

DB

- Users

- Id CHAR(36) NOT NULL PRIMARY KEY,
- Nickname VARCHAR(255),
- Email VARCHAR(255) UNIQUE NOT NULL,
- PasswordHash VARCHAR(255) NOT NULL,
- Image LONGBLOB

- Projects

- Id CHAR(36) NOT NULL PRIMARY KEY,
- UserId CHAR(36) NOT NULL,
- Name VARCHAR(255) NOT NULL,
- TimeStamp TIMESTAMP NOT NULL,
- FOREIGN KEY (UserId) REFERENCES Users (Id)

- Tabs

- Id CHAR(36) NOT NULL PRIMARY KEY,
- ProjectId CHAR(36) NOT NULL,
- UserId CHAR(36) NOT NULL,
- Name VARCHAR(255) NOT NULL,
- TimeStamp TIMESTAMP NOT NULL,
- FOREIGN KEY (ProjectId) REFERENCES Projects (Id),
- FOREIGN KEY (UserId) REFERENCES Users (Id)

- Tiles

- Id CHAR(36) NOT NULL PRIMARY KEY,
- TabId CHAR(36) NOT NULL,
- UserId CHAR(36) NOT NULL,
- Name VARCHAR(255) NOT NULL,
- X SMALLINT UNSIGNED NOT NULL,
- Y SMALLINT UNSIGNED NOT NULL,
- H SMALLINT UNSIGNED NOT NULL,
- W SMALLINT UNSIGNED NOT NULL,
- TimeStamp TIMESTAMP NOT NULL,
- FOREIGN KEY (TabId) REFERENCES Tabs (Id),
- FOREIGN KEY (UserId) REFERENCES Users (Id)

- TilesItems

- Id CHAR(36) NOT NULL PRIMARY KEY,
- TileId CHAR(36) NOT NULL,
- UserId CHAR(36) NOT NULL,
- Content VARCHAR(255) NOT NULL,
- Type VARCHAR(255) NOT NULL,
- TimeStamp TIMESTAMP NOT NULL,
- FOREIGN KEY (TileId) REFERENCES Tiles (Id),
- FOREIGN KEY (UserId) REFERENCES Users (Id)

Controllers

<https://localhost:7023/> или <http://localhost:5087> пропущено.

Далее будет в таком виде: `api/<Controller>/<Method>`

Полный путь будет выглядеть так: <https://localhost:7023/api/<Controller>/<Method>>

Если нужна авторизация, нужно в header добавить поле `Authorization: Bearer <JWT токен>`

Users

1. **GET** `api/Users/all` – получение всех пользователей в БД.
 - a. **Тело запроса:** отсутствует.
 - b. **Авторизация:** не нужна.
 - c. **Ответ (при коде 200):** массив со всеми пользователями с всей доступной информацией. Доступно только при разработке.
 - d. **Другие ответы:** 500 код – ошибка сервера.
 - e. **Пример ответа:**

Code	Details
200	<div>Response body</div> <pre>[{ "id": "b575ad32-5925-41a0-9391-01b0b87ddccf", "nickname": "string", "email": "user@example.com", "passwordHash": "\$2a\$10\$Nun1vUXvYF8GzcjdGI9vo053F3HX/y885PWrZmA2fmNLCF10ygn20", "image": null }, { "id": "c53d6f50-5259-47d7-9f8f-4cb31e1bc4df", "nickname": "string", "email": "user2@example.com", "passwordHash": "\$2a\$10\$s9zo7F3mV9/P3mLn1ELn6emyXUX8/gcRNLo3bhgq/Q/PvfM/IIIGa", "image": null }]</pre>

3. **POST** api/Users/register - регистрация

a. **Тело запроса:** объект с полями:

- i. “nickname” (string)
- ii. “email” (string)
- iii. “password” (string)
- iv. “confirmPassword” (string)
- v. “image” (byte[]).
- vi. **“nickname” и “image” – не обязательно.**

b. **Авторизация:** не нужна.

c. **Ответ (при коде 200):** JWT токен для авторизации.

d. **Другие ответы:** 400 код (объект с полем “message” (string)) – ошибка запроса (Почта уже используется либо другие ошибки). 500 код – ошибка сервера. Также возможны ошибки валидации полей самим ASP.NET.

