



**ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ**

**UNIVERSITY OF PIRAEUS**

**ΤΜΗΜΑ: ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ  
"ΠΡΟΗΓΜΕΝΑ ΣΥΣΤΗΜΑΤΑ ΠΛΗΡΟΦΟΡΙΚΗΣ"**

**ΠΜΣ : ΑΝΑΠΤΥΞΗ ΛΟΓΙΣΜΙΚΟΥ ΚΑΙ ΤΕΧΝΗΤΗΣ  
ΝΟΗΜΟΣΥΝΗΣ**

**ΜΑΘΗΜΑ: ΠΡΟΗΓΜΕΝΟΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ  
FULLSTACK JAVASCRIPT FRAMEWORKS**

**ΤΕΛΙΚΗ ΑΤΟΜΙΚΗ ΕΡΓΑΣΙΑ**

**ΟΝΟΜΑΤΕΠΩΝΥΜΟ: ΚΑΨΗΣ ΣΤΥΛΙΑΝΟΣ ΜΙΧΑΗΛ**

**Α.Μ. : ΜΠΣΠ24018**

# Smart Tourism – Documentation

## Εισαγωγή

Το **Smart Tourism** είναι μια web εφαρμογή που επιτρέπει τη διαχείριση και προβολή δεδομένων για διάφορες χώρες, βασισμένα σε κριτήρια όπως **Quality of Life, Adventure, Heritage, Cost of Living Index, Restaurant Price Index**.

Η εφαρμογή αποτελείται από **backend** σε **Node.js/Express** με βάση δεδομένων **MongoDB**, και **frontend** σε **React** με **TailwindCSS**.

## Οδηγίες Εκτέλεσης

Οι οδηγίες που θα δωθούν εφαρμόστηκαν στο Visual Studio Code IDE.

### Εκκίνηση Backend

1. Άνοιξε το τερματικό στο φάκελο **Smart Tourism** και πήγαινε στον φάκελο **backend**:

**cd backend**

2. Εγκατάσταση όλων των εξαρτήσεων:

**npm install**

3. Δημιουργία και ρύθμιση αρχείου **.env** (περιγράφεται παρακάτω).

4. Εκκίνηση του backend server:

**npm start**

Ο **server** θα εκκινήσει στο **http://localhost:5000/**.

### Εκκίνηση Frontend

1. Άνοιξε νέο τερματικό στο φάκελο **Smart Tourism** και πήγαινε στον φάκελο **frontend**:

**cd frontend**

2. Εγκατάσταση όλων των εξαρτήσεων:

**yarn install ή npm install**

3. Εκκίνηση του frontend:

**yarn start ή npm start**

Ο **frontend** θα εκκινήσει στο **http://localhost:3000/**.

## ΠΡΟΣΟΧΗ

### Ρύθμιση .env (Backend)

Δημιούργησε το αρχείο **.env** μέσα στον φάκελο **backend** (αν δεν υπάρχει ήδη) και πρόσθεσε τα παρακάτω:

1. **MONGO\_URI=your\_mongodb\_connection\_string**

2. **PORT=5000**

Αντικατάστησε το «**your\_mongodb\_connection\_string**» με το πραγματικό **connection string** της βάσης δεδομένων **MongoDB**.

# Περιγραφή Backend

## Δομή φακέλων (Backend)

```
backend/
├── config/
│   └── db.js
├── controllers/
│   └── countryController.js
├── data/
│   └── Merged_Country_Data.json
├── models/
│   └── country.js
├── routes/
│   └── countryRoutes.js
├── server.js
├── .env
├── package.json
├── package-lock.json
└── importData.js
```

## Περιγραφή αρχείων

- **db.js** → Σύνδεση με MongoDB
- **countryController.js** → Περιέχει τις CRUD λειτουργίες του API
- **Merged\_Country\_Data.json** → Περιλαμβάνει δεδομένα για διάφορες χώρες, με διαφορετικούς δείκτες που σχετίζονται με την ποιότητα ζωής, την περιπέτεια, την κληρονομιά, το κόστος ζωής και το κόστος εστίασης.
- **country.js** → Ορισμός του MongoDB Schema
- **countryRoutes.js** → Διαχείριση των routes του API
- **server.js** → Εκκίνηση του Express server
- **.env** → Σύνδεση στη βάση δεδομένων MongoDB
- **importData.js** → Εισαγωγή δεδομένων από ένα αρχείο JSON στη βάση δεδομένων MongoDB
- **package.json** → Περιέχει τις πληροφορίες για την εφαρμογή, όπως τις εξαρτήσεις (dependencies), τις εντολές για την εκκίνηση της εφαρμογής, καθώς και τις γενικές πληροφορίες για το project.

