

Mini Dossier - AthletIA

Integrantes: Gabriel Cevallos, David Paccha, Ivan Fernandez

Introducción.

El propósito de la API es proporcionar un conjunto completo de operaciones para gestionar una colección de usuarios y ejercicios de entrenamiento. Permite a las aplicaciones crear, leer, actualizar y eliminar (CRUD) ejercicios, facilitando así la integración de funcionalidades de fitness en cualquier software.

Endpoints Desarrollados.

1. AUTH

Este módulo gestiona la autenticación, registro y seguridad de las cuentas.

Ruta	Método	Descripción	Parámetros	Códigos de Respuesta
/auth/login	Post	Autentica credenciales y emite un token JWT.	Body: <code>email</code> (string), <code>password</code> (string)	200, 400, 401, 500
/auth/register-account	Post	Registra una nueva cuenta de usuario.	Body: <code>email</code> (string), <code>password</code> (string)	201, 400, 500
/auth/complete-profile-setup	Post	Completa la configuración del perfil de un usuario.	Body: <code>accountId</code> (string), <code>profileRequest</code> (object)	201, 400, 500
/auth/change-password	Patch	Permite a un usuario autenticado cambiar su contraseña.	Body: <code>accountId</code> (string), <code>oldPassword</code> (string), <code>newPassword</code> (string)	200, 400, 401, 404, 500
/auth/refresh-token	Post	Genera un nuevo token de acceso usando un token de refresco.	Body: <code>refreshToken</code> (string)	200, 400, 500
/auth/logout	Post	Cierra la sesión del usuario invalidando su token de refresco.	Body: <code>accountId</code> (string)	200, 401, 500

2. ACCOUNT

Endpoints dedicados a la administración de cuentas de usuario. Requieren roles específicos.

Ruta	Método	Descripción	Parámetros	Códigos de Respuesta
/users	Get	Obtiene una lista paginada de todos los usuarios. Requiere el rol de ADMIN o MODERATOR.	Query: page, limit, search	200, 403, 500
/users/:id	Get	Obtiene la información detallada de un usuario. Requiere el rol de ADMIN o MODERATOR.	URL: id del usuario	200, 401, 403, 404, 500
/users/:id/suspend	Patch	Suspende la cuenta de un usuario. Requiere el rol de ADMIN o MODERATOR.	URL: id del usuario	200, 401, 403, 404, 500
/users/:id/give-role	Patch	Asigna o cambia el rol de un usuario. Requiere rol de ADMIN.	URL: id del usuario Body: role	200, 400, 401, 403, 404, 500

3. PROFILES

Módulo para la gestión de los perfiles públicos y privados de los usuarios.

Ruta	Método	Descripción	Parámetros	Códigos de Respuesta
/profiles/:id	Get	Obtiene la información detallada de un perfil. Requiere autenticación.	URL: id del perfil	200, 401, 404, 500
/profiles/:id	Patch	Actualiza la información de un perfil. Un usuario solo puede actualizar su propio perfil (a menos que sea ADMIN).	URL: id del perfil Body: campos a actualizar	200, 400, 401, 403, 404, 500

4. EXERCISES

Endpoints para la gestión de los ejercicios de entrenamiento.

Ruta	Método	Descripción	Parámetros	Códigos de Respuesta
/workout/exercises	Post	Crea un nuevo ejercicio.	Body: datos del ejercicio	201, 400, 500
/workout/exercises	Get	Obtiene una lista de todos los ejercicios.	Ninguno	200, 500
/workout/exercises/:id	Get	Obtiene la información detallada de un ejercicio.	URL: id del ejercicio	200, 404, 500
/workout/exercises/:id	Patch	Actualiza parcialmente un ejercicio existente.	URL: id del ejercicio Body: campos a actualizar	200, 400, 404, 500
/workout/exercises/:id	Delete	Elimina un ejercicio de forma permanente.	URL: id del ejercicio	204, 404, 500

Pruebas Realizadas en Postman.

