

Proyecto Full stack

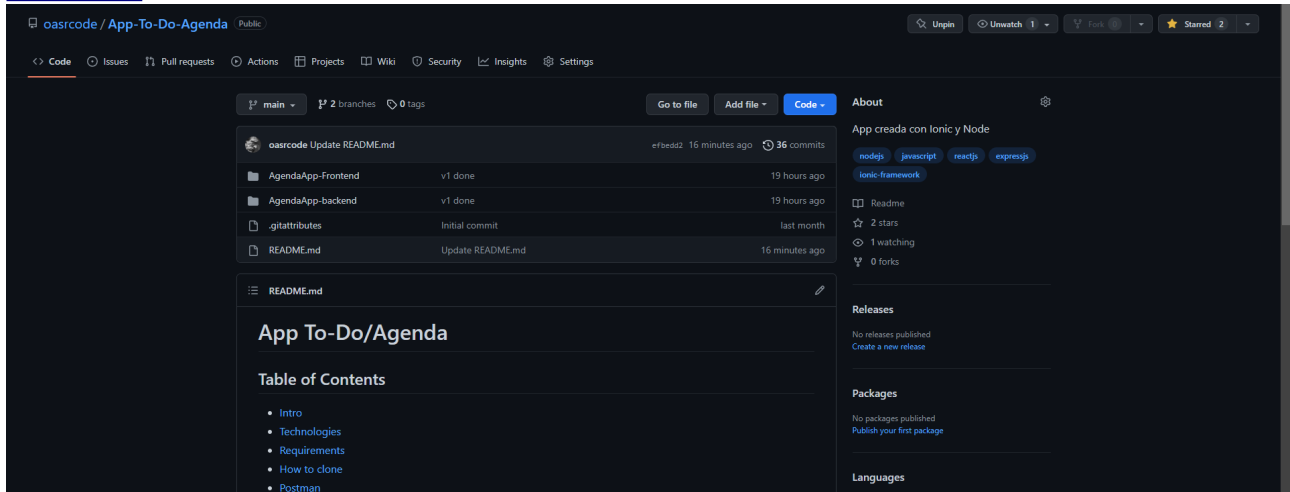
App To Do/ Agenda con Ionic

Índice

1.Capturas Github.....	3
2.Capturas Postman.....	3
3.CRUD ToDos.....	4
4.CRUD Photos.....	6
5.CRUD User.....	8
6.CRUD desde IONIC.....	9
7.Componentes de Ionic.....	12