## Λειτουργίες API (Backend)

Μέθοδος	Endpoint	Περιγραφή
GET	/api/v1/countries	Επιστρέφει όλες τις χώρες
GET	/api/v1/countries/:id	Επιστρέφει χώρα με βάση το ID
GET	/api/v1/countries/search/:name	Επιστρέφει χώρα με βάση το όνομα

Μέθοδος	Endpoint	Περιγραφή
POST	/api/v1/countries	Προσθήκη νέας χώρας
PUT	/api/v1/countries/:id	Ενημέρωση χώρας
DELETE	/api/v1/countries/:id	Διαγραφή χώρας
GET	/api/v1/countries/filter?criterion=Adventure&type=highest&limit=5	Φιλτράρει τις χώρες με βάση κάποιο κριτήριο (στην συγκεκριμένη περίπτωση είναι η Περιπέτεια), ταξινομεί τις χώρες σε αύξουσα ή φθίνουσα σειρά και εμφανίζει τόσες χώρες όσο έχουμε ορίσει (π.χ. limit=5)

## Περιγραφή Frontend

### Δομή φακέλων (Frontend)

```

frontend/
├── src/
│   ├── AddCountry/
│   │   ├── AddCountry.js
│   │   └── AddCountry.css
│   ├── CountryList /
│   │   ├── CountryList.js
│   │   └── CountryList.css
│   ├── CountrySearch /
│   │   └── CountrySearch.js
│   ├── EditCountry /
│   │   └── EditCountry.js
│   ├── NotFound.js
│   ├── App.js
│   ├── App.css
│   ├── index.js
│   └── index.css
└── package.json

```

### Περιγραφή αρχείων

- **CountryList.js** → Το αρχείο είναι υπεύθυνο για την εμφάνιση και τη διαχείριση της λίστας των χωρών της εφαρμογής. Εδώ οι χρήστες μπορούν να δουν τις χώρες, να τις φιλτράρουν, να τις ταξινομήσουν, να αλλάξουν τις πληροφορίες τους και να τις διαγράψουν. Πατώντας κάποιο από τα κουμπιά των λειτουργιών «προσθήκη χώρας» και «αλλαγή πληροφοριών χώρας», ο χρήστης θα μεταφερθεί σε νέα σελίδα, για την ολοκλήρωση της ενέργειας του.
- **EditCountry.js** → Σελίδα επεξεργασίας χώρας. Επιτρέπει στους χρήστες να επεξεργαστούν τα δεδομένα μιας χώρας που ήδη υπάρχει στο σύστημα. Οι χρήστες μπορούν να τροποποιήσουν τις πληροφορίες μιας χώρας και να τις αποθηκεύσουν πίσω στο **backend**.
- **AddCountry.js** → Σελίδα προσθήκης χώρας. Περιέχει τη λειτουργία που επιτρέπει στους χρήστες να προσθέσουν μια νέα χώρα στην εφαρμογή, και να την αποθηκεύσουν πίσω στο

## backend.

- **CountrySearch.js** → Αρχείο υπεύθυνο για την αναζήτηση μιας χώρας από το χρήστη, μέσω της ονομασίας της. Οι χρήστες μπορούν να εισάγουν το όνομα μιας χώρας και να δουν τα σχετικά αποτελέσματα από το API.
- **NotFound.js** → Το αρχείο είναι ένα απλό React component που χρησιμοποιείται για να εμφανίσει ένα μήνυμα όταν ο χρήστης προσπαθεί να επισκεφτεί μια σελίδα που δεν υπάρχει (404 Error Page).
- **App.js** → Το αρχείο είναι το κεντρικό component της εφαρμογής και περιλαμβάνει τη διαχείριση των routes για την πλοήγηση μεταξύ των διαφορετικών σελίδων της.
- **index.js** → Το αρχείο είναι το σημείο εκκίνησης για την εφαρμογή React και είναι υπεύθυνο για τη δημιουργία του root στοιχείου της εφαρμογής και την απόδοση της εφαρμογής στον DOM.

## Οδηγίες Χρήσης API με Postman

Για να κάνεις δοκιμές στο API μέσω **Postman**:

1. **Άνοιξε το Postman** και κάνε **Import** το αρχείο **Postman Collection**.
2. **Χρησιμοποίησε τα παρακάτω requests:**
  - **GET** `http://localhost:5000/api/v1/countries` (**getAllCountries**)
  - **GET** `http://localhost:5000/api/v1/countries/search/` (**getCountryByName**) (π.χ. `http://localhost:5000/api/v1/countries/search/Greece`)
  - **GET** `http://localhost:5000/api/v1/countries/:id` (**getCountryById**) (π.χ. `http://localhost:5000/api/v1/countries/67adf338e29cb8dbdfbc9c89`)
  - **GET** `http://localhost:5000/api/v1/countries/filter? criterion=... &type=... &limit=...` (**filterCountries**) (π.χ. `http://localhost:5000/api/v1/countries/filter?criterion=Adventure&type=highest&limit=5`)
  - **POST** `http://localhost:5000/api/v1/countries` (**createCountry**) (Διευκρίνιση: Εδώ ο χρήστης χρειάζεται να δώσει τα στοιχεία της χώρας που θέλει να δημιουργήσει. Π.χ. 

```
{
  "Country": "FakeCountry",
  "Quality_Of_Life": 85.6,
  "Adventure": 78.2,
  "Heritage": 90.3,
  "Cost_of_Living_Index": 70.1,
  "Restaurant_Price_Index": 65.4
}
```

)
  - **PUT** `http://localhost:5000/api/v1/countries/:id` (**updateCountry**) (π.χ. `http://localhost:5000/api/v1/countries/67adf338e29cb8dbdfbc9c89`). Διευκρίνιση: Εδώ ο χρήστης χρειάζεται να δώσει και τα νέα στοιχεία, πριν αλλάξει της χώρας που επιθυμεί. Π.χ. 

```
{
  "Country": "Testing",
  "Quality_of_Life": 5,
  "Adventure": 3,
  "Heritage": 2,
  "Cost_of_Living_Index": 1,
  "Restaurant_Price_Index": 0
}
```