e. **Пример ответа:**

Code	Details
200	<div>Response body</div> <div>eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJodHRwOi8vc2NoZW1hcy54bWxzbn2FwLm9yZy93cy8yMDA1LzA1L2lkZW50aXR5L2NsYW1tcy91bmFpbGFnKZHZ1c3MiOiJ1c2VyMTIzQGmcvd3MvMjAwNS8wNS9pZGVudG10eS9jbGFnbnXvbmFtZSI6InN0cm1uZyIsIm5iZiI6MTY5MjI5MjQ5NiwiZXhwIjoxNjkyMjkzMDk2LCJpc3MiOiJ3ZWJhcGkiLCJhdWQiOiJyZWJjdGFW</div>

4. **PUT** api/Users – обновление информации о пользователе

a. **Тело запроса:** объект с полями:

- i. “nickname” (string)
- ii. “email” (string)
- iii. “password” (string)
- iv. “confirmPassword” (string)
- v. “image” (byte[]).

vi. Каждое поле не обязательно. Можно отправить, например, только “email” или другие поля. Пароль обязательно меняется с двумя полями. После каждого успешного обновления нужно заново получить JWT токен!!!

b. **Авторизация:** нужна.

c. **Ответ (при коде 200):** объект с полем “message” (string).

d. **Другие ответы:** 400 код (объект с полем “message” (string)) – ошибка запроса (Почта уже используется либо другие ошибки). 500 код – ошибка сервера. Так же возможны ошибки валидации полей самим ASP.NET.

e. **Пример ответа:**

Code	Details
200	<div>Response body</div> <pre>{ "message": "User updated" }</pre>
400 <i>Undocumented</i>	<div>Error: response status is 400</div> <div>Response body</div> <pre>{ "message": "Model was empty" }</pre>

5. **DELETE** api/Users – удаление аккаунта

- a. **Тело запроса:** отсутствует.
- b. **Авторизация:** нужна.
- c. **Ответ (при коде 200):** объект с полем “message” (string).
- d. **Другие ответы:** 400 код (объект с полем “message” (string)) – ошибка запроса (Не найден пользователь. Возможно если JWT токен не обновлен.). 500 код – ошибка сервера.
- e. **Пример ответа:**

Code	Details
200	<div>Response body</div> <div><pre>{ "message": "User deleted" }</pre></div>
400	<div>Error: response status is 400</div> <div>Response body</div> <div><pre>{ "message": "User not found" }</pre></div>

JWT токен хранит в себе Email пользователя и, если есть, Nickname.

Projects

1. **GET** api/Projects/all – получение всех проектов в БД.
 - a. **Тело запроса:** отсутствует.
 - b. **Авторизация:** не нужна.
 - c. **Ответ (при коде 200):** массив со всеми проектами с всей доступной информацией. Доступно только при разработке.
 - d. **Другие ответы:** 500 код – ошибка сервера.
 - e. **Пример ответа:**

Code	Details
200	<div>Response body</div> <pre>[{ "id": "341d3c39-0a73-422b-b9f2-80b056774116", "userId": "c53d6f50-5259-47d7-9f8f-4cb31e1bc4df", "name": "2proj", "timeStamp": "2023-08-17T14:16:47+03:00" }, { "id": "4ce9c4fb-0d19-4ed5-ba74-342cdb7a1149", "userId": "c53d6f50-5259-47d7-9f8f-4cb31e1bc4df", "name": "2proj2", "timeStamp": "2023-08-17T14:16:49+03:00" }, { "id": "5df0a309-de61-4435-a4f6-8a8e150315a6", "userId": "b575ad32-5925-41a0-9391-01b0b87ddccf", "name": "proj1", "timeStamp": "2023-08-17T14:15:10+03:00" }]</pre>

2. **GET** api/Projects – получение всех проектов пользователя.