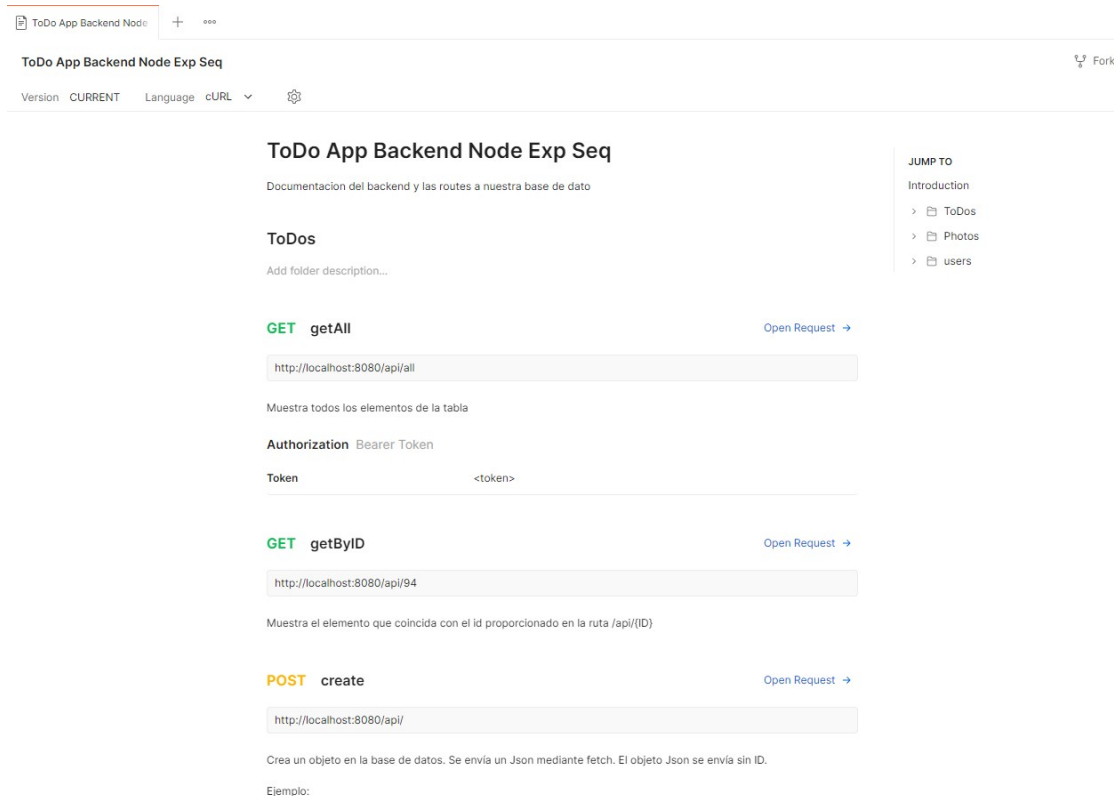
1.Capturas Github

Capturas del repositorio github, donde se el proyectos subido con su readme. [Clic para ir al enlace.](#)



2.Capturas Postman

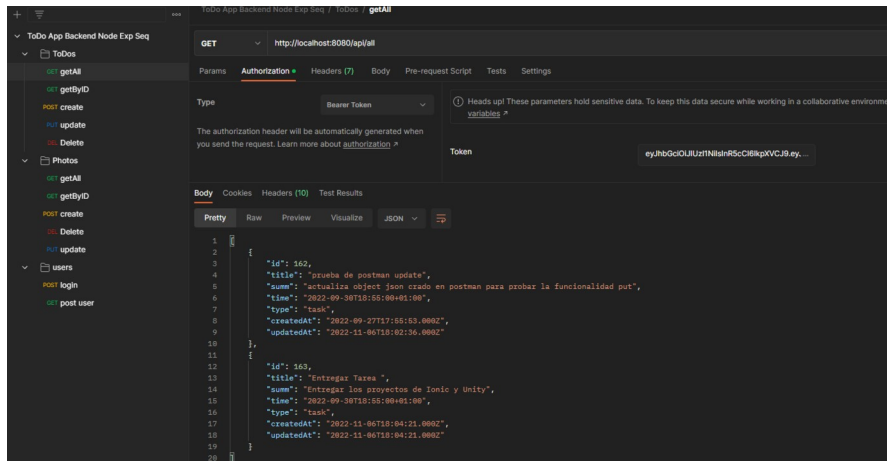
[Enlace de postman donde están todas la request al backend.](#)



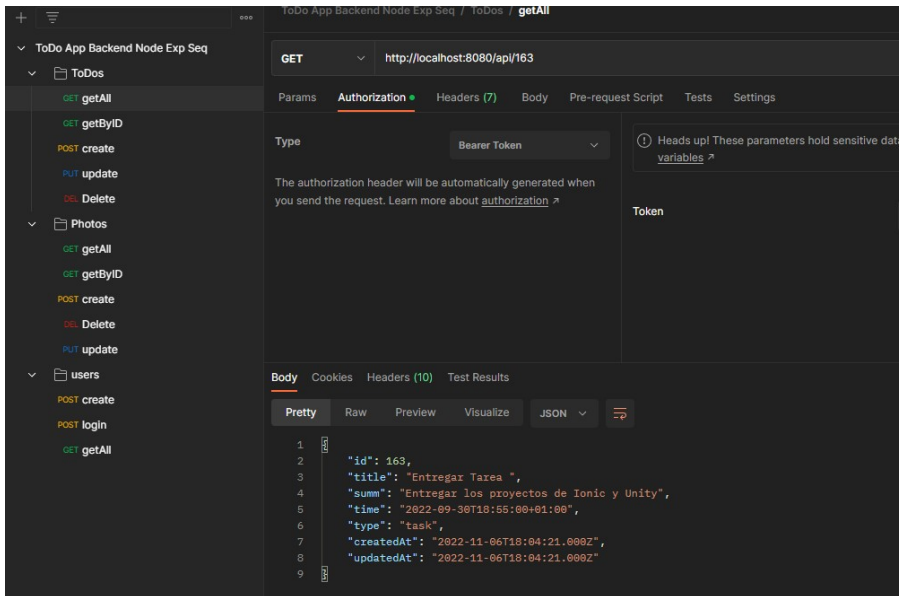
3.CRUD Todos

Todos usa bearer tokens para poder realizar el crud.