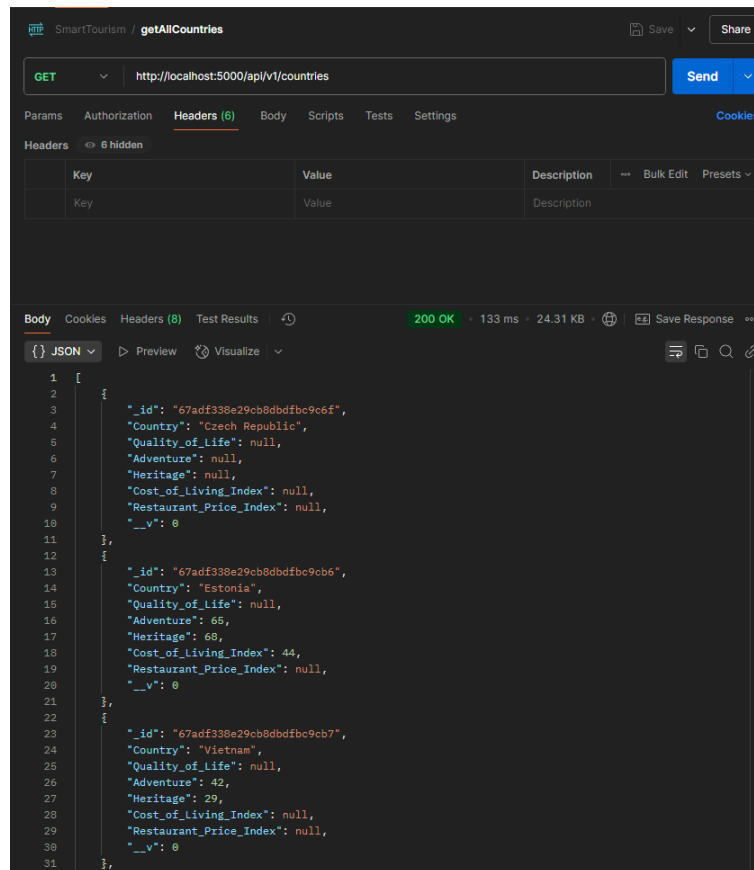
, αν θέλει να αλλάξει πολλά στοιχεία ταυτόχρονα ,

```
{
  "Heritage": 2
}, αν θέλει να αλλάξει ένα στοιχείο μόνο.
```

- **DELETE** <http://localhost:5000/api/v1/countries/:id> (**deleteCountry**) (π.χ. <http://localhost:5000/api/v1/countries/67adf338e29cb8dbdfbc9c89>)

Παρακάτω φαίνονται παραδείγματα με φωτογραφίες, μέσω του Postman:

## getAllCountries



## getCountryByName

SmartTourism / getCountryByName

GET

http://localhost:5000/api/v1/countries/search/Testing

Send

Params

Authorization

Headers (6)

Body

Scripts

Tests

Settings

Cookies

Query Params

	Key	Value	Description	...	Bulk Edit
	Key	Value	Description		

Body

Cookies

Headers (8)

Test Results

200 OK

47 ms

429 B

Save Response

{ } JSON

Preview

Visualize

```
1 {
2   "_id": "67c86d98b8344ccfdf23798f",
3   "Country": "Testing",
4   "Quality_of_Life": 5,
5   "Adventure": 3,
6   "Heritage": 2,
7   "Cost_of_Living_Index": 1,
8   "Restaurant_Price_Index": 0,
9   "__v": 0
10 }
```

## getCountryById

SmartTourism / getCountryById

GET

http://localhost:5000/api/v1/countries/67adf338e29cb8dbdfbc9c89

Send

Params

Authorization

Headers (6)

Body

Scripts

Tests

Settings

Cookies

Query Params

	Key	Value	Description	...	Bulk Edit
	Key	Value	Description		

Body

Cookies

Headers (8)

Test Results

200 OK

46 ms

437 B

Save Response

{ } JSON

Preview

Visualize

```
1 {
2   "_id": "67adf338e29cb8dbdfbc9c89",
3   "Country": "Greece",
4   "Quality_of_Life": null,
5   "Adventure": 3,
6   "Heritage": 3,
7   "Cost_of_Living_Index": null,
8   "Restaurant_Price_Index": null,
9   "__v": 0
10 }
```

filterCountries

GET

http://localhost:5000/api/v1/countries/filter?criterion=Adventure&type=highest&limit=5

Send

Params

Authorization

Headers (6)

Body

Scripts

Tests

Settings

Cookies

Query Params

<input checked="" type="checkbox"/>	Key	Value	Description	Bulk Edit
<input checked="" type="checkbox"/>	criterion	Adventure		
<input checked="" type="checkbox"/>	type	highest		
<input checked="" type="checkbox"/>	limit	5		

Body

Cookies

Headers (8)

