

Important Links

Public Repositories (they are the same):

GitHub: https://github.com/LewisPrice-Rapturate/Idea_Tracker_API.git

Git.lewispricedev.com: https://git.lewispricedev.com/Rapturate/Idea_Tracker_API.git

Live API on Render: <https://idea-tracker-api.onrender.com/>

Live API documentation: <https://idea-tracker-api.onrender.com/api-docs>

Testing Plan

Authentication Endpoints

POST /api/auth/signup

Access Control: Public

Case 1 – 201 : Enter a new username and password.

- Click **Try it out**
- Expected response:
 - Status: 201 User created successfully
 - JSON: { "id": 4, "username": "new_user_1" }

Case 2 – 400 Error: Use the same username again.

- Click **Try it out**
- Expected response:
 - Status: 400 Invalid input or validation error

Case 3 – 400 Error: Password is too short.

- Click **Try it out**
- Replace the password with anything less than 8 characters
- Click **Execute**
- Expected response:
 - Status: 400 Invalid input or validation error
 - JSON: { "errors": [{ "field": "password", "message": "Password must be at least 8 characters" }] }

Case 4 – 409 Error: Use the same username again.

- Click **Try it out**
- Expected response:
 - Status: 409 Conflict (Username already exists)
 - JSON: { "error": "Username already exists" }

POST /api/auth/login

Access Control: Public

Case 1 – 401 Error: Invalid Credentials

- Click **Try it out**
- Change the password field to something other than the provided value

- Click **Execute**
- Expected response:
 - Status: 401 Invalid Credentials
 - JSON: { "error": "Invalid credentials" }

Case 2 – 200: Valid credentials

- Click **Reset**
- Click **Try it out**
- Click **Execute**
- Expected response:
 - Status: 200 Login successful
 - JSON: { "accessToken": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9..." }

Setup for further tests:

- Copy the access token and then click the big green **Authorize** button at the top of the page.
 - NOTE: Just grab the text within the “eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...”. Do NOT include the quotation marks.
- Paste the access token in the text field and click **Authorize** and then **Close**.

Idea Management

[GET /api/ideas](#)

Access Control: Authenticated User

Case 1 – 200 OK:

- Click **Try it out**
- Click **Execute**
- Expected response:
 - Status: 200 OK
 - JSON:


```
[
  {
    "id": 1,
    "name": "alice_dev Idea 1",
    "description": "Smart Tasking",
    "date_created": "2026-04-28T16:11:24.956Z",
    "userId": 1
  },
  {
    "id": 2,
    "name": "alice_dev Idea 2",
    "description": "Recipe Hub",
    "date_created": "2026-04-28T16:11:24.956Z",
    "userId": 1
  },
  {
    "id": 3,
    "name": "alice_dev Idea 3",
    "description": "Collab Tool",
    "date_created": "2026-04-28T16:11:24.956Z",
    "userId": 1
  }
]
```

Case 2 – 401 Unauthorized: Remove the JWT from the Authorize feature.

- Click the **Authorize** button at the top of the page and then press **logout**
- Click **Try it out**
- Expected response:
 - Status: 401 Unauthorized – Missing or invalid token
 - JSON: { "error": "Not authenticated. Please provide a valid token." }

POST /api/ideas

Access Control: Authenticated User

Case 1 – 201 Idea Created Successfully

- Click **Try it out**
- Click **Execute**
- Expect response:
 - Status: 201
 - JSON:


```
{
  "id": 10,
  "name": "A Task Manager",
  "description": "A web app that manages tasks.",
  "date_created": "2026-04-28T17:45:09.600Z",
  "userId": 1
}
```

Case 2 – 400 Bad Request: Name and Description are both empty.

- Go through the login process again.
- Click **Try it out**
- Delete everything in the request body except for {}.
- Click **Execute**
- Expected response:
 - Status: 400
 - JSON:


```
{ "errors": [{ "field": "name", "message": "Name is required" }, { "field": "description", "message": "Description is required" } ] }
```

Case 3 – 409 Conflict: Idea with that name already exists

- Click **reset**
- Click **Try it out**
- Click **Execute**
- Expected response:
 - Status: 409
 - JSON: { "error": "Idea with name \"A Task Manager\" already exists" }

Case 4 – 401 Unauthorized: Remove the JWT from the Authorize feature.

