

Name: Filmon Araya  
Week\_3:Re\_Service Layers

The eTraining site will use the REST API's as Follows:

A RESTful web service **request** contains:

## 1, An Endpoint URL.

The application will implement a RESTful API that define one or more URL endpoints with a domain, port, path, and or querying string

<http://localhost:127.0.0.1:8000/user/123?format=json>

## 2, The HTTP method.

HTTP method	CRUD	Action	Web address
GET	read	returns requested data	<a href="http://localhost:127.0.0.1:8000/all">http://localhost:127.0.0.1:8000/all</a>
POST	create	creates a new record	<a href="http://localhost:127.0.0.1:8000/add">http://localhost:127.0.0.1:8000/add</a>
PUT or PATCH	update	updates an existing record	<a href="http://localhost:127.0.0.1:8000/update">http://localhost:127.0.0.1:8000/update</a>
DELETE	delete	deletes an existing record	<a href="http://localhost:127.0.0.1:8000/delete/&lt;int:id&gt;">http://localhost:127.0.0.1:8000/delete/&lt;int:id&gt;</a>

<http://127.0.0.1:8000/api/topics/?format=json>

```
[
  {
    "id":1,
    "topic_title":"Day-1-Orientation",
  },
  {
    "id":2,
```

```
    "title": "safety rule",
  },
  {
    "id": 3,
    "title": "polices and procedures",

  },
  {
    "id": 4,
    "title": "Introduction working rule",

  }
]
```

[http://127.0.0.1:8000/api/topics/\[id\]](http://127.0.0.1:8000/api/topics/[id])

<http://127.0.0.1:8000/api/topics/4>

```
{
  "id": 4,
  "title": "Introduction working rule",

}
```

## GET List users

Request [/api/users?page=2](#)

Response [200](#)

```
{
  "page": 2,
  "per_page": 6,
  "total": 12,
  "total_pages": 2,
  "data": [
    {
      "id": 7,
      "email": "michael.lawson@reqres.in",
      "first_name": "Michael",
      "last_name": "Lawson",
    }
  ]
}
```

## GET Single user

Request/api/users/2

Response200

```
{
  "data": {
    "id": 2,
    "email": "janet.weaver@reqres.in",
    "first_name": "Janet",
    "last_name": "Weaver",

  },
  "ad": {
    "company": "StatusCode Weekly",
    "url": "http://statuscode.org/",
    "text": "A weekly newsletter focusing on software development, infrastructure, the server, performance, and the stack end of things."
  }
}
```

## GET Single user not found

Request/api/users/23

Response404

```
{}
```

## Login – successful

Request/api/login

```
{
  "email": "eve.holt@reqres.in",
  "password": "cityslicka"
}
```

Response200

```
{
```

```
    "token": "QpwL5tke4Pnpja7X4"
  }
```

## Login-unsuccessful

Request/`api/login`

```
{
  "email": "peter@klaven"
}
```

Response`400`

```
{
  "error": "Missing password"
}
```

## GET Delayed Response

Request/`api/users?delay=3`

Response`200`

```
{
  "page": 1,
  "per_page": 6,
  "total": 12,
  "total_pages": 2,
  "data": [
    {
      "id": 1,
      "email": "george.bluth@reqres.in",
      "first_name": "George",
      "last_name": "Bluth",

```

## DELETE

Request/`api/users/2`

Response`204`

## [HTTP status code](#) and Handling authentication

response header.

HTTP 200: OK is most often used for successful requests

HTTP 400: Bad Request,

HTTP 404: Not Found,

Request/[api/unknown/23](#)

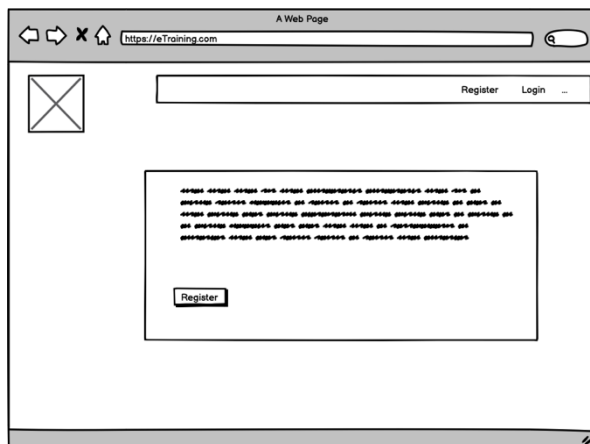
Response404

```
{}
```

HTTP 401: Unauthorized,

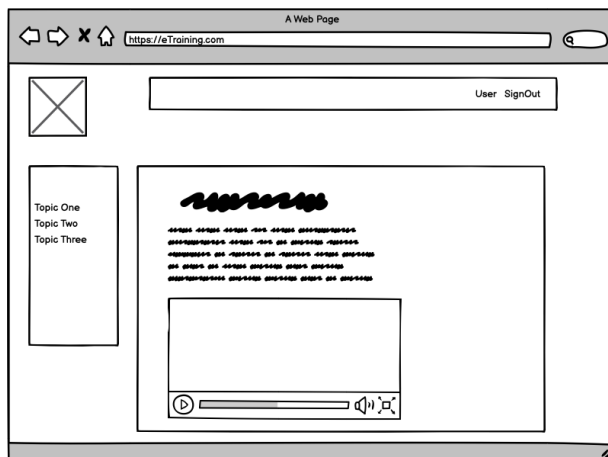
HTTP 403: Permission denied

[http://localhost:127.0.0.1:8000/api/topics\\_content/](http://localhost:127.0.0.1:8000/api/topics_content/) | j



<http://localhost:127.0.0.1:8000/api>

- user landing page



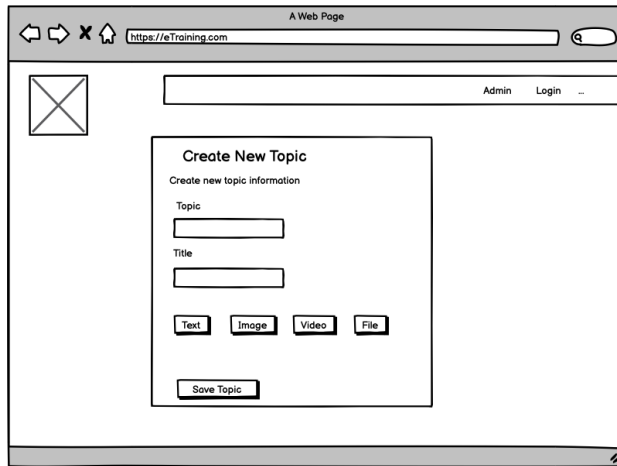
GET

<http://localhost:127.0.0.1:8000/api/user/id> ---

Json

user can see all topics after created a registration

Post



The screenshot shows a web browser window titled "A Web Page" with the address bar displaying "http://eTraining.com". The page content includes a placeholder image (a box with an 'X'), a navigation bar with "Admin" and "Login" links, and a "Create New Topic" form. The form contains fields for "Topic" and "Title", a row of buttons for "Text", "Image", "Video", and "File", and a "Save Topic" button at the bottom.

<http://localhost:127.0.0.1:8000/api/admin/topic> --Json

admin can post

PUT

<http://localhost:127.0.0.1:8000/api/>--json

admin can edit or delete