Test Results

200 OK

53 ms

1.11 KB

Save Response

JSON

Preview

Visualize

```
1  [
2    {
3      "_id": "67adf338e29cb8dbdfbc9c7d",
4      "Country": "Ukraine",
5      "Quality_of_Life": null,
6      "Adventure": 86,
7      "Heritage": 81,
8      "Cost_of_Living_Index": null,
9      "Restaurant_Price_Index": null,
10     "__v": 0
11   },
12   {
13     "_id": "67adf338e29cb8dbdfbc9c6c",
14     "Country": "Belarus",
15     "Quality_of_Life": null,
16     "Adventure": 84,
17     "Heritage": 87,
18     "Cost_of_Living_Index": null,
19     "Restaurant_Price_Index": null,
20     "__v": 0
21   },
22   {
23     "_id": "67adf338e29cb8dbdfbc9cde",
24     "Country": "Uzbekistan",
25     "Quality_of_Life": null,
26     "Adventure": 83,
27     "Heritage": 86,
28     "Cost_of_Living_Index": null,
29     "Restaurant_Price_Index": null,
30     "__v": 0
31   }
],
```

GET

http://localhost:5000/api/v1/countries/filter?criterion=Heritage&type=lowest&limit=5

Send

Params

Authorization

Headers (6)

Body

Scripts

Tests

Settings

Cookies

Query Params

<input checked="" type="checkbox"/>	Key	Value	Description	Bulk Edit
<input checked="" type="checkbox"/>	criterion	Heritage		
<input checked="" type="checkbox"/>	type	lowest		
<input checked="" type="checkbox"/>	limit	5		

Body

Cookies

Headers (8)

Test Results

200 OK

47 ms

1.16 KB

Save Response

JSON

Preview

Visualize

```
1  [
2    {
3      "_id": "67adf338e29cb8dbdfbc9cbc",
4      "Country": "Tanzania",
5      "Quality_of_Life": null,
6      "Adventure": null,
7      "Heritage": null,
8      "Cost_of_Living_Index": null,
9      "Restaurant_Price_Index": null,
10     "__v": 0
11   },
12   {
13     "_id": "67adf338e29cb8dbdfbc9c6f",
14     "Country": "Czech Republic",
15     "Quality_of_Life": null,
16     "Adventure": null,
17     "Heritage": null,
18     "Cost_of_Living_Index": null,
19     "Restaurant_Price_Index": null,
20     "__v": 0
21   },
22   {
23     "_id": "67adf338e29cb8dbdfbc9c85",
24     "Country": "North Macedonia",
25     "Quality_of_Life": null,
26     "Adventure": null,
27     "Heritage": null,
28     "Cost_of_Living_Index": null,
29     "Restaurant_Price_Index": null,
30     "__v": 0
31   }
],
```



GET  Send

Params • Authorization Headers (6) Body Scripts Tests Settings Cookies

Query Params

<input checked="" type="checkbox"/>	Key	Value	Description	⋮ Bulk Edit
<input checked="" type="checkbox"/>	criterion	Cost_of_Living_Index		🗑
<input checked="" type="checkbox"/>	type	highest		
<input checked="" type="checkbox"/>	limit	2		
	Key	Value	Description	

Body Cookies Headers (8) Test Results 200 OK • 46 ms • 614 B • Save Response

{ } JSON Preview Visualize

```
1  [
2    {
3      "_id": "67c8a2077031c2aab3dcb9d0",
4      "Country": "789",
5      "Quality_of_Life": null,
6      "Adventure": 78.2,
7      "Heritage": 90.3,
8      "Cost_of_Living_Index": 70.1,
9      "Restaurant_Price_Index": 65.4,
10     "_v": 0
11   },
12   {
13     "_id": "67adf338e29cb8dbdfbc9cb6",
14     "Country": "Estonia",
15     "Quality_of_Life": null,
16     "Adventure": 65,
17     "Heritage": 68,
18     "Cost_of_Living_Index": 44,
19     "Restaurant_Price_Index": null,
20     "_v": 0
21   }
22 ]
```

## createCountry

POST  Send

Params Authorization Headers (8) Body • Scripts Tests Settings Cookies

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON Beautify

```
1  {
2    "Country": "FakeCountry",
3    "Quality_of_Life": 85.6,
4    "Adventure": 78.2,
5    "Heritage": 90.3,
6    "Cost_of_Living_Index": 70.1,
7    "Restaurant_Price_Index": 65.4
8  }
9
```

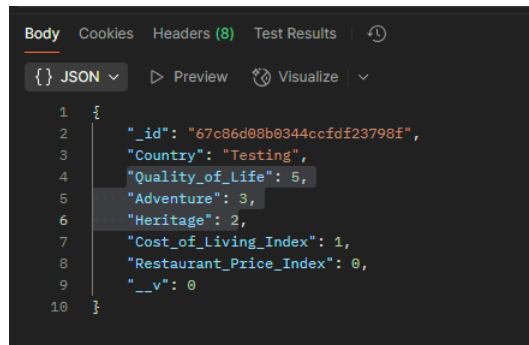
Body Cookies Headers (8) Test Results 201 Created • 109 ms • 453 B • Save Response

{ } JSON Preview Visualize

```
1  {
2    "Country": "FakeCountry",
3    "Quality_of_Life": null,
4    "Adventure": 78.2,
5    "Heritage": 90.3,
6    "Cost_of_Living_Index": 70.1,
7    "Restaurant_Price_Index": 65.4,
8    "_id": "67d8346977959f83c6f7fedf",
9    "_v": 0
10 }
```