The screenshot shows the Postman application interface. On the left, there's a sidebar with 'AthletIA' selected, containing collections like 'Auth', 'Person', and 'Workout'. The main workspace shows a POST request to 'Logout' with the URL `http://AthletIA/Auth/Logout`. The 'Body' tab is selected, showing a raw JSON payload with the key 'accountId': "7e654954-0f67-49a9-9067-7fe9df0bb544". The response status is '200 OK' with a duration of '102 ms' and a size of '287 B'. The response body is a JSON object with 'success': true and 'message': "Logged out successfully".

This screenshot is identical to the one above, showing the same POST Logout request in Postman. The URL is `http://AthletIA/Auth/Logout`, the body is a raw JSON object with 'accountId': "7e654954-0f67-49a9-9067-7fe9df0bb544", and the response is a 200 OK status with the message "Logged out successfully".

Home Workspaces API Network

Search Postman Ctrl K

AthletIA / Account / findById

GET {{(domain)}} /users/7e654954-0f67-49a9-9067-7fe9df0bb544

Params Authorization Headers (9) Body Scripts Settings Cookies

Headers:

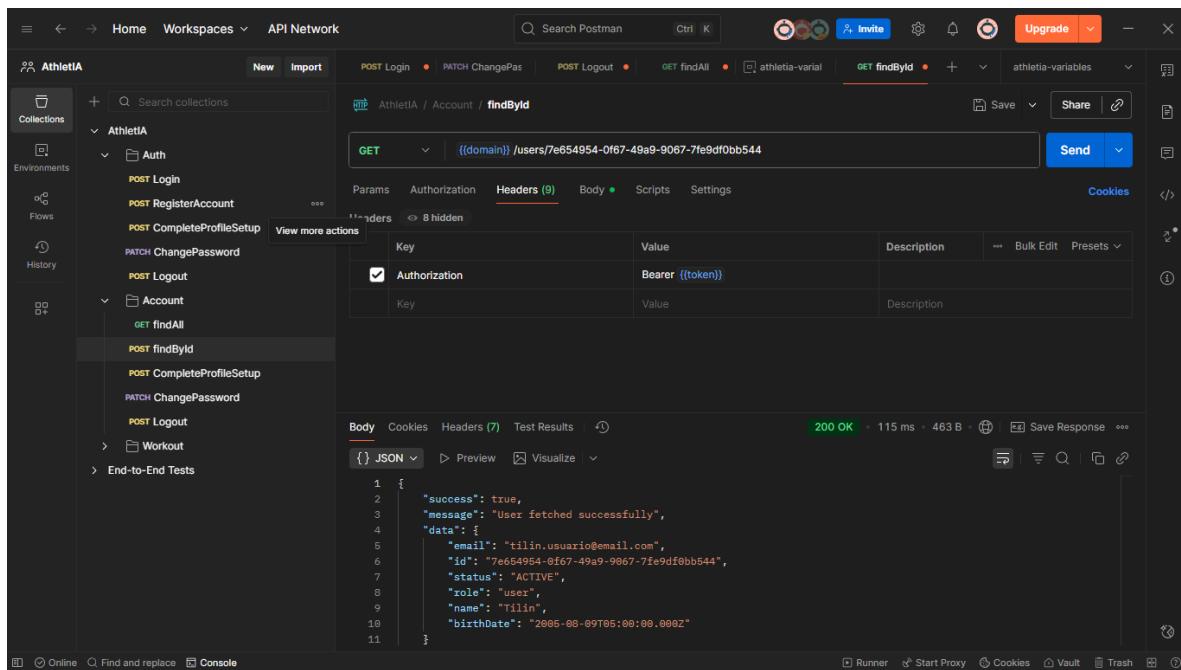
Key	Value
Authorization	Bearer {{(token)}}

