

TECMILENIO



CIATEQ RESEARCHERS PROJECT

WEB DESIGN

TEAM

Juan Pablo Vázquez Arévalo

Carlos Eduardo Sanchez

PROFESSOR

Francisco Javier Olivera

Frameworks:

- React.jsx
- Django
- Bootstrap (only backend templates)
- Django REST Framework

Libraries:

- Axios
- JWT

Languages:

- JavaScript (JSX)
- Python
- HTML
- CSS

Dependencies:

- Corsheaders
- Custom Middleware
- Postman

Installation instructions

On cd \backend\mysite:

- pip install django-cors-headers==4.7.0
- pip install djangorestframework==3.15.2
- pip install djangorestframework-simplejwt==5.5.0

On cd Centro-de-investigacion:

- npm install
- npm install axios

Login

The user enters their email and password into a React form. Upon submitting the data, a POST request is made to the Django backend (/login-api/). If the credentials are valid, the backend returns a user ID (user_id), which is saved in localStorage to keep the session active on the frontend. Additionally, the backend stores the data in the Django session.

If authentication fails, an error message is displayed. Upon logging out, the information is deleted and the user is redirected to the login page.

Iniciar Sesión

Correo:

juan.perez@example.com

Contraseña:

.....

Ingresar

Sidebar Menu

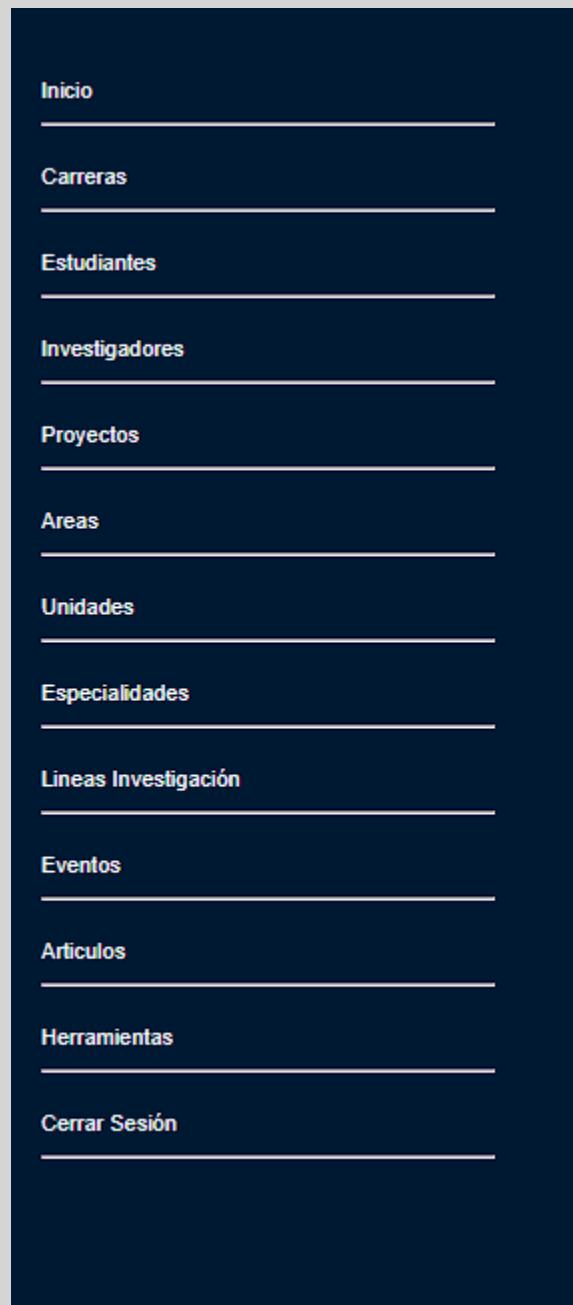
The sidebar menu in React contains links to various sections of the system. The list of options is defined in a `menuList` array, which includes paths such as Careers, Students, Projects, and others.