- Click the **Authorize** button at the top of the page and then press **logout**
- Click **Try it out**
- Expected response:
 - Status: 401 Unauthorized.
 - JSON: { "error": "Not authenticated. Please provide a valid token." }

[PUT /api/ideas/:id](#)

Access Control: Owner of the Idea

Case 1 – 404 Not Found: Use a non-existent positive number for the ID (e.g., 100).

- Click **Try it out**
- Enter **a positive number** into the box for **id**
- Click **Execute**
- Expected response:
 - Status: 404
 - JSON: { "error": "Idea 100 not found" }

Case 2 – 200 OK: Update an Idea

- Click **Try it out**
- Enter **2** into the box for **id**
- Click **Execute**
- Expect response:
 - Status: 200
 - JSON:

```
{
  "id": 2,
  "name": "Updated Idea",
  "description": "Updated Idea description.",
  "date_created": "2026-04-28T18:07:07.644Z",
  "userId": 1
}
```

Case 3 – 400 Bad Request: Use an invalid format for ID (e.g., -10).

- Click **Try it out**
- Enter **a negative number** into the box for **id**
- Click **Execute**
- Expected response:
 - Status: 400
 - JSON: { "errors": [{ "field": "id", "message": "Id must be a positive integer" }] }

Case 4 – 409 Conflict: Idea with that name already exists

- Click **reset**
- Click **Try it out**
- Click **Execute**
- Expected response:
 - Status: 409
 - JSON: { "error": "Idea with name “Updated Idea” already exists" }

Case 5 – 404 Not Found: Try to access an idea that the user does not own.

- Click **Try it out**
- Enter **8** into the box for **id**
- Click **Execute**

- Note that this returns 404 instead of a 401 Unauthorized. This keeps the Project IDs private from user to user by not saying if that ID exists, just that the server cannot find a project id of that value associated with the current user.
- Expected response:
 - Status: 404 Not Found.
 - JSON: { "error": "Idea 8 not found" }

Case 6 – 401 Unauthorized: Missing or invalid token

- Click the **Authorize** button at the top of the page and then press **logout**
- Click **Try it out**
- Expected response:
 - Status: 401 Unauthorized.
 - JSON: { "error": "Not authenticated. Please provide a valid token." }

[DELETE /api/ideas/:id](#)

Access Control: Authenticated User

Case 1 – 204: Idea deleted successfully

- Click **Try it out**
- Enter **2** into the box for **id**
- Click **Execute**
- Expect response:
 - Status: 204

Case 2 – 404 Not Found: No idea with that id belongs to this user

- Click **Try it out**
- Enter **2** into the box for **id**
- Click **Execute**
- Expect response:
 - Status: 404 Not Found.
 - JSON: { "error": "Idea 8 not found" }

Case 3 – 400 Bad Request: Id not a positive integer

- Click **Try it out**
- Enter **a negative number** into the box for **id**
- Click **Execute**
- Expected response:
 - Status: 400
 - JSON: { "errors": [{ "field": "id", "message": "Id must be a positive integer" }] }

Case 4 – 401 Unauthorized: Missing or invalid token

- Click the **Authorize** button at the top of the page and then press **logout**
 - Click **Try it out**
 - Expected response:
 - Status: 401 Unauthorized.
 - JSON: { "error": "Not authenticated. Please provide a valid token." }
-

Project & File Management

[GET /api/projects](#)

Access Control: Authenticated User

Case 1 – 200 OK:

- Click **Try it out**
- Click **Execute**
- Expected response:
 - Status: 200 OK
 - JSON:

```
[
  {
    "id": 1,
    "name": "alice_dev Project 1",
    "description": "Workspace for project 1",
    "date_created": "2026-04-28T16:11:24.956Z"
  },
  {
    "id": 2,
    "name": "alice_dev Project 2",
    "description": "Workspace for project 2",
    "date_created": "2026-04-28T16:11:24.956Z",
    "userId": 1
  },
  {
    "id": 3,
    "name": "alice_dev Project 3",
    "description": "Workspace for project 3",
    "date_created": "2026-04-28T16:11:24.956Z",
    "userId": 1
  }
]
```

Case 2 – 401 Unauthorized: Remove the JWT from the Authorize feature.

- Click the **Authorize** button at the top of the page and then press **logout**
- Click **Try it out**
- Expected response:
 - Status: 401 Unauthorized – Missing or invalid token
 - JSON: { "error": "Not authenticated. Please provide a valid token." }

[POST /api/projects](#)

Access Control: Authenticated User

Case 1 – 201 Project Created Successfully

- Click **Try it out**
- Click **Execute**
- Expect response:
 - Status: 201
 - JSON:

```
{
  "id": 4,
  "name": "Project 4",
  "description": "Project 4 description",
  "date_created": "2026-04-28T19:00:05.863Z",
```

}

Case 2 – 400 Bad Request: Name and Description are both empty.