Body:

```
{ "success": true, "message": "User fetched successfully", "data": { "email": "tilin.usuario@email.com", "id": "7e654954-0f67-49a9-9067-7fe9df0bb544", "status": "ACTIVE", "role": "user", "name": "Tilin", "birthDate": "2005-08-09T05:00:00.000Z" } }
```

200 OK 115 ms 463 B Save Response

Runner Start Proxy Cookies Vault Trash



Home Workspaces API Network

Search Postman Ctrl K

AthletIA / Account / findAll

GET {{(domain)}} /users?limit=5&offset=0

Params Authorization Headers (7) Body Scripts Settings Cookies

Headers:

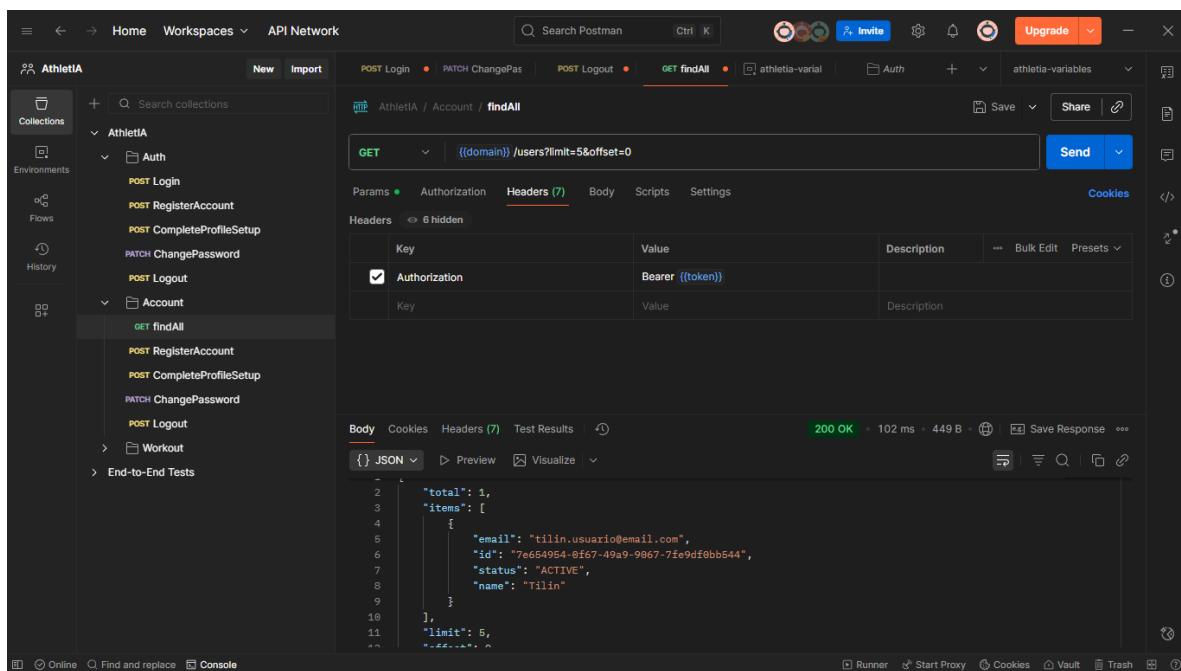
Key	Value
Authorization	Bearer {{(token)}}

Body:

```
{ "total": 1, "items": [ { "email": "tilin.usuario@email.com", "id": "7e654954-0f67-49a9-9067-7fe9df0bb544", "status": "ACTIVE", "name": "Tilin" } ], "limit": 5, "offset": 0 }
```

200 OK 102 ms 449 B Save Response

Runner Start Proxy Cookies Vault Trash



Home Workspaces API Network

AthletIA

POST RegisterAccount

HTTP AthletIA / Auth / RegisterAccount

POST

Params Authorization Headers (8) Body Scripts Settings

Body (raw)

```
1 {{normal-user}}
```

Body Cookies Headers (7) Test Results

201 Created 413 ms 381 B Save Response

1 {
2 "success": true,
3 "message": "Account was registered, continue with profile setup",
4 "data": {
5 "accountId": "3eadb0b-1d56-47c3-9269-6bd4711e20e9"
6 }
7 }

Online Find and replace Console

Runner Start Proxy Cookies Vault Trash

This screenshot shows the Postman interface for a POST request to the 'RegisterAccount' endpoint of the AthletIA API. The request URL is `https://{{domain}}/auth/register-account`. The 'Body' tab is selected with the raw JSON payload `1 {{normal-user}}`. The response status is 201 Created, and the response body is a JSON object indicating success and providing an account ID.