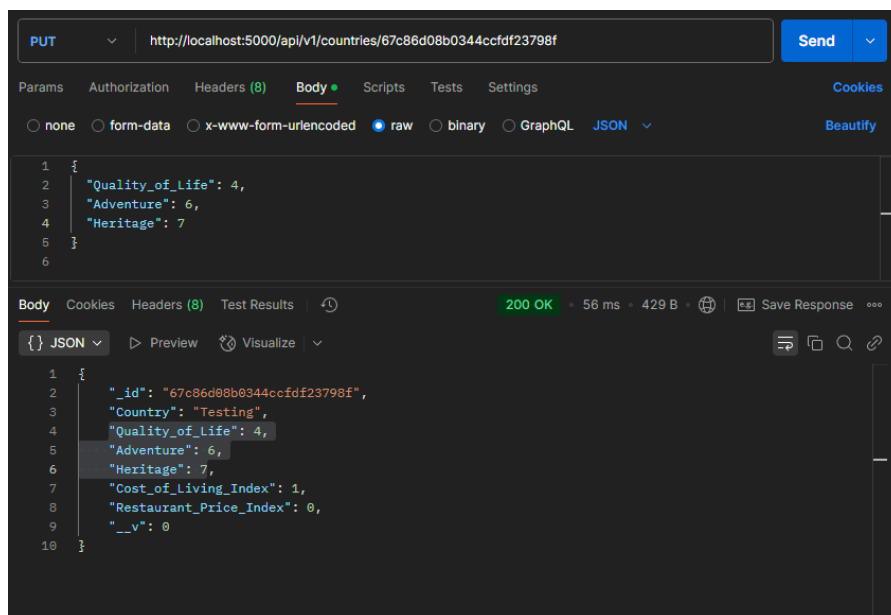
## updateCountry

Στοιχεία χώρας πριν:



```
1 {
2   "_id": "67c86d08b0344ccdf23798f",
3   "Country": "Testing",
4   "Quality_of_Life": 5,
5   "Adventure": 3,
6   "Heritage": 2,
7   "Cost_of_Living_Index": 1,
8   "Restaurant_Price_Index": 0,
9   "__v": 0
10 }
```

Στοιχεία χώρας μετά:



PUT http://localhost:5000/api/v1/countries/67c86d08b0344ccdf23798f

Body (raw):

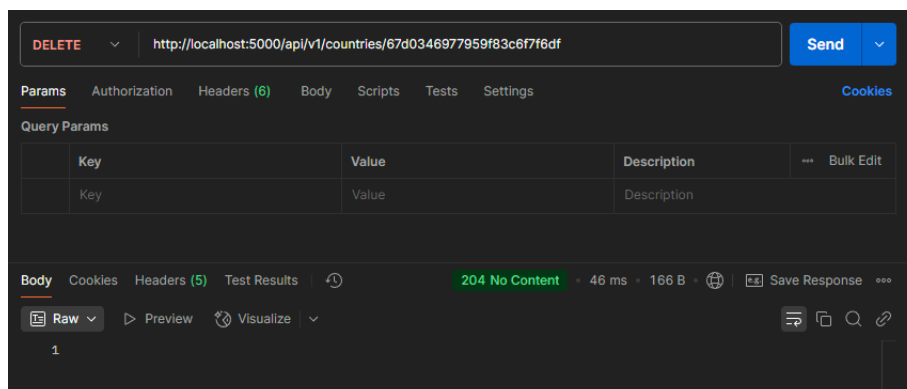
```
1 {
2   "Quality_of_Life": 4,
3   "Adventure": 6,
4   "Heritage": 7
5 }
```

Response: 200 OK - 56 ms - 429 B

Body (JSON):

```
1 {
2   "_id": "67c86d08b0344ccdf23798f",
3   "Country": "Testing",
4   "Quality_of_Life": 4,
5   "Adventure": 6,
6   "Heritage": 7,
7   "Cost_of_Living_Index": 1,
8   "Restaurant_Price_Index": 0,
9   "__v": 0
10 }
```

## deleteCountry



DELETE http://localhost:5000/api/v1/countries/67d0346977959f83c6f7f6df

Query Params:

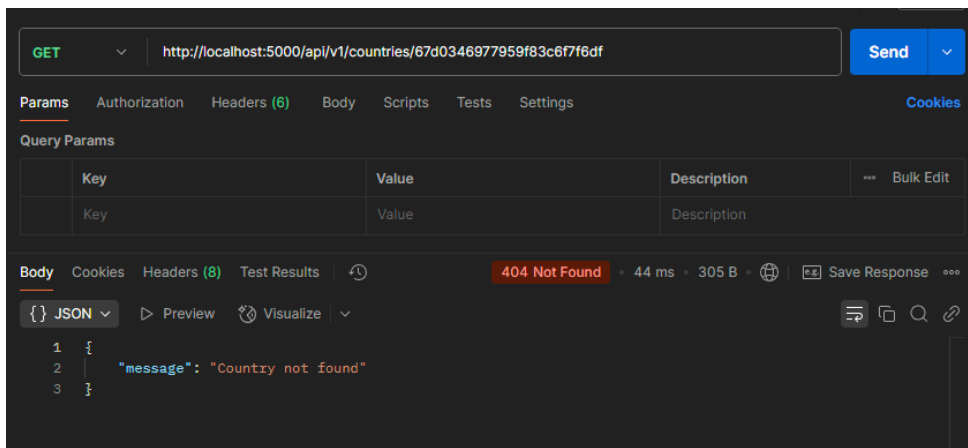
Key	Value	Description
Key	Value	Description