- Go through the login process again.
- Click **Try it out**
- Delete everything in the request body except for {}.
- Click **Execute**
- Expected response:
 - Status: 400
 - JSON:

```
{ "errors": [{ "field": "name", "message": "Name is required" }, { "field": "description", "message": "Description is required" } ]}
```

Case 3 – 409 Conflict: Project with that name already exists

- Click **reset**
- Click **Try it out**
- Click **Execute**
- Expected response:
 - Status: 409
 - JSON: { "error": "Idea with name \"A Task Manager\" already exists" }

Case 4 – 401 Unauthorized: Remove the JWT from the Authorize feature.

- Click the **Authorize** button at the top of the page and then press **logout**
- Click **Try it out**
- Expected response:
 - Status: 401 Unauthorized.
 - JSON: { "error": "Not authenticated. Please provide a valid token." }

[GET /api/projects/:id](#)

Access Control: Authenticated User

Case 1 – 200 OK:

- Click **Try it out**
- Click **Execute**
- Expected response:
 - Status: 200 OK
 - JSON:

```
{
  "id": 1,
  "name": "alice_dev Project 1",
  "description": "Workspace for project 1",
  "date_created": "2026-04-28T16:11:24.956Z",
  "userId": 1
}
```

Case 2 – 401 Unauthorized: Remove the JWT from the Authorize feature.

- Click the **Authorize** button at the top of the page and then press **logout**
- Click **Try it out**
- Expected response:
 - Status: 401 Unauthorized – Missing or invalid token

- JSON: { "error": "Not authenticated. Please provide a valid token." }

[PUT /api/projects/:id](#)

Access Control: Owner of the Idea

Case 1 – 404 Not Found: Use a non-existent positive number for the ID (e.g., 100).

- Click **Try it out**
- Enter **a positive number** into the box for **id**
- Click **Execute**
- Expected response:
 - Status: 404
 - JSON: { "error": "Idea 100 not found" }

Case 2 – 200 OK: Update an Idea

- Click **Try it out**
- Enter **2** into the box for **id**
- Click **Execute**
- Expect response:
 - Status: 200
 - JSON:

```
{
  "id": 2,
  "name": "alice_dev Updated Project 2",
  "description": "Updated Workspace for project 2",
  "date_created": "2026-04-28T18:07:07.644Z"
}
```

Case 3 – 400 Bad Request: Use an invalid format for ID (e.g., -10).

- Click **Try it out**
- Enter **a negative number** into the box for **id**
- Click **Execute**
- Expected response:
 - Status: 400
 - JSON: { "errors": [{ "field": "id", "message": "Id must be a positive integer" }] }

Case 4 – 409 Conflict: Project with that name already exists

- Click **reset**
- Click **Try it out**
- Click **Execute**
- Expected response:
 - Status: 409
 - JSON: { "error": "Project with name “alice_dev Updated Project 2” already exists" }

Case 5 – 404 Not Found: Try to access a project that the user does not own.

- Click **Try it out**
- Enter **8** into the box for **id**
- Click **Execute**
- Expected response:
 - Status: 404 Not Found.

- JSON: { "error": "Project 8 not found" }

Case 6 – 401 Unauthorized: Missing or invalid token

- Click the **Authorize** button at the top of the page and then press **logout**
- Click **Try it out**
- Expected response:
 - Status: 401 Unauthorized.
 - JSON: { "error": "Not authenticated. Please provide a valid token." }

[DELETE /api/projects/:id](#)

Access Control: Authenticated User

Case 1 – 204: Project deleted successfully

- Click **Try it out**
- Enter **2** into the box for **id**
- Click **Execute**
- Expect response:
 - Status: 204

Case 2 – 404 Not Found: No project with that id belongs to this user

- Click **Try it out**
- Enter **2** into the box for **id**
- Click **Execute**
- Expect response:
 - Status: 404 Not Found.
 - JSON: { "error": "Project 2 not found" }

Case 3 – 400 Bad Request: Id not a positive integer

- Click **Try it out**
- Enter **a negative number** into the box for **id**
- Click **Execute**
- Expected response:
 - Status: 400
 - JSON: { "errors": [{ "field": "id", "message": "Id must be a positive integer" }] }