Home Workspaces API Network

AthletIA

PATCH Logon POST Logout GET findAll GET findById PATCH suspend POST suspend PATCH ChangePassword POST Logout

PATCH /users/d8ea27d6-67fa-44f2-8185-78257ea91efa/suspend?

PATCH Headers (9) Body Scripts Settings

Headers (8 hidden)

Key	Value	Description	Bulk Edit	Presets
<input checked="" type="checkbox"/> Authorization	Bearer {{token}}			
Key	Value	Description		

Body Cookies Headers (7) Test Results

200 OK 90 ms 294 B Save Response

1 {
2 "success": true,
3 "message": "Account suspended successfully"
4 }

Online Find and replace Console

Runner Start Proxy Cookies Vault Trash

This screenshot shows the Postman interface for a PATCH request to the 'suspend' endpoint of the AthletIA API. The request URL is `https://{{domain}}/users/d8ea27d6-67fa-44f2-8185-78257ea91efa/suspend?`. The 'Headers' tab is selected with the 'Authorization' header set to `Bearer {{token}}`. The response status is 200 OK, and the response body is a JSON object confirming account suspension.

The screenshot shows the Postman interface with the following details:

- Workspace:** AthletIA
- Request Type:** PATCH
- URL:** {{domain}} /auth/change-password
- Body (JSON):**

```
1 {
2     "accountId": "7e654954-0f67-49a9-9067-7fe9df0bb544",
3     "changePasswordRequest": {
4         "oldPassword": "contrasena123",
5         "newPassword": "contrasenatilinaza"
6     }
7 }
```

- Response Status:** 200 OK
- Response Body (JSON):**

```
1 {
2     "success": true,
3     "message": "Password changed successfully"
4 }
```

The screenshot shows the Postman interface with the following details:

- Workspace:** AthletIA
- Request Type:** GET
- URL:** {{domain}} /profiles/4ee44b49-799c-44c5-96d9-0f9bee9893e5
- Headers:**

Key	Value
Authorization	Bearer {{token}}

- Response Status:** 200 OK
- Response Body (JSON):**

```
1 {
2     "id": "4ee44b49-799c-44c5-96d9-0f9bee9893e5",
3     "name": "David",
4     "birthDate": "1991-05-05T05:00:00.000Z",
5     "phoneNumber": "8888888888",
6     "updatedAt": "2025-10-16T23:41:55.000Z",
7     "gender": "male"
8 }
```

Home Workspaces API Network

AthletIA / Profile / update

GET {{domain}} /profiles/4ee44b49-799c-44c5-96d9-0f9bee9893e5

Params Authorization Headers (9) Body Scripts Settings

Body:

```
1 {  
2   "name": "Pepin"  
3 }
```

200 OK 111 ms 417 B Save Response

Body Cookies Headers (7) Test Results

Body:

```
1 {  
2   "id": "4ee44b49-799c-44c5-96d9-0f9bee9893e5",  
3   "name": "David",  
4   "birthDate": "1991-05-05T05:00:00.000Z",  
5   "phoneNumber": "8888888888",  
6   "updatedat": "2025-10-16T23:41:55.888Z",  
7   "gender": "male"  
8 }
```