Response: 204 No Content - 46 ms - 166 B

Body (Raw):

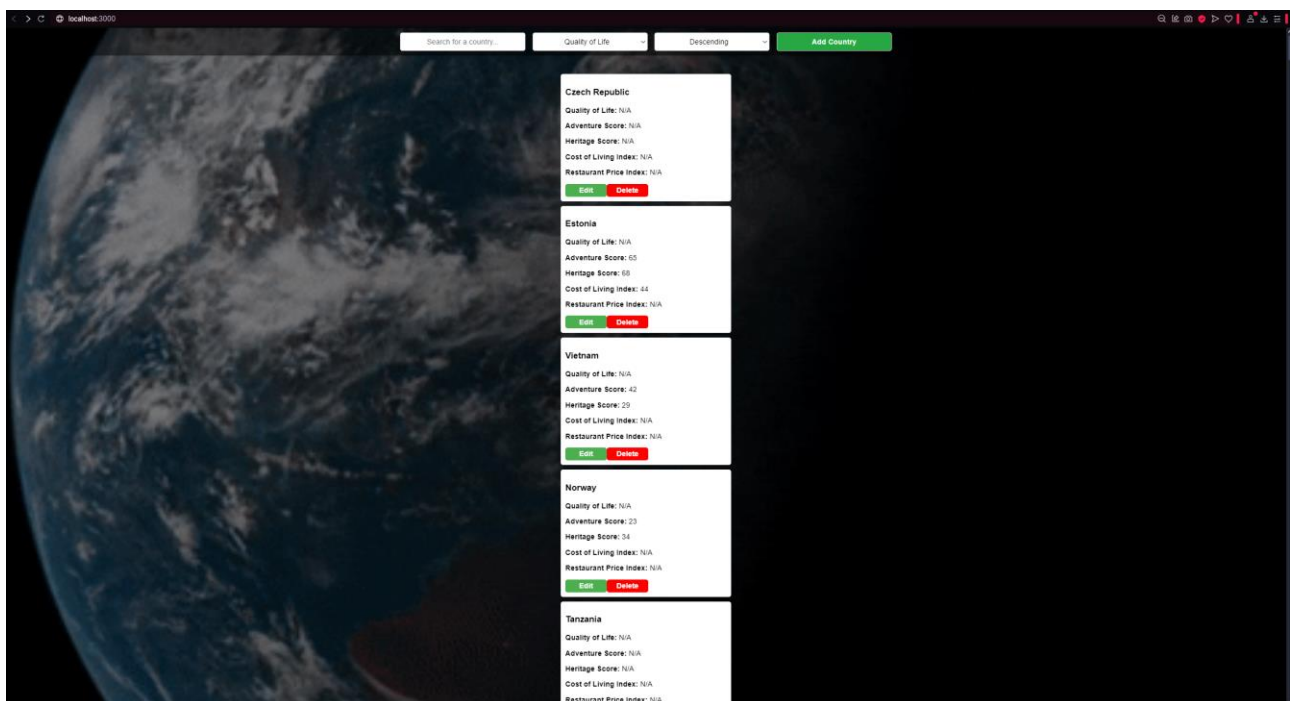
```
1
```

Εδώ δεν φαίνεται αμέσως η διαγραφή της χώρας με το id, που επέλεξε ο χρήστης. Βέβαια, αν πάμε στα **getCountryById** ή **getCountryByName** request, και ψάξουμε τη χώρα που διαγράφηκε, θα εμφανιστεί το κατάλληλο μήνυμα.

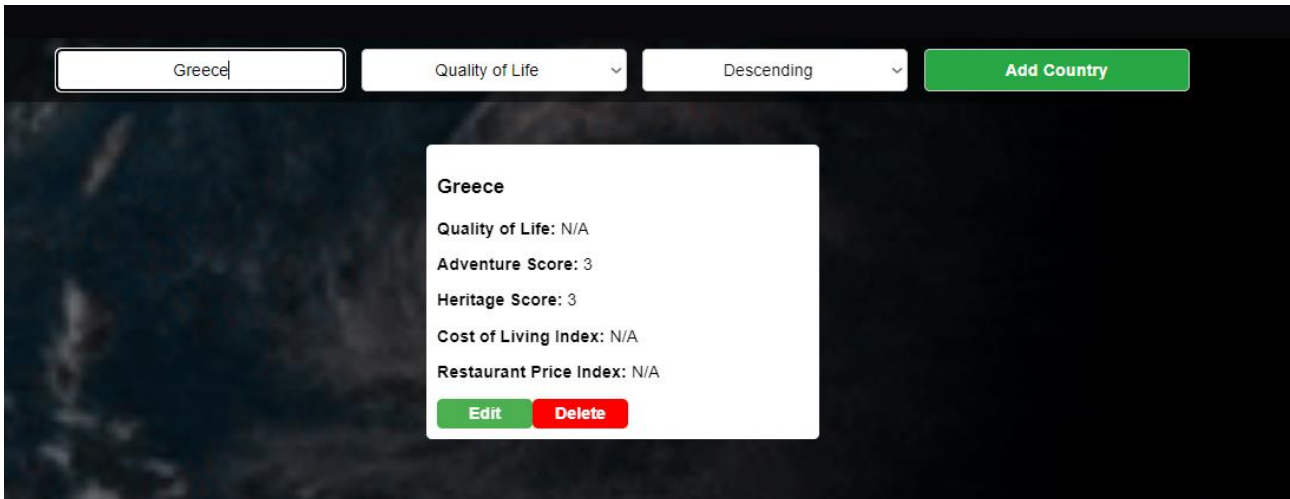


Παρακάτω παρατίθενται φωτογραφίες από το Frontend (UI).

