



**ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ**

**ΜΑΘΗΜΑ: ΥΠΗΡΕΣΙΟΣΤΡΕΦΕΣ\_ΛΟΓΙΣΜΙΚΟ**  
**ΕΡΓΑΣΙΑ ΕΞΑΜΗΝΟΥ 2022-23**  
**ΦΟΙΤΗΤΡΙΑ: ΜΠΑΝΤΑΝΑ ΔΑΝΑΗ-ΙΩΑΝΝΑ**  
**ΑΜ: 17081**

**“ΕΓΧΕΙΡΙΔΙΟ ΧΡΗΣΗΣ”**

## Table of Contents

|  |    |
|--|----|
| Εισαγωγή.....                              | 3  |
| Ενέργειες.....                             | 3  |
| API Κλήσεις Εφαρμογής.....                 | 4  |
| 1. Signup.....                             | 4  |
| 2. Login.....                              | 5  |
| 3. Logout.....                             | 5  |
| 4. Todos.....                              | 6  |
| 1. Get all todos.....                      | 6  |
| 2. Get a todo.....                         | 6  |
| 3. Create a new todo.....                  | 6  |
| 4. Update a todo.....                      | 7  |
| 5. Delete a todo and its items.....        | 7  |
| 5. Todo Items.....                         | 9  |
| 1. Get all todo items of a todos list..... | 9  |
| 2. Get a todo item.....                    | 9  |
| 3. Create a new todo item.....             | 10 |
| 4. Update a todo item.....                 | 10 |
| 5. Delete a todo item.....                 | 11 |

# Εισαγωγή

## Τίτλος: Todos API

Το ‘ Todos API’ είναι ένα REST API με χρήση RAILS με τις παρακάτω λειτουργίες για μία λίστα εκκρεμοτήτων (todo) η οποία έχει λεπτομέρειες για κάθε todo (list of todo items).

Η εφαρμογή καλύπτει ένα εύρος δυνατοτήτων. Ο χρήστης έχει την δυνατότητα να φτιάξει λογαριασμό και να κάνει είσοδο και έξοδο στην εφαρμογή. Μπορεί να δει, δημιουργήσει, ανανεώσει και διαγράψει todo λίστες με τις αντίστοιχες λειτουργίες για τα αντικείμενα τους (todo items).

Είναι απλό στην χρήση και απαιτείται από την χρήστη να γνωρίζει λίγο το περιβάλλον ενός τερματικού (terminal) και να έχει εγκατεστημένο το πακέτο ‘curl’.

## Ενέργειες

Παρακάτω φαίνονται αναλυτικά οι ενέργειες που μπορεί να πραγματοποιήσει ένας χρήστης στην εφαρμογή και παρακάτω θα αναλυθούν πως μπορεί να αξιοποιήσει την κάθε ενέργεια.

| Endpoint                    | Functionality                 |
|-----------------------------|-------------------------------|
| POST /signup                | Signup                        |
| POST /auth/login            | Login                         |
| GET /auth/logout            | Logout                        |
| GET /todos                  | List all todos and todo items |
| POST /todos                 | Create a new todo             |
| GET /todos/:id              | Get a todo                    |
| PUT /todos/:id              | Update a todo                 |
| DELETE /todos/:id           | Delete a todo and its items   |
| GET /todos/:id/items/:id    | Get a todo item               |
| POST /todos/:id/items       | Create a new todo item        |
| PUT /todos/:id/items/:id    | Update a todo item            |
| DELETE /todos/:id/items/:id | Delete a todo item            |

# API Κλήσεις Εφαρμογής

## 1. Signup

Εγγραφή στην εφαρμογή.

Εντολή: **POST**

```
curl -X POST http://localhost:3000/signup -H "Content-Type: application/json" -d '{"name": "<name>", "email": "<email>", "password": "<password>"}'
```

Παραμέτρους: Όχι

Request Body: Application json (required)

```
{
  "name": "example1",
  "email": "example@gmail.com",
  "password": "12345"
}
```

Responses:

| Code              | Description  |
|-------------------|--|
| 200               | {"message": "Account created successfully", "auth_token": "eyJhbGciOiJIUzI1NiJ9.eyJ1c2VyX2lkIjoxLCJleHAiOjE2Nzc1MDg4NzR9.yBesbdCx5w7DrSoLKysEXMEMDzyCLFewgnX0GVW7gHo"} |
| 400 – Bad Request | Missing required values in request body  |

## 2. Login

Είσοδος στην εφαρμογή.

