clicking the **[Create Listing]** link should open a form.

- After a **successful** listing creation, a notification message “listing created.” should be displayed and the **home catalog** should be shown.
- The **title** length must not exceed 33 characters!
- The **description** length must not exceed 450 characters and should be at least 30!
- The **brand,fuelType and model** length must not exceed 11 characters!
- The **model** length should be at least 4 characters!
- The **year** must be only 4 chars long!
- The maximum price is 1000000\$
- Link url should always start with “**http**”.
- In case of **empty/incorrect** input fields an appropriate **error** message should be **displayed** and the user should be able to fill the **form** again.
- Ensure you handle properly all HTML **special characters**, e.g. the description text could hold "Hi, <peter>".

[Home](#)
[All Listings](#)
[My Listings](#)
[Create Listing](#)
Welcome mincho [logout](#)

Create Car Listing

Please fill in this form to create an listing.

Title

Enter Title

Description

Enter Description

Car Brand

Enter Car Brand

Car Model

Enter Car Model

Car Year

Enter Car Year

Car Image

Enter Car Image

Car Fuel Type

Enter Car Fuel Type

Car Price

Enter Car Price

Create Listing

Created by MarkMayhem

Edit Listing Screen (15 pts)

Sellers **should** be able to **edit** their own **listings**. Clicking the [**Edit**] link on each listing should open an edit **form**. Inside the edit form all input fields should be **filled** with the **current** data of each post.

- After a **successful** listing update, a notification message “Listing {listingTitle} updated.” should be displayed and the **home catalog** should be shown.
- **The validation is same as create listing!**
- Ensure you handle properly all HTML **special characters**, e.g. the description text could hold "Hi, <peter>".

[Home](#)
[All Listings](#)
[My Listings](#)
[Create Listing](#)

Welcome mincho [logout](#)

Edit Car Listing

Please fill in this form to edit an listing.

Title

3er Coupe (E82) Hot mod.

Description

Hot mod, mozbaz na e-Suncana razred! Krasna okolijska!

Car Brand

BMW

Car Model

E82

Car Year

2007

Car Image

https://images.cdn.carfax.co/image/2/1200/1305/uploads/topics/bmw_2050d_e82_m-paket_bmw-56e130d02we64.jpg

Car Fuel Type

Gasoline

Car Price

14000 \$

Edit Listing

Delete Listing (5 pts)

Sellers **should** be able to **delete** their **own** listing by clicking the **[Delete]** button.

- After **successful** listing delete a notification message "Listing deleted." should be displayed and the **home catalog** should be **shown**.
- In case of **error** (e.g. Internet connection lost / unauthorized request / missing message), an error message should be displayed.
- Deleting works immediately, with **no confirmation**.


My Listings (10 pts)

Each **user** should be able to view his own listings by clicking **[My Listings]**.

- The **listings** should be listed like in the **home catalog** section.
- In case of **error** (e.g. Internet connection lost), an error message should be displayed.
- In case of **no listings**, display 'No listings in database'.

My car listings

Audi a3 много запазено



Brand:	Model:	Year:	Price:
Audi	A3	1998	2500

[Details](#)
[Edit](#)
[Delete](#)

Created by MarkMayhem

Listing Details (10 pts)

Each **listing** has a details view. By clicking the **[details]** link **partial** information about each post should be listed!

- In case of **error** (e.g. Internet connection lost), an error message should be displayed.
- Ensure you handle properly all HTML **special characters**, e.g. the description text could hold "*Hi, <peter>*".

Audi a3 много запазено



Brand: Audi Model: A3 Year: 1998 Fuel: Gasoline Price: 2500\$

[Edit](#)
[Delete](#)

Description:

Не съм навъртал на колата 100000км в никакъв случай. Като нова е

Created by MarkMayhem

Problem 5. Submitting Your Solution

Place in a ZIP file your project folder. Exclude the **node_modules** folder. Upload the archive to the Judge.