Authentication:

- If the user is logged in (checked via `idusuario` in `localStorage`), a **Log Out** link is displayed.
- If the user is not logged in, links for **Login** and **Register** are shown.

Redirection:

- When clicking any menu item, the user is redirected to the corresponding route using **React Router (Link)**.
- When the user clicks "Log Out," the `idusuario` and `idpermiso` are removed from `localStorage`, and the user is redirected to the home page.



Careers P

The **Careers** section allows users to view, add, edit, and delete careers. This functionality is implemented in React and connects to a Django backend.

1. Display Careers:

- When the page loads, a GET request is made to the API (/api/carreras-api/) to fetch the list of careers stored in the database.
- If there's an error loading the careers, an error message is displayed.

2. Add Career:

- The user can add a new career by filling out a form. When the form is submitted, a POST request is made to the API to create the career in the database.
- If the creation is successful, the new career is added to the list.

3. Edit Career:

- To edit a career, the user can click the "Edit" button next to the career in the table. This displays a form with the current data of the career.
- When the form is submitted, a PUT request is made to the API to update the career's data.

4. Delete Career:

- The user can delete a career by clicking the "Delete" button. After confirming the action, a DELETE request is made to the API to remove the career from the database.

Listado de carreras

+ Nueva Carrera

Nueva Carrera

Nombre Carrera:

Universidad:

Guardar Cancelar

ID	Nombre Carrera	Universidad	Acciones
2	Bioquimica	ITESO	<button>Editar</button> <button>Eliminar</button>
3	Lic. Pedagogía	UNIVA	<button>Editar</button> <button>Eliminar</button>
4	Ing. Electricidad	TecMilenio	<button>Editar</button> <button>Eliminar</button>
5	Ing en Electronica	Anahuac	<button>Editar</button> <button>Eliminar</button>
6	Ing en Mecatronica	Tecnologico de Monterrey	<button>Editar</button> <button>Eliminar</button>

Editar Carrera

Nombre Carrera:

 Bioquimica

Universidad:

 ITESO

Actualizar Cancelar

localhost:5173 dice

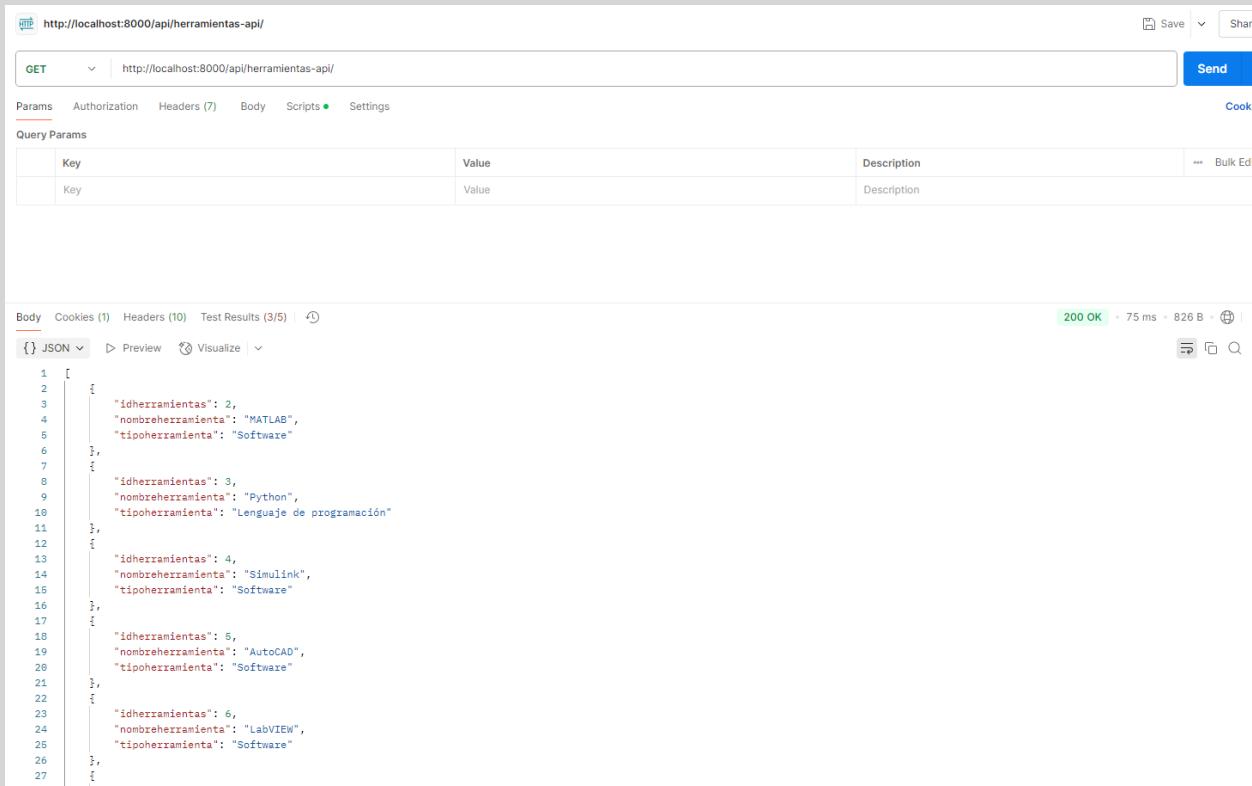
¿Estás seguro de eliminar esta carrera?