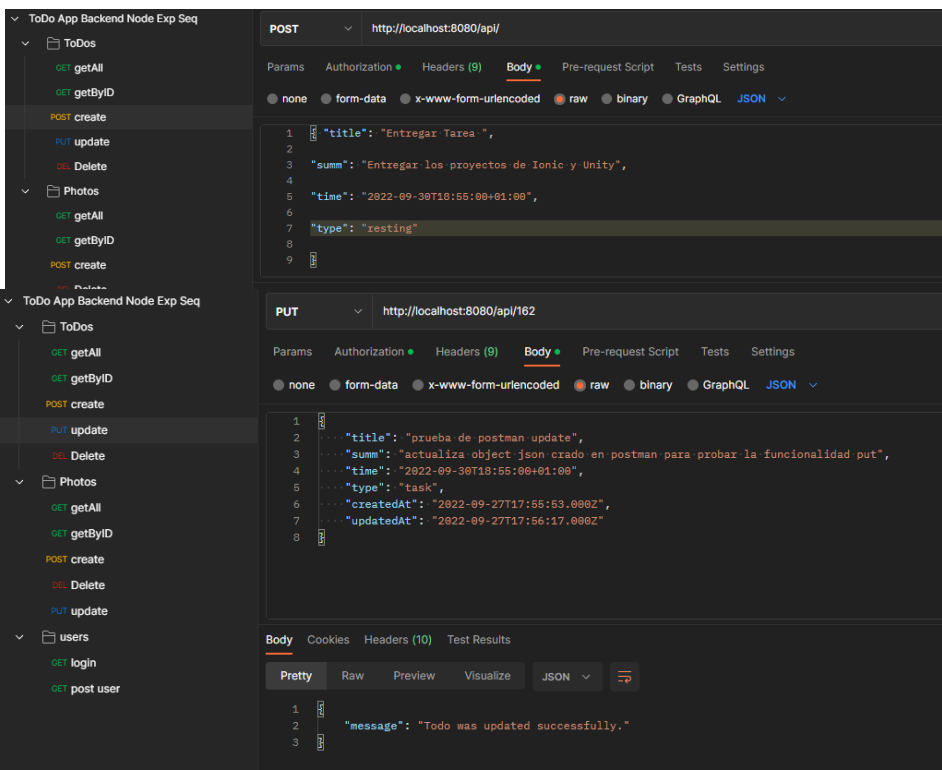
Get All



Get ID

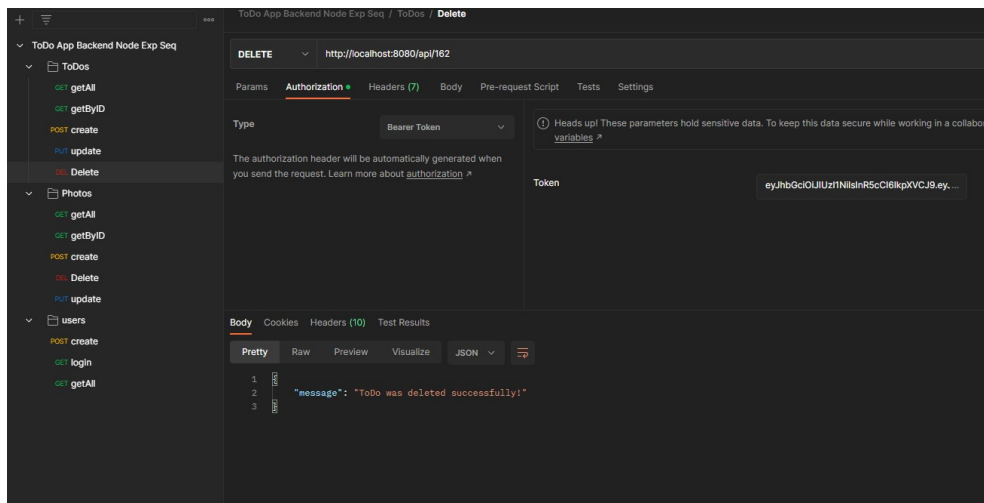


Post



Put

Delete



4.CRUD Photos

Photos usa bearer tokens para poder realizar el crud.