Εντολή: **POST**

```
curl -X POST http://localhost:3000/auth/login -H "Content-Type: application/json" -d '{"email": "<email>", "password": "<password>"}'
```

Παραμέτρους: Όχι

Request Body: Application json (required)

```
{
  "email": "example@gmail.com",
  "password": "12345"
}
```

Responses:

| Code              | Description   |
|-------------------|---|
| 200               | {"auth_token": "eyJhbGciOiJIUzI1NiJ9.eyJ1c2VyX2lkIjo2LCJleHAiOjE2Nzc1MTAxNzR9.01VWHn5Wnx8W3R3Rw0-dmA02nNl5jJweDKKJgum4X_Y"} |
| 400 – Bad Request | Missing required values in request body   |

## 3. Logout

Έξοδος από την εφαρμογή.

Εντολή: **GET**

```
curl -X GET http://localhost:3000/auth/logout
```

Παραμέτρους: Όχι

Request Body: Όχι

Responses:

| Code | Description                             |
|------|---|
| 200  | {"message": "Logged out successfully."} |

## 4. Todos

### 1. Get all todos

Παρουσίαση όλων των λιστών.

Εντολή: **GET**

```
curl -X GET http://localhost:3000/todos
```

Παραμέτρους: Όχι

Request Body: Όχι

Responses:

| Code | Description   |
|------|---|
| 200  | [{"id":2,"title":"myTodo","created_by":"example1","created_at":"2023-02-26T15:14:35.178Z","updated_at":"2023-02-26T15:14:35.178Z"}] |

### 2. Get a todo

Παρουσίαση μίας λίστας.

Εντολή: **GET**

```
curl -X GET http://localhost:3000/todos/{id}
```

Παραμέτρους:

| Name          | Description         |
|---------------|---------------------|
| id (required) | Id μίας λίστας todo |

Request Body: Όχι

Responses:

| Code            | Description   |
|-----------------|---|
| 200             | {"id":2,"title":"myTodo","created_by":"example1","created_at":"2023-02-26T15:14:35.178Z","updated_at":"2023-02-26T15:14:35.178Z"} |
| 404 – Not Found | {"message":"Couldn't find Todo with 'id'=<id>"}   |

### 3. Create a new todo

Δημιουργία νέας λίστας.

Εντολή: **POST**

```
curl -X POST http://localhost:3000/todos -H "Content-Type: application/json" -d '{"title": "<myTodoTitle>", "created_by": "<username>"}'
```

Παραμέτρους: Όχι

Request Body: Application json (required)

```
{
  "title": "myTodo",
  "created_by": "example1"
}
```

Responses:

| Code                       | Description  |
|----------------------------|--|
| 200                        | <code>{"id":2,"title":"myTodo","created_by":"example1","created_at":"2023-02-26T15:14:35.178Z","updated_at":"2023-02-26T15:14:35.178Z"}</code> |
| 422 - Unprocessable Entity | Missing required values in request body  |

#### 4. Update a todo

Ενημέρωση μίας λίστας.

Εντολή: PUT

```
curl -X PUT http://localhost:3000/todos/{id} -H "Content-Type: application/json" -d '{"title": "<updatedTitle>"}
```

Παραμέτρους:

| Name          | Description         |
|---------------|---------------------|
| id (required) | Id μίας λίστας todo |

Request Body: Application json (required)

```
{
  "title": "myUpdatedTodo"
}
```

Responses:

| Code           | Description  |
|----------------|--|
| 200            | OK   |
| 404 -Not Found | <code>{"message":"Couldn't find Todo with 'id'={id}"}</code> |

#### 5. Delete a todo and its items

Διαγραφή μίας λίστας.