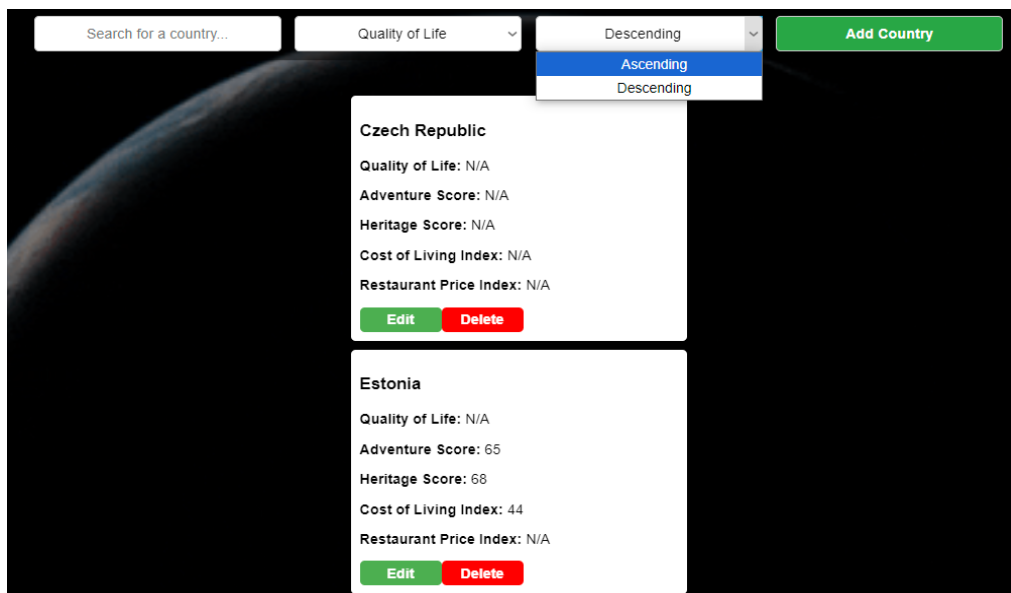
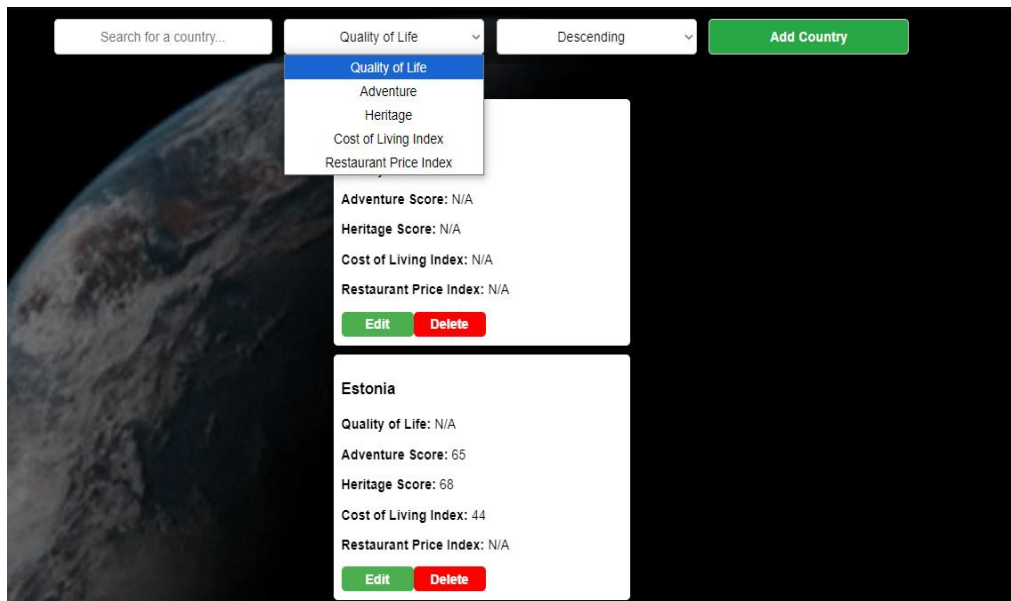
Σελίδα: CountryList.js / .css