Aceptar

Cancelar

POSTMAN EXAMPLES: TABLE CAREERS:

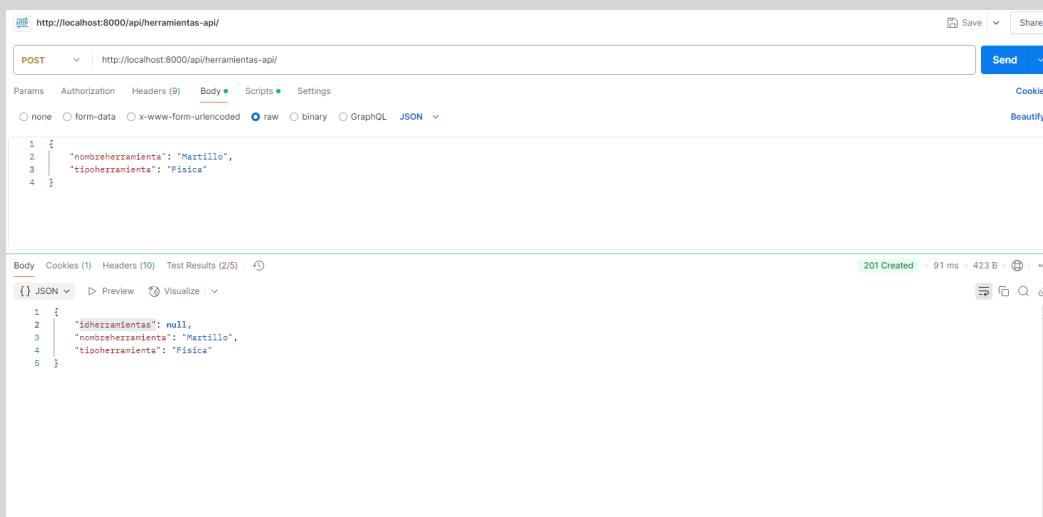
GET:



The screenshot shows a Postman request to `http://localhost:8000/api/herramientas-api/`. The response is a 200 OK status with a response time of 75 ms and a size of 826 B. The JSON data returned is as follows:

```
1 [
2   {
3     "idherramientas": 2,
4     "nombreherramienta": "MATLAB",
5     "tipoherramienta": "Software"
6   },
7   {
8     "idherramientas": 3,
9     "nombreherramienta": "Python",
10    "tipoherramienta": "Lenguaje de programación"
11  },
12  {
13    "idherramientas": 4,
14    "nombreherramienta": "Simulink",
15    "tipoherramienta": "Software"
16  },
17  {
18    "idherramientas": 5,
19    "nombreherramienta": "AutoCAD",
20    "tipoherramienta": "Software"
21  },
22  {
23    "idherramientas": 6,
24    "nombreherramienta": "LabVIEW",
25    "tipoherramienta": "Software"
26  },
27]
```