Console

Home Workspaces API Network

AthletIA / GetAll

GET {{domain}} /workout/exercises

Params Authorization Headers (6) Body Scripts Settings

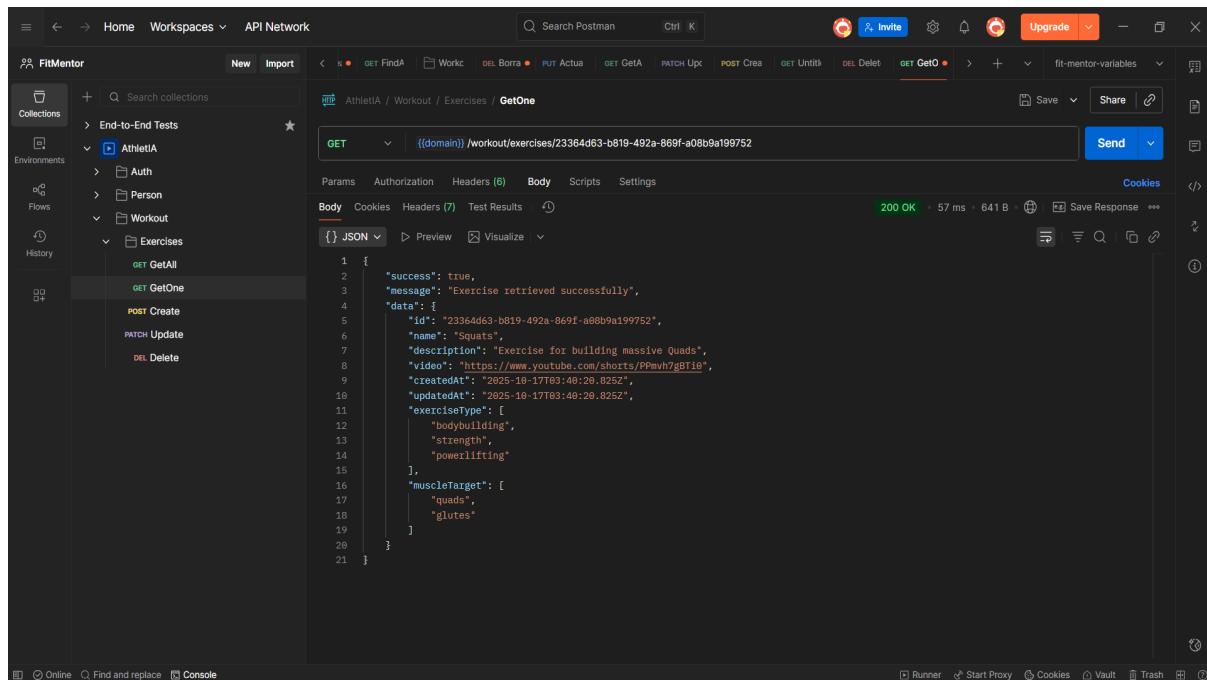
Body Cookies Headers (7) Test Results

Body:

```
1 {  
2   "success": true,  
3   "message": "Exercises retrieved successfully",  
4   "data": [  
5     {  
6       "id": "23364dd63-8819-492a-869f-a88b9a199752",  
7       "name": "Squats",  
8       "description": "Exercise for building massive Quads",  
9       "video": "https://www.youtube.com/shorts/PWmh7gBt18",  
10      "createdAt": "2025-10-17T03:40:20.025Z",  
11      "updatedAt": "2025-10-17T03:40:20.825Z",  
12      "exerciseType": [  
13        "bodybuilding",  
14        "strength",  
15        "powervlifting"  
16      ],  
17      "muscleTarget": [  
18        "quads",  
19        "glutes"  
20      ]  
21    },  
22    {  
23      "id": "c5fc03dc-c36e-46bf-ab61-66b0176a9d72",  
24      "name": "Bench Press",  
25      "description": "Exercise for building massive Pecs",  
26      "video": "https://www.youtube.com/shorts/hW0lk05Ms4",  
27      "createdAt": "2025-10-17T03:40:24.035Z",  
28    }  
]
```

200 OK 5 ms 997 B Save Response

Console



HTTP AthletA / Workout / Exercises / GetOne

```
GET {{domain}} /workout/exercises/23364d63-b819-492a-869f-a08b9a199752
```