Get All

The screenshot shows the Postman interface for the 'getAll' endpoint in the 'Photos' collection. The request is a GET to 'http://localhost:8080/api/photos/all'. The 'Authorization' tab is selected, showing a Bearer Token. The 'Body' tab is also visible, showing a JSON response with two photo objects.

```
GET http://localhost:8080/api/photos/all
```

Params Authorization Headers (7) Body Pre-request Script Tests Settings

Type Bearer Token

The authorization header will be automatically generated when you send the request. Learn more about [authorization](#)

Token

Body Cookies Headers (10) Test Results

Pretty Raw Preview Visualize JSON

```
1 {
2   "id": 235,
3   "title": "imagen 7",
4   "filename": "",
5   "createdAt": "2022-11-06T19:10:18.000Z",
6   "updatedAt": "2022-11-06T19:10:18.000Z"
7 },
8 {
9   "id": 236,
10  "title": "imagen 7",
11  "filename": "",
12  "createdAt": "2022-11-06T19:21:11.000Z",
13  "updatedAt": "2022-11-06T19:21:11.000Z"
14 }
15 }
```

Get ID

The screenshot shows the Postman interface for the 'getByID' endpoint in the 'Photos' collection. The request is a GET to 'http://localhost:8080/api/photos/235'. The 'Authorization' tab is selected, showing a Bearer Token. The 'Body' tab is also visible, showing a JSON response with a single photo object.

```
GET http://localhost:8080/api/photos/235
```

Params Authorization Headers (7) Body Pre-request Script Tests Settings

Type Bearer Token

The authorization header will be automatically generated when you send the request. Learn more about [authorization](#)

Token

Body Cookies Headers (10) Test Results

Pretty Raw Preview Visualize JSON

```
1 {
2   "id": 235,
3   "title": "imagen 7",
4   "filename": "",
5   "createdAt": "2022-11-06T19:10:18.000Z",
6   "updatedAt": "2022-11-06T19:10:18.000Z"
7 }
```

The screenshot shows the Postman interface for the 'create' endpoint in the 'Photos' collection. The request is a POST to 'http://localhost:8080/api/photos/'. The 'Body' tab is selected, showing a form with 'title' and 'file' fields. The 'Body' tab is also visible, showing a JSON response with a single photo object.

```
POST http://localhost:8080/api/photos/
```

Params Authorization Headers (9) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL

KEY	VALUE
<input checked="" type="checkbox"/> title	imagen 7
<input checked="" type="checkbox"/> file	Sea-wave-Wallpaper-300x585.jpg
Key	Value

