

# Note Taking

## Json Files

- users.json
  - {
    - "id":1,
    - "name": "name",
    - "email": "example@test.com",
    - "password": "password"
  - }
- notes.json
  - {
    - "id": 1,
    - "text": "text"
  - }
- user-note.json
  - {
    - "id": 1,
    - "userId": 2,
    - "noteId": 1
  - }
- authorized.json:
  - {
    - "id": 1
    - "userId": 2
    - "email": "example@test.com"
  - }

! Ավելացրեք users.json ֆայլում երկու user

## Cases

- user-ը պետք է ունենա՝
  - login լինելու (sign in) լինելու հնարավորություն

- իր `note`-երը տեսնելու, փոխելու, ավելացնելու և ջնջելու հնարավորություն (CRUD)
- **!** `login` փուլը անցնելուց հետո `authorized.json` ֆայլում ավելացրեք այդ `user`-ի `email`-ը և `id`-ն
- **!** `user`-ը պետք է աշխատի միայն իրան համապատասխանող `note`-երի ցանկի հետ (կազմակերպեք դա `user-notes.json`-ի և `authorized.json`-ի միջոցով)
- **!** Եթե `user`-ը `authorized` չի վերադարձնել `error` (`notes`-ին դիմելիս)

## Scenarios

- **Get Notes:**
  - Method: `GET`
  - Endpoint: `/notes`
- **User Scenario**
  - The user wants to retrieve a list of all their notes.
  - They send a `GET` request to `/notes`.
  - The server responds with a JSON array containing all the notes.
- **Create Note**
  - Method: `POST`
  - Endpoint: `/notes`
  - Request Body: JSON object with a text field containing the note's content.
- **User Scenario:**
  - The user wants to create a new note.
  - They send a `POST` request to `/notes` with the note content in the request body.
  - The server creates the note and responds with a success message.
- **Edit Note**
  - Method: `PUT`
  - Endpoint: `/notes/<note_id>` (where `<note_id>` is the ID of the note to update)

- Request Body: JSON object with a text field containing the updated note content.
- User Scenario:
  - The user wants to update an existing note.
  - They send a PUT request to /notes/<note\_id> with the updated content in the request body.
  - The server updates the note and responds with a success message.
  
- Delete Note
  - Method: DELETE
  - Endpoint: /notes/<note\_id> (where <note\_id> is the ID of the note to delete)
- User Scenario:
  - The user wants to delete an existing note.
  - They send a DELETE request to /notes/<note\_id>.
  - The server deletes the note and responds with a success message.