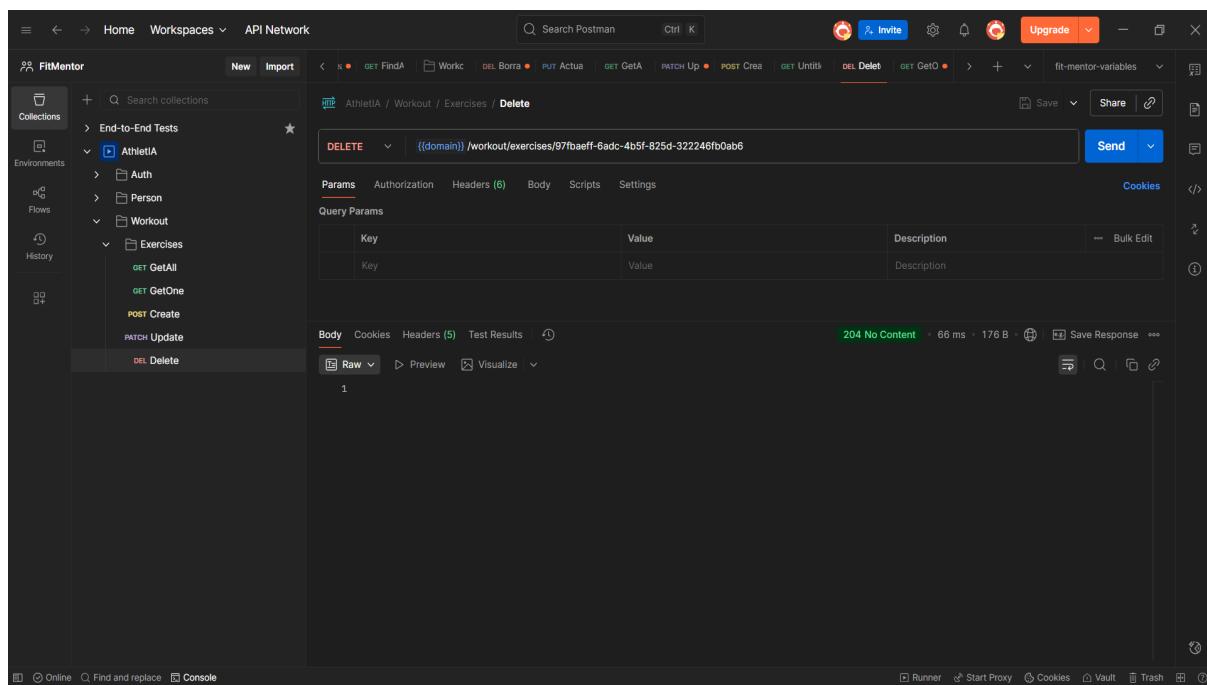
Params Authorization Headers (6) Body Scripts Settings

Body Cookies Headers (7) Test Results

JSON Preview Visualize

```
1 {  
2     "success": true,  
3     "message": "Exercise retrieved successfully",  
4     "data": {  
5         "id": "23364d63-b819-492a-869f-a08b9a199752",  
6         "name": "Squats",  
7         "description": "Exercise for building massive Quads",  
8         "video": "https://www.youtube.com/shorts/PfMvh7gBTI0",  
9         "createdAt": "2025-10-17T03:40:20.825Z",  
10        "updatedAt": "2025-10-17T03:40:20.825Z",  
11        "exerciseType": [  
12            "bodybuilding",  
13            "strength",  
14            "powerlifting"  
15        ],  
16        "muscleTarget": [  
17            "quads",  
18            "glutes"  
19        ]  
20    }  
21 }
```

200 OK 57 ms 641 B Save Response



HTTP AthletA / Workout / Exercises / Delete

```
DELETE {{domain}} /workout/exercises/97baeff-6adc-4b5f-825d-322246fb0ab6
```