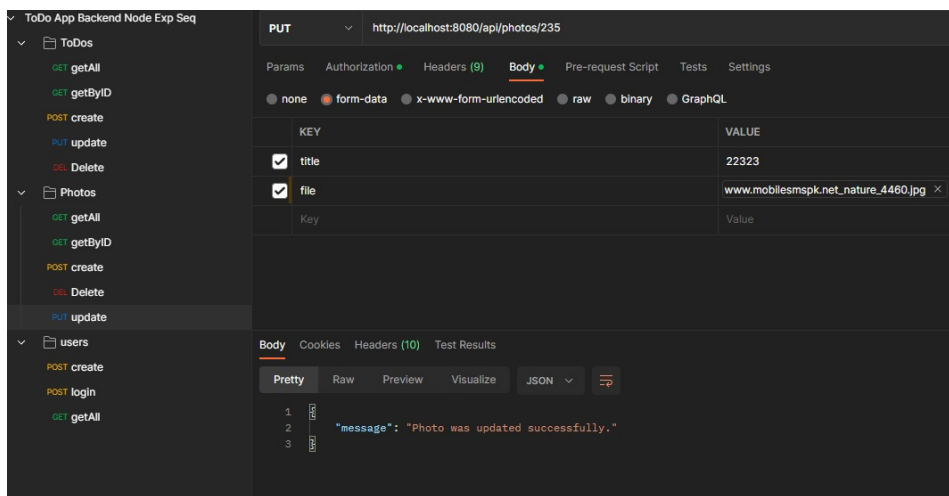
Body Cookies Headers (10) Test Results

Pretty Raw Preview Visualize JSON

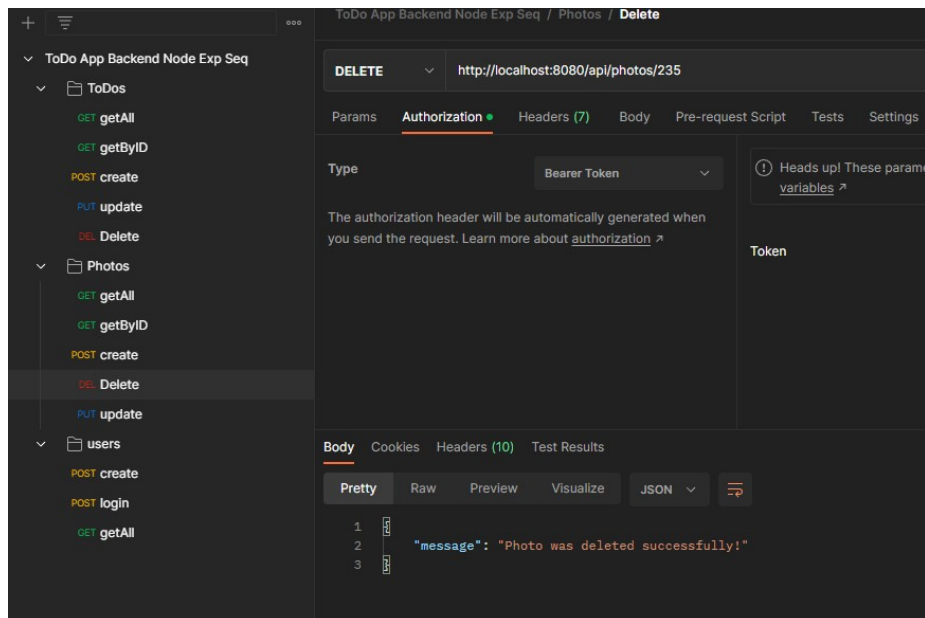
```
1 {
2   "id": 236,
3   "title": "imagen 7",
4   "filename": "",
5   "updatedAt": "2022-11-06T19:21:11.486Z",
6   "createdAt": "2022-11-06T19:21:11.486Z"
7 }
```