Εντολή: **DELETE**

```
curl -X DELETE http://localhost:3000/todos/{id}
```

Παραμέτρους:

| Name          | Description         |
|---------------|---------------------|
| id (required) | Id μίας λίστας todo |

Request Body: Όχι

Responses:

| Code            | Description                                      |
|-----------------|--|
| 200             | OK   |
| 404 – Not Found | {"message": "Couldn't find Todo with 'id'={id}"} |



## 5. Todo Items

### 1. Get all todo items of a todos list

Παρουσίαση μίας λίστας και των στοιχείων της.

Εντολή: GET

```
curl -X GET http://localhost:3000/todos/{id}/items
```

Παραμέτρους:

| Name          | Description         |
|---------------|---------------------|
| id (required) | Id μίας λίστας todo |

Request Body: Όχι

Responses:

| Code | Description   |
|------|---|
| 200  | {"id":3,"name":"item1","done":null,"todo_id":2,"created_at":"2023-02-26T15:23:11.225Z","updated_at":"2023-02-26T15:23:11.225Z"} |

### 2. Get a todo item

Παρουσίαση ενός στοιχείου μίας λίστας.

Εντολή: GET

```
curl -X GET http://localhost:3000/todos/{id}/items/{item_id}
```

Παραμέτρους:

| Name               | Description                       |
|--------------------|-----------------------------------|
| id (required)      | Id μίας λίστας todo               |
| item_id (required) | Id ενός στοιχείου της λίστας todo |

Request Body: Όχι

Responses:

| Code | Description   |
|------|---|
| 200  | {"id":3,"name":"item1","done":null,"todo_id":2,"created_at":"2023-02-26T15:23:11.225Z","updated_at":"2023-02-26T15:23:11.225Z"} |
| 404  | {"message":"Couldn't find Item with [WHERE \"items\".\"todo_id\" = ? AND \"items\".\"id\" = ?]\"}. Item not found.              |

### 3. Create a new todo item

Δημιουργία νέου αντικειμένου μίας λίστας.

Εντολή: **POST**

```
curl -X POST http://localhost:3000/todos/{id}/items -H "Content-Type: application/json" -d '{"name": "<item>"}
```

Παραμέτρους:

| Name          | Description         |
|---------------|---------------------|
| id (required) | Id μίας λίστας todo |

Request Body: Application json (required)

```
{  
  "name": "item1"  
}
```

Responses:

| Code                       | Description   |
|----------------------------|---|
| 200                        | {"id":2, "title":"myTodo", "created_by":"example1", "created_at":"2023-02-26T15:14:35.178Z", "updated_at":"2023-02-26T15:14:35.178Z"} |
| 422 - Unprocessable Entity | Missing required values in request body   |

### 4. Update a todo item

Ενημέρωση ενός στοιχείου μίας λίστας.

Εντολή: **PUT**

```
curl -X PUT http://localhost:3000/todos/{id}/items/{item_id} -H "Content-Type: application/json" -d '{"name": "<updatedItem>"}
```

Παραμέτρους:

| Name               | Description                       |
|--------------------|-----------------------------------|
| id (required)      | Id μίας λίστας todo               |
| item_id (required) | Id ενός στοιχείου της λίστας todo |

Request Body: Application json (required)

```
{  
  "name": "updateditem1"  
}
```

Responses:

| Code | Description |
|------|-------------|
|------|-------------|

|                 |  |
|-----------------|--|
| 200             | OK   |
| 404 – Not Found | { "message": "Couldn't find Item with [WHERE \"items\".\"todo_id\" = ? AND \"items\".\"id\" = ?]\"}. Item not found. |

## 5. Delete a todo item

Διαγραφή ενός στοιχείου μίας λίστας.

Εντολή: **DELETE**

```
curl -X DELETE http://localhost:3000/todos/{id}/items/{item_id}
```

Παραμέτρους:

| Name               | Description                       |
|--------------------|-----------------------------------|
| id (required)      | Id μίας λίστας todo               |
| item_id (required) | Id ενός στοιχείου της λίστας todo |

Request Body: Όχι

Responses:

| Code            | Description  |
|-----------------|--|
| 200             | OK   |
| 404 – Not Found | { "message": "Couldn't find Item with [WHERE \"items\".\"todo_id\" = ? AND \"items\".\"id\" = ?]\"}. Item not found. |