Case 4 – 401 Unauthorized: Missing or invalid token

- Click the **Authorize** button at the top of the page and then press **logout**
- Click **Try it out**
- Expected response:
 - Status: 401 Unauthorized.
 - JSON: { "error": "Not authenticated. Please provide a valid token." }

[GET /api/projects/:id/files](#)

Access Control: Authenticated User

Case 1 – 200 OK:

- Click **Try it out**

- Click **Execute**
- Expected response:
 - Status: 200 OK
 - JSON:


```
[
    {
      "id": 4,
      "projectId": 2,
      "name": "U1_P2_0_seed_1.docx",
      "size": 14499,
      "mimeType": " application/vnd.openxmlformats-officedocument.wordprocessingml.document"
    },
    {
      "id": 5,
      "projectId": 2,
      "name": "U1_P2_1_seed_2.docx",
      "size": 33930,
      "mimeType": "application/pdf"
    },
    {
      "id": 6,
      "projectId": 2,
      "name": "U1_P2_2_seed_3.docx",
      "size": 276129,
      "mimeType": "image/pdf"
    }
  ]
```

Case 2 – 401 Unauthorized: Remove the JWT from the Authorize feature.

- Click the **Authorize** button at the top of the page and then press **logout**
- Click **Try it out**
- Expected response:
 - Status: 401 Unauthorized – Missing or invalid token
 - JSON: { "error": "Not authenticated. Please provide a valid token." }

[GET /api/projects/:id/files/download](#)

Access Control: Owner of the Project

Case 1 – 200 OK:

- Click **Try it out**
- Click **Execute**
- Expected response:
 - Status: 200 OK
 - Response Body: a link to download the project files as a .zip

Case 2 – 401 Unauthorized: Remove the JWT from the Authorize feature.

- Click the **Authorize** button at the top of the page and then press **logout**
- Click **Try it out**
- Expected response:
 - Status: 401 Unauthorized – Missing or invalid token
 - JSON: { "error": "Not authenticated. Please provide a valid token." }

Case 3 – 400 Bad Request: Id not a positive integer

- Click **Try it out**
- Enter **a negative number** into the box for **id**

- Click **Execute**
- Expected response:
 - Status: 400
 - JSON: { "errors": [{ "field": "id", "message": "Id must be a positive integer" }] }

[POST /api/projects/:id/files](#)

Access Control: Owner of the Project

Case 1 – 201 : Use a project ID you own. Upload a small image or text file.

- Click **Try it out**
- Expect response: 201 Created.

Case 2 – 413 Payload Too Large: Attempt to upload a file larger than 10MB.

- Click **Try it out**
- Expect 413 Payload Too Large.

[DELETE /api/projects/files/:id](#)

Access Control: Authenticated User

Case 1 – 204: Project files deleted successfully

- Click **Try it out**
- Enter **2** into the box for **id**
- Click **Execute**
- Expect response:
 - Status: 204

Case 2 – 404 Not Found: No project with that id belongs to this user

- Click **Try it out**
- Enter **4** into the box for **id**
- Click **Execute**
- Expect response:
 - Status: 404 Not Found.
 - JSON: { "error": "Project 4 not found" }

Case 3 – 400 Bad Request: Id not a positive integer

- Click **Try it out**
- Enter **a negative number** into the box for **id**
- Click **Execute**
- Expected response:
 - Status: 400
 - JSON: { "errors": [{ "field": "id", "message": "Id must be a positive integer" }] }

Case 4 – 401 Unauthorized: Missing or invalid token

- Click the **Authorize** button at the top of the page and then press **logout**
- Click **Try it out**
- Expected response:
 - Status: 401 Unauthorized.
 - JSON: { "error": "Not authenticated. Please provide a valid token." }

Materials

[GET /api/materials](#)

Access Control: Authenticated User

Case 1 – 200 OK:

- Click **Try it out**
- Click **Execute**
- Expected response:
 - Status: 200 OK
 - JSON:

```
{
  "id": 9,
  "name": "Budget Template",
  "description": "Finance",
  "source": "Internal",
  "author": "Dept C",
  "text": "Allocations...",
  "projectId": 3
},
{
  "id": 8,
  "name": "API Docs",
  "description": "Specs",
  "source": "Dev Portal",
  "author": "Team B",
  "text": "Technical data...",
  "projectId": 3
},
{
  ...
}
]
```

Case 2 – 401 Unauthorized: Remove the JWT from the Authorize feature.