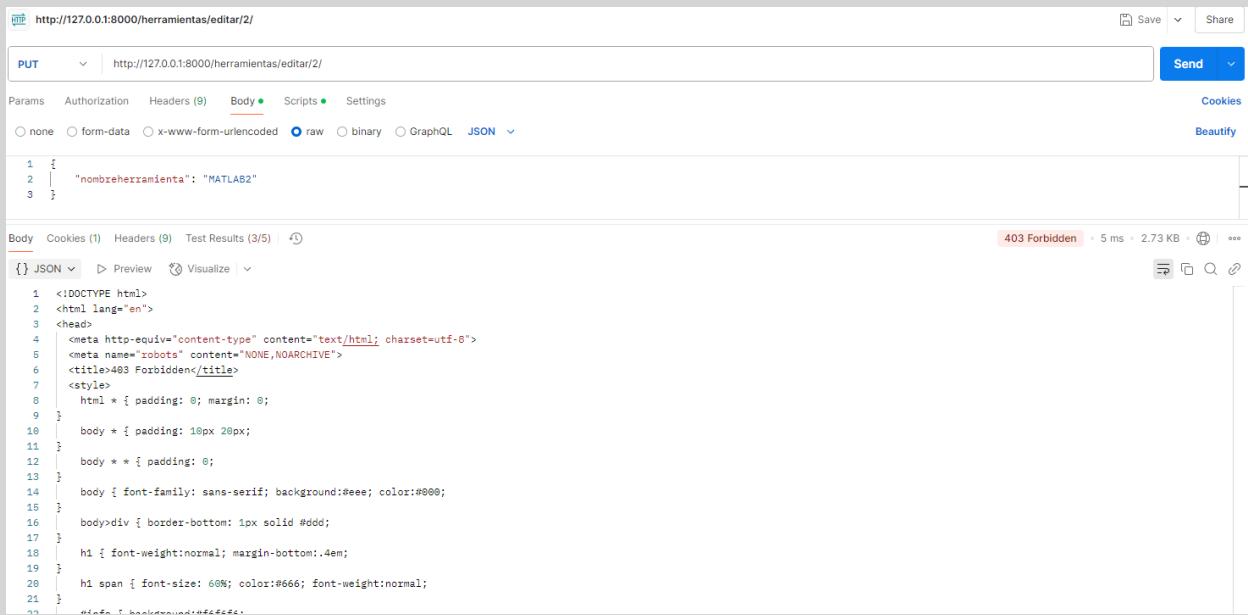
POST:



The screenshot shows a Postman request to `http://localhost:8000/api/herramientas-api/` using the POST method. The response is a 201 Created status with a response time of 91 ms and a size of 423 B. The JSON data returned is as follows:

```
1 {
2   "idherramientas": null,
3   "nombreherramienta": "Martillo",
4   "tipoherramienta": "Fisica"
5 }
```

PUT:



http://127.0.0.1:8000/herramientas/editar/2

PUT http://127.0.0.1:8000/herramientas/editar/2

Params Authorization Headers (9) Body Scripts Settings

Body (1) Headers (9) Test Results (3/5)