Post

Put

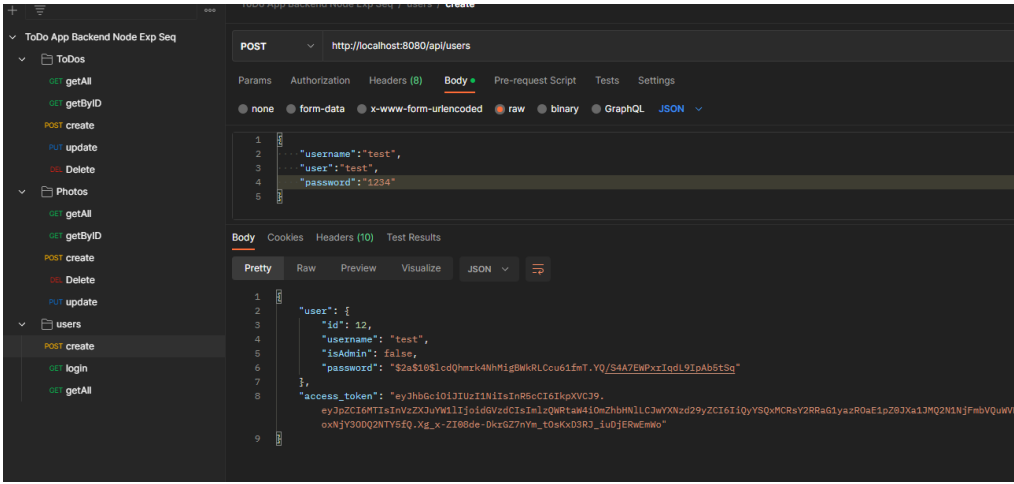


Delete

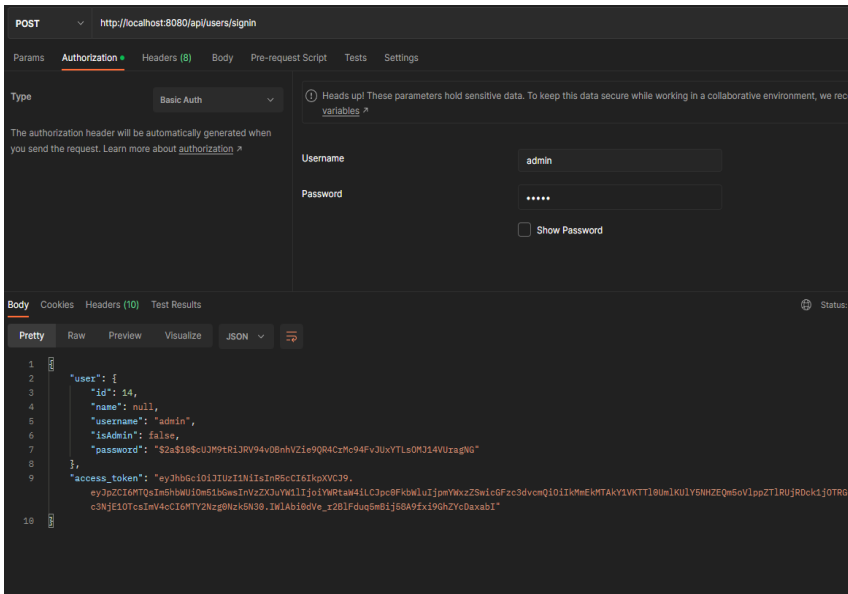


5.CRUD User

Post



SignIn



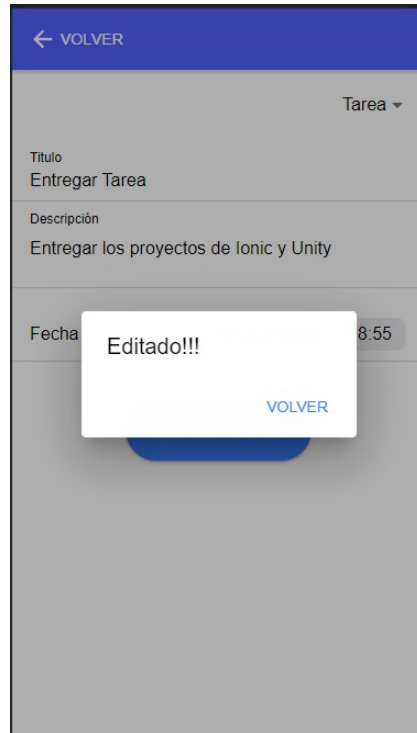
6.CRUD desde IONIC

ToDoS

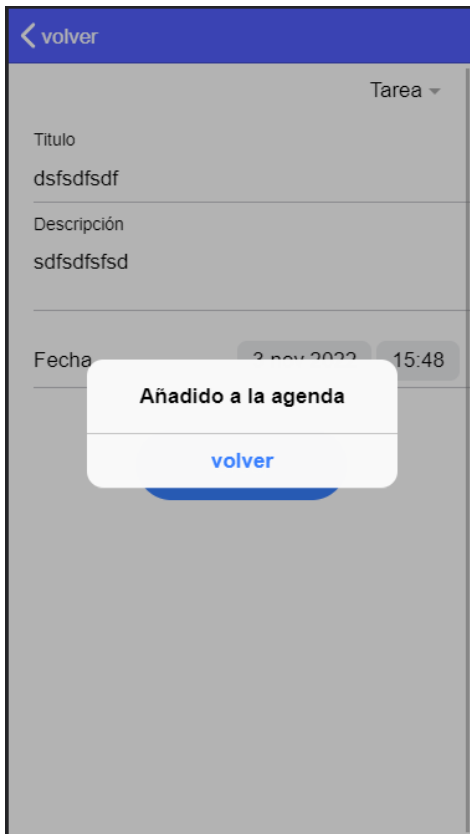
Get All