- a. **Тело запроса:** отсутствует.
- b. **Авторизация:** нужна.
- c. **Ответ (при коде 200):** массив объектов с полями:
 - i. "id" (string)
 - ii. "name" (string)
 - iii. "timeStamp" (datetime)
 - iv. "isMyProject" (bool) (здесь всегда true)
- d. **Другие ответы:** 400 код (объект с полем "message" (string)) – ошибка запроса (Не найден пользователь. Возможно если JWT токен не обновлен.). 500 код – ошибка сервера.
- e. **Пример ответа:**

Code	Details
200	<div>Response body</div> <pre>[{ "id": "5df0a309-de61-4435-a4f6-8a8e150315a6", "name": "proj1", "timeStamp": "2023-08-17T14:15:10+03:00", "isMyProject": true }]</pre>

3. **POST** api/Projects – создание проекта

a. **Тело запроса:** объект с полями:

i. “name” (string)

b. **Авторизация:** нужна.

c. **Ответ (при коде 200):** объект с полем “message” (string).

d. **Другие ответы:** 400 код (объект с полем “message” (string)) – ошибка запроса (Не найден пользователь. Возможно если JWT токен не обновлен.). 500 код – ошибка сервера. Так же возможны ошибки валидации полей самим ASP.NET.

e. **Пример ответа:**

Code	Details
200	<div>Response body</div> <pre>{ "message": "Project created" }</pre>

4. **GET** api/Projects/<Id> - получение проекта по Id.

- a. **Тело запроса:** отсутствует.
- b. **Авторизация:** нужна.
- c. **Ответ (при коде 200):** объект с полями:
 - i. "id" (string)
 - ii. "name" (string)
 - iii. "timeStamp" (datetime)
 - iv. "isMyProject" (bool) (false когда проект другого пользователя)
- d. **Другие ответы:** 400 код (объект с полем "message" (string)) – ошибка запроса (Не найден пользователь. (Возможно если JWT токен не обновлен.) Не найден проект.). 500 код – ошибка сервера.
- e. **Пример ответа:**

Request URL

```
https://localhost:7023/api/Projects/b842d566-3808-4783-a88e-050acb2ad6cd
```

Server response

Code	Details
200	<p>Response body</p> <pre>{ "id": "b842d566-3808-4783-a88e-050acb2ad6cd", "name": "test", "timeStamp": "2023-08-18T14:11:20+03:00", "isMyProject": true }</pre>

Code	Details
200	<p>Response body</p> <pre>{ "id": "341d3c39-0a73-422b-b9f2-80b056774116", "name": "2proj", "timeStamp": "2023-08-17T14:16:47+03:00", "isMyProject": false }</pre>

5. **PUT** api/Projects/<Id> - обновление проекта по Id.

- a. **Тело запроса:** объект с полем “name” (string).
- b. **Авторизация:** нужна.
- c. **Ответ (при коде 200):** объект с полем “message” (string)
- d. **Другие ответы:** 400 код (объект с полем “message” (string)) – ошибка запроса (Не найден пользователь (Возможно если JWT токен не обновлен.). Не найден проект. Проект другого пользователя.). 500 код – ошибка сервера.

e. **Пример ответа:**

Code	Details
200	<div>Response body</div> <div><pre>{ "message": "Project updated" }</pre></div>

Code	Details
400 <i>Undocumented</i>	<div>Error: response status is 400</div> <div>Response body</div> <div><pre>{ "message": "This is not your project." }</pre></div>

Code	Details
400 <i>Undocumented</i>	<div>Error: response status is 400</div> <div>Response body</div> <div><pre>{ "message": "Project not found" }</pre></div>

Code	Details
400 <i>Undocumented</i>	<div>Error: response status is 400</div> <div>Response body</div> <div><pre>{ "message": "Model was empty" }</pre></div>

6. **DELETE** api/Projects/<Id> - удаление проекта по <Id>.

- a. **Тело запроса:** отсутствует.
- b. **Авторизация:** нужна.
- c. **Ответ (при коде 200):** объект с полем “message” (string)
- d. **Другие ответы:** 400 код (объект с полем “message” (string)) – ошибка запроса (Не найден пользователь (Возможно если JWT токен не обновлен.). Не найден проект. Проект другого пользователя.). 500 код – ошибка сервера.
- e. **Пример ответа:**

Code	Details
200	<div>Response body</div> <div><pre>{ "message": "Project deleted" }</pre></div>

Code	Details
400 <i>Undocumented</i>	<div>Error: response status is 400</div> <div>Response body</div> <div><pre>{ "message": "This is not your project." }</pre></div>

Code	Details
400 <i>Undocumented</i>	<div>Error: response status is 400</div> <div>Response body</div> <div><pre>{ "message": "Project not found" }</pre></div>