Params Authorization Headers (6) Body Scripts Settings

Query Params

Key	Value	Description
Key	Value	Description

Body Cookies Headers (5) Test Results

Raw Preview Visualize

204 No Content 66 ms 176 B Save Response

The screenshot shows the Postman interface with the following details:

- Collection:** FitMentor
- Workspace:** AthletA / Workout / Exercises / Create
- Method:** POST
- URL:** {{domain}} /workout/exercises
- Body (raw JSON):**

```
1 {
2   "name" : "Bench Press",
3   "description" : "Exercise for building massive Pecs",
4   "video" : "https://www.youtube.com/shorts/hWbulkb5Ms4",
5   "muscleTarget" : [
6     "chest", "deltoids", "triceps"
7   ],
8   "exerciseType" : [
9     "bodybuilding", "strength", "powerlifting"
10  ]
11 }
```

- Response Status:** 201 Created
- Response Body (JSON):**

```
1 {
2   "success": true,
3   "message": "Exercise created successfully",
4   "data": {
5     "id": "c5fc03dc-c36e-46bf-ab61-66b0176a9d72",
6     "name": "Bench Press",
7     "description": "Exercise for building massive Pecs",
8     "video": "https://www.youtube.com/shorts/hWbulkb5Ms4",
9     "createdAt": "2025-10-17T03:44:24.035Z",
10    "updatedAt": "2025-10-17T03:44:24.035Z",
11    "exerciseType": [
12      "bodybuilding",
13      "strength"
14    ],
15    "muscleTarget": []
16  }
17 }
```

The screenshot shows the Postman interface with the following details:

- Collection:** FitMentor
- Workspace:** AthletA / Workout / Exercises / Update
- Method:** PATCH
- URL:** {{domain}} /workout/exercises/c5fc03dc-c36e-46bf-ab61-66b0176a9d72
- Body (raw JSON):**

```
1 {
2   "name" : "Bench Press",
3   "description" : "Good exercise for building massive pecs",
4   "video" : "https://www.youtube.com/shorts/hWbulkb5Ms4",
5   "muscleTarget" : [
6     "chest", "deltoids", "triceps"
7   ],
8   "exerciseType" : [
9     "bodybuilding", "strength"
10  ]
11 }
```

- Response Status:** 200 OK
- Response Body (JSON):**

```
1 {
2   "success": true,
3   "message": "Exercise updated successfully",
4   "data": {
5     "id": "c5fc03dc-c36e-46bf-ab61-66b0176a9d72",
6     "name": "Bench Press",
7     "description": "Good exercise for building massive pecs",
8     "video": "https://www.youtube.com/shorts/hWbulkb5Ms4",
9     "createdAt": "2025-10-17T03:44:24.035Z",
10    "updatedAt": "2025-10-17T03:46:07.755Z",
11    "exerciseType": [
12      "bodybuilding",
13      "strength"
14    ],
15    "muscleTarget": []
16  }
17 }
```

The screenshot shows the Postman application interface. On the left, there's a sidebar with 'AthletIA' selected, showing collections like 'Auth', 'Person', 'Workout', and 'End-to-End Tests'. The main area displays a POST request to 'AthletIA / Auth / CompleteProfileSetup'. The 'Body' tab is selected, showing a raw JSON payload:

```
1 {  
2     "accountId": "93eadb8b-1d56-47c3-9269-6bd4711e20e9",  
3     "profileRequest": {  
4         "name": "Ilin",  
5         "birthdate": "08/09/2005",  
6         "phoneNumber": "0000000000",  
7         "gender": "male"  
8     }  
9 }
```

Below the request, the response is shown in a JSON format:

```
1 {  
2     "success": true,  
3     "message": "Profile setup completed successfully"  
4 }
```

The status bar at the bottom indicates a '201 Created' response with a duration of '161 ms' and a size of '305 B'.

Repositorio de GitHub:

<https://github.com/GabrielCevallos/AthletIA>