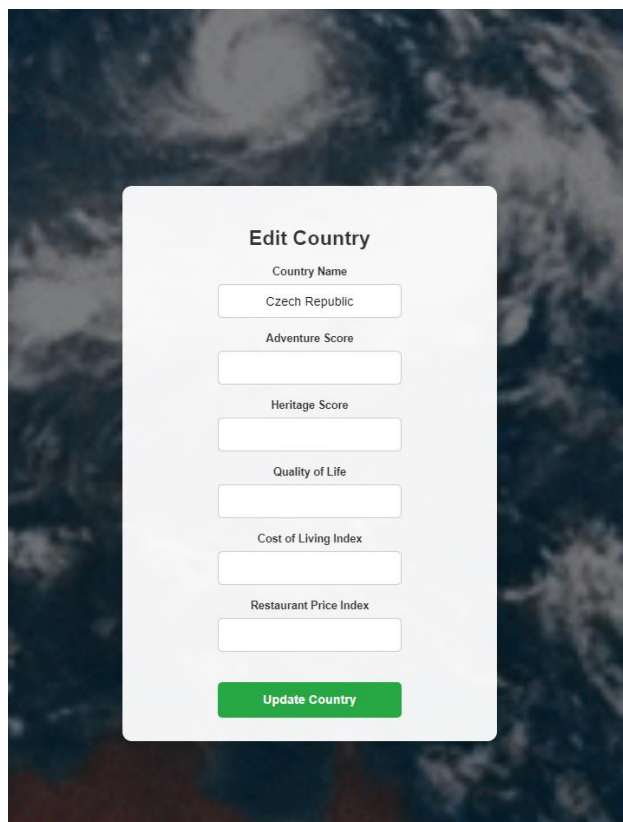
Σελίδα: CountrySearch.js, CountryList.js / .css , εύρεση χώρας με όνομα.



Σελίδα: CountryList.js / .css , ταξινόμηση χωρών με επιλογή φίλτρου.



Σελίδα: EditCountry.js / .css



**Edit Country**

Country Name  
Czech Republic

Adventure Score

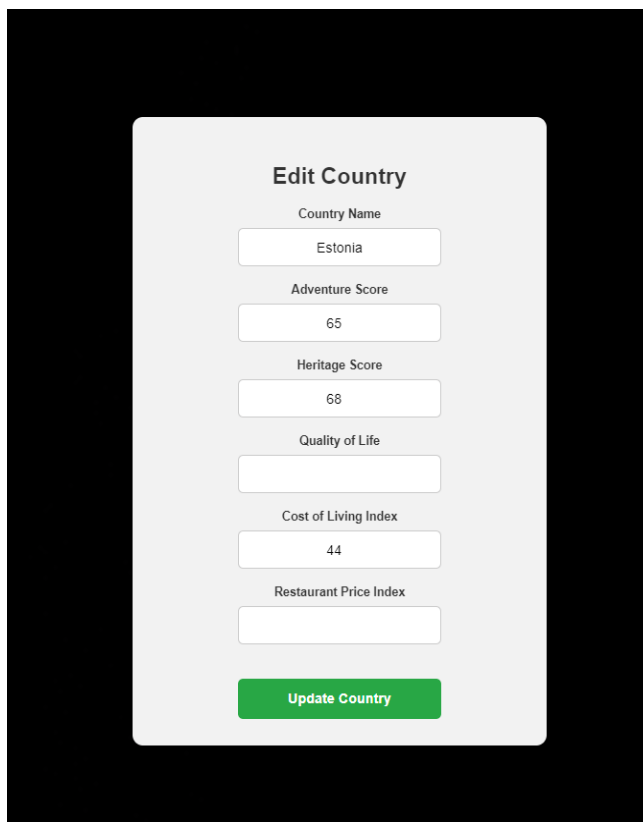
Heritage Score

Quality of Life

Cost of Living Index

Restaurant Price Index

**Update Country**



**Edit Country**

Country Name  
Estonia

Adventure Score  
65

Heritage Score  
68

Quality of Life

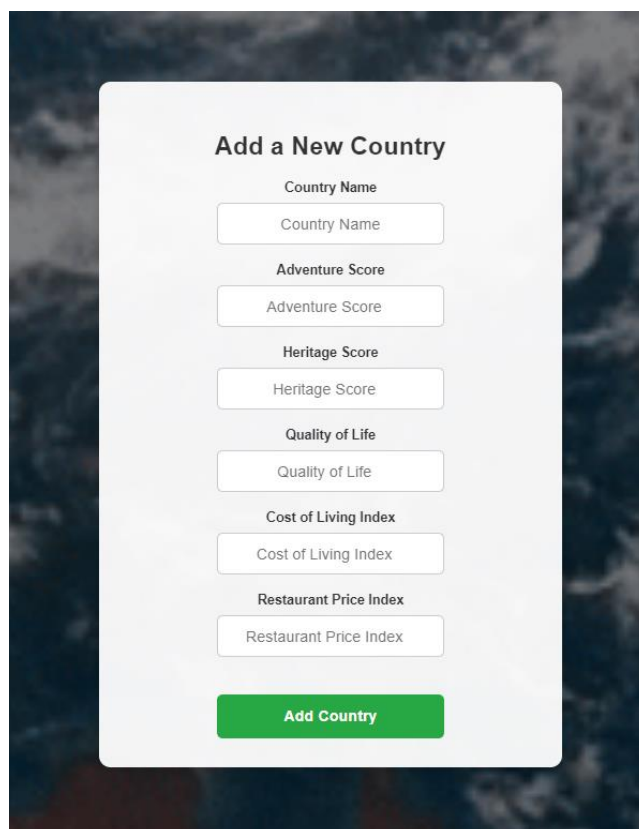
Cost of Living Index  
44

Restaurant Price Index

**Update Country**

Σημείωση: Τα πεδία που φαίνονται κενά έχουν null τιμές.

Σελίδα: EditCountry.js / .css



**Add a New Country**

Country Name

Adventure Score

Heritage Score

Quality of Life

Cost of Living Index

Restaurant Price Index

**Add Country**

ΤΕΛΟΣ ΕΡΓΑΣΙΑΣ