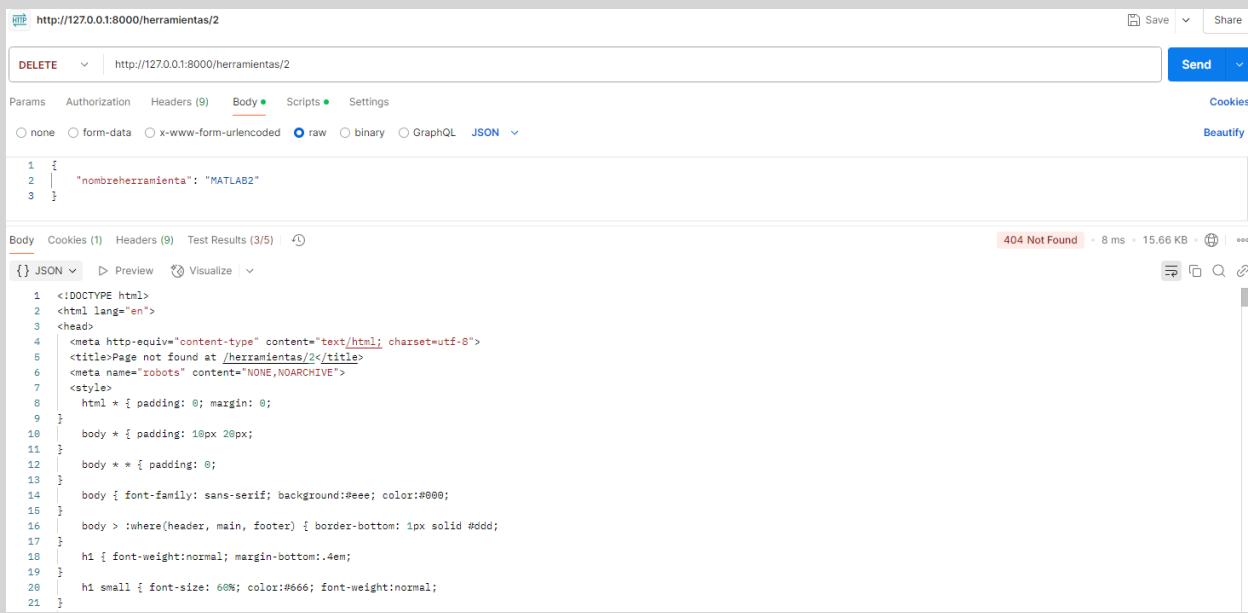
403 Forbidden 5 ms 2.73 KB

```
1 {  
2   "nombreherramienta": "MATLAB2"  
3 }
```

{} JSON Preview Visualize

```
1 <!DOCTYPE html>  
2 <html lang="en">  
3 <head>  
4   <meta http-equiv="content-type" content="text/html; charset=utf-8">  
5   <meta name="robots" content="NONE,NOARCHIVE">  
6   <title>403 Forbidden</title>  
7   <style>  
8     html * { padding: 0; margin: 0;  
9   }  
10  body * { padding: 10px 20px;  
11  }  
12  body * * { padding: 0;  
13  }  
14  body { font-family: sans-serif; background:#eee; color:#000;  
15  }  
16  body>div { border-bottom: 1px solid #ddd;  
17  }  
18  h1 { font-weight:normal; margin-bottom:.4em;  
19  }  
20  h1 span { font-size: 60%; color:#666; font-weight:normal;  
21  }  
22  minfo { background-color:#f0f0f0;
```

DELETE:



http://127.0.0.1:8000/herramientas/2

DELETE http://127.0.0.1:8000/herramientas/2

Params Authorization Headers (9) Body Scripts Settings

Body (1) Headers (9) Test Results (3/5)

404 Not Found 8 ms 15.66 KB

```
1 {  
2   "nombreherramienta": "MATLAB2"  
3 }
```

{} JSON Preview Visualize

```
1 <!DOCTYPE html>  
2 <html lang="en">  
3 <head>  
4   <meta http-equiv="content-type" content="text/html; charset=utf-8">  
5   <title>Page not found at /herramientas/2</title>  
6   <meta name="robots" content="NONE,NOARCHIVE">  
7   <style>  
8     html * { padding: 0; margin: 0;  
9   }  
10  body * { padding: 10px 20px;  
11  }  
12  body * * { padding: 0;  
13  }  
14  body { font-family: sans-serif; background:#eee; color:#000;  
15  }  
16  body> :where(header, main, footer) { border-bottom: 1px solid #ddd;  
17  }  
18  h1 { font-weight:normal; margin-bottom:.4em;  
19  }  
20  h1 small { font-size: 60%; color:#666; font-weight:normal;  
21  }
```

DIAGRAM