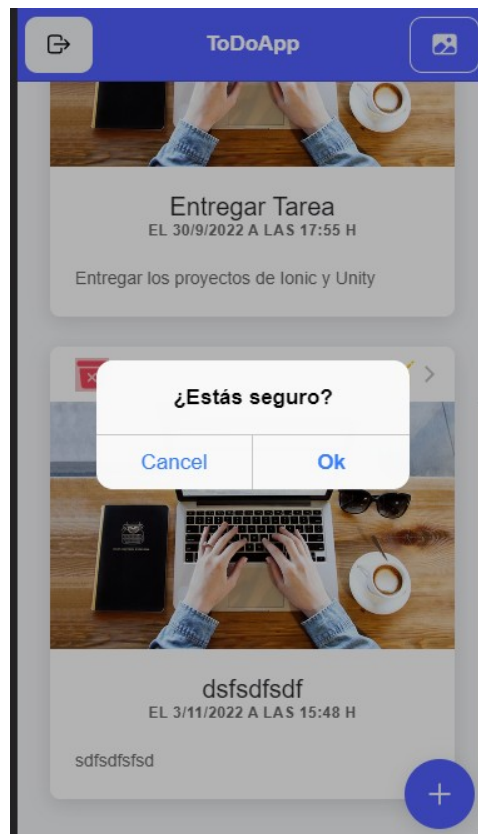
Edit



Create

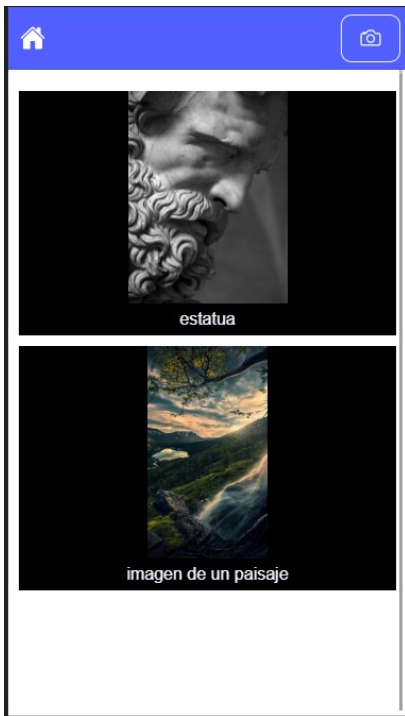


Delete

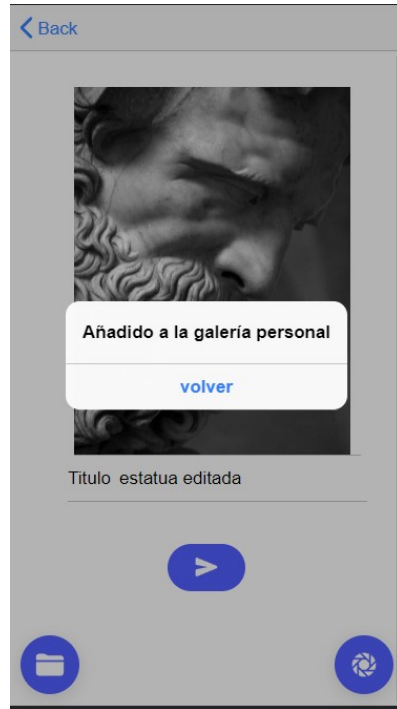


Photos

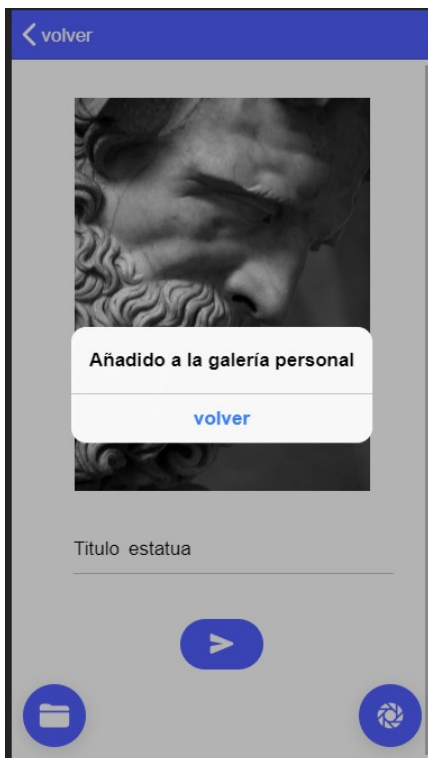
Get All



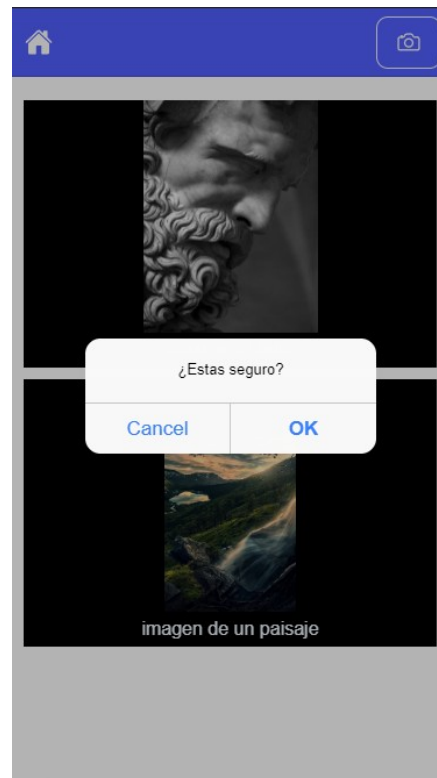
Edit: clic sobre la foto para ver el menú