- Click the **Authorize** button at the top of the page and then press **logout**
- Click **Try it out**
- Expected response:
 - Status: 401 Unauthorized – Missing or invalid token
 - JSON: { "error": "Not authenticated. Please provide a valid token." }

[GET /api/materials/:projectId](#)

Access Control: Authenticated User

Case 1 – 200 OK:

- Click **Try it out**
- Enter **1** into the box for **id**
- Click **Execute**
- Expected response:
 - Status: 200 OK
 - JSON:

```
[
{
  "id": 28,
  "name": "Resource 1",
  "description": "Resource 1 description",
```

```

        "source": "Source 1",
        "author": "Author 1",
        "text": "Resource 1 text",
        "projectId": 1
    },
    {
        "id": 3,
        "name": "Budget Template",
        "description": "Finance",
        "source": "Internal",
        "author": "Dept C",
        "text": "Allocations...",
        "projectId": 1
    },
    {
        ...
    }
]

```

Case 2 – 400 Bad Request: Id not a positive integer

- Click **Try it out**
- Enter **a negative number** into the box for **id**
- Click **Execute**
- Expected response:
 - Status: 400
 - JSON: { "errors": [{ "field": "id", "message": "Id must be a positive integer" }] }

Case 3 – 401 Unauthorized: Missing or invalid token

- Click the **Authorize** button at the top of the page and then press **logout**
- Click **Try it out**
- Expected response:
 - Status: 401 Unauthorized.
 - JSON: { "error": "Not authenticated. Please provide a valid token." }

[POST /api/materials](#)

Access Control: Authenticated User

Case 1 – 201: Material Created Successfully

- Click **Try it out**
- Click **Execute**
- Expect response:
 - Status: 201
 - JSON:


```

{
  "id": 28,
  "name": "Resource 1",
  "description": "Resource 1 description",
  "source": "Source 1",
  "author": "Author 1",
  "text": "Resource 1 text",
  "projectId": 1
}
                    
```

Case 2 – 400 Bad Request: Name and Description are both empty.

- Go through the login process again.
- Click **Try it out**
- Delete everything in the request body except for {}.
- Click **Execute**
- Expected response:

```

    ○ Status: 400
    ○ JSON:
      {
        "errors": [
          {
            "field": "projectId",
            "message": "ProjectId is required"
          },
          {
            "field": "name",
            "message": "Name is required"
          },
          {
            "field": "description",
            "message": "Description is required"
          },
          {
            "field": "source",
            "message": "Source is required"
          },
          {
            "field": "author",
            "message": "Author is required"
          },
          {
            "field": "text",
            "message": "Text is required"
          }
        ]
      }
    }

```

Case 3 – 409 Conflict: Project with that name already exists

- Click **reset**
- Click **Try it out**
- Click **Execute**
- Expected response:
 - Status: 409
 - JSON: { "error": "Idea with name \"A Task Manager\" already exists" }

Case 4 – 401 Unauthorized: Remove the JWT from the Authorize feature.

- Click the **Authorize** button at the top of the page and then press **logout**
- Click **Try it out**
- Expected response:
 - Status: 401 Unauthorized.
 - JSON: { "error": "Not authenticated. Please provide a valid token." }

[DELETE /api/materials/:id](#)

Access Control: Authenticated User

Case 1 – 204: Material deleted successfully

- Click **Try it out**
- Enter **2** into the box for **id**

- Click **Execute**
- Expect response:
 - Status: 204

Case 2 – 404 Not Found: No material with that id belongs to this user

- Click **Try it out**
- Enter **2** into the box for **id**
- Click **Execute**
- Expect response:
 - Status: 404 Not Found.
 - JSON: { "error": "Project 4 not found" }

Case 3 – 400 Bad Request: Id not a positive integer

- Click **Try it out**
- Enter **a negative number** into the box for **id**
- Click **Execute**
- Expected response:
 - Status: 400
 - JSON: { "errors": [{ "field": "id", "message": "Id must be a positive integer" }] }

Case 4 – 401 Unauthorized: Missing or invalid token

- Click the **Authorize** button at the top of the page and then press **logout**
- Click **Try it out**
- Expected response:
 - Status: 401 Unauthorized.
 - JSON: { "error": "Not authenticated. Please provide a valid token." }

Global Features

[GET /api/ideas or /api/projects or /api/materials](#)

Access Control: Only files associated with the currently authenticated user

Parameters: Set limit=1 and offset=0.

- Click **Try it out**
- Expect response: 200 OK with only one item in the list.

Search Case: Enter a keyword into the search parameter that exists in one of your titles.

- Click **Try it out**
- Expect response: Only items containing that keyword.