Create

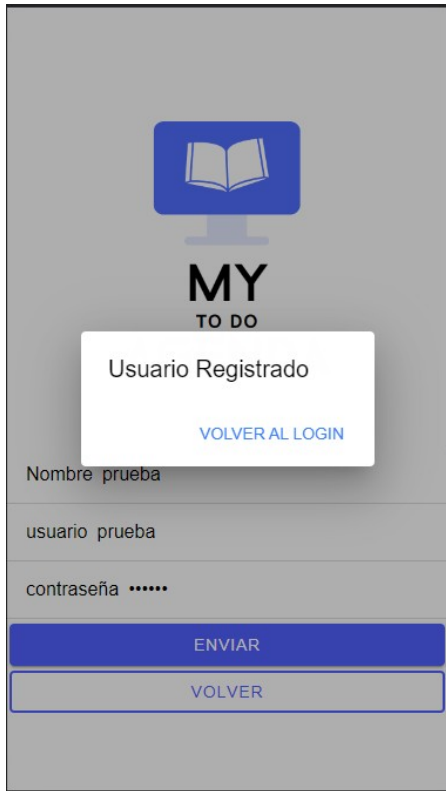


Delete:clic sobre la foto para ver el menú



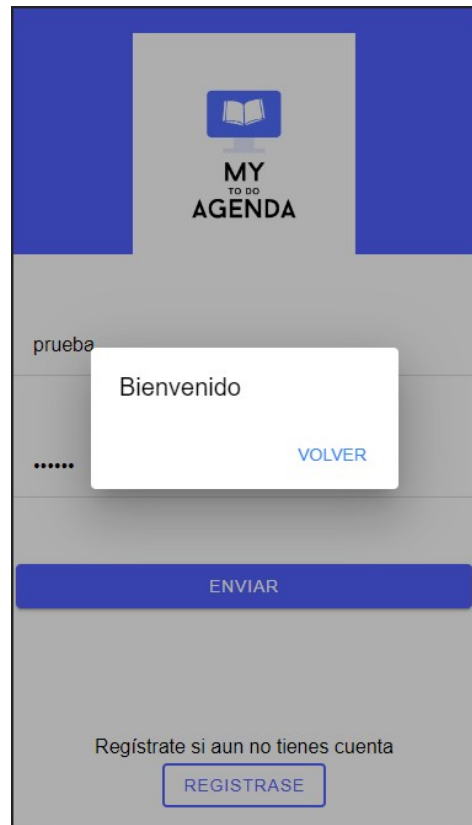
User

Registrar usuario



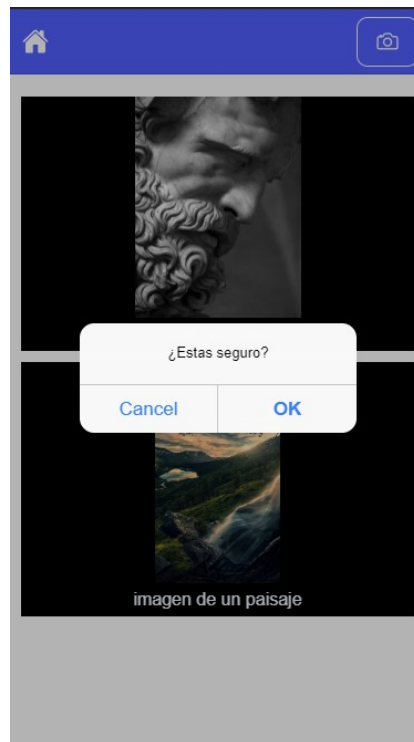
The registration form for 'MY TO DO' features a logo at the top consisting of a blue book icon on a stand, with the text 'MY TO DO' below it. The form includes three input fields: 'Nombre prueba', 'usuario prueba', and 'contraseña'. Below these fields are two buttons: a solid blue 'ENVIAR' button and a white 'VOLVER' button with a blue border. A white modal box is displayed in the center, containing the text 'Usuario Registrado' and a blue link 'VOLVER AL LOGIN'.

Login Usuario



The login form for 'MY TO DO AGENDA' has a logo at the top with a blue book icon on a stand and the text 'MY TO DO AGENDA'. It contains two input fields: 'prueba' and '.....'. Below these is a solid blue 'ENVIAR' button. A white modal box is centered on the screen, showing 'Bienvenido' and a blue link 'VOLVER'. At the bottom of the form, there is a text prompt 'Regístrate si aun no tienes cuenta' and a blue button labeled 'REGISTRASE'.

7.Componentes de Ionic



- IonPage
- IonContent
- IonHeader
- IonToolBar
- IonButton
- IonButtons
- IonBackButton
- IonItem
- IonIcon
- IonCard
 - IonCardHeader
 - IonCardTitle
 - IonCardSubtitle
 - IonCardContent
 - IonLabel
- IonGrid
- IonFab
- IonImage
- IonInput
- IonSelect
- IonSelectOption
- IonTextArea
- IonDateTimeButton
- IonModal
- IonDateTime
- IonActionSheet
- IonAlert
 - useIonAlert()

