

# 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>

2. **POST** api/Users/login – вход

a. **Тело запроса:** объект с полями:

- i. “email” (string)
- ii. “password” (string).

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 (Неверная почта или пароль). 500 код – ошибка сервера. Так же возможны ошибки валидации полей самим ASP.NET.

e. **Пример ответа:**

Code	Details
200	<div>Response body</div> <div>eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJodHRwOi8vc2NoZW1hcy54bWxzb2FwLn9yZy93cy8yMDA1LzA1L21kZW50aXR5L2NsYWItdy91bWVpY29tIiwiaHR0cDovL3NjaGVtYX03MvMjAwNS8wNS9pZGVudG10eS9jbGZpbXQvbmFtZSI6InN0cm1uZyIsIm5iZiI6MTY5MjI5MjM0M0MwIiwiaWF0IjoxNjY5MjI5MjM0M0MwLCJpc3MiOiJ3ZWJhcGkiLCJhdWQiOiJyZWJkdGFwcCJ9. c035aRHdVCYKjhQcRKE3i3Z1W8mBCC6</div>

400	<div>Error: response status is 400</div> <div>Response body</div> <div>{   "message": "User not found" }</div>
-----	--

400	<div>Error: response status is 400</div> <div>Response body</div> <div>{   "message": "Invalid password" }</div>
-----	--

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

a. **Тело запроса:** объект с полями:

- i. “nickname” (string)
- ii. “email” (string)
- iii. “password” (string) (минимальная длина 6 символов)
- iv. “confirmPassword” (string)
- v. “image” (byte[]).
- vi. “nickname” и “image” – необязательные поля.

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.eyJodHRwOi8vc2NoZW1hcy54bWxzbnZwLm9yZy93cy8yMDA1LzA1L2lkZW50aXR5L2NsYW1tcy91bmFpbGFnKZHZ1c3MiOiJ1c2VyMTIzQGmcvd3MvMjAwNS8wNS9pZGVudG10eS9jbGFpbXQvbmFtZSI6InN0cm1uZyIsIm5iZiI6MTY5MjI5MjQ5NiwiZm91dXhwIjoibG9ja3MkZDk2LjJpc3MiOiJ3ZWJhcGkiLCJhdWQiOiJyZWJjdGFw</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 является необязательным. 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>Undocumented 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>

# Tabs

1. **GET** api/Tabs/all – получение всех страниц проектов в БД.
  - a. **Тело запроса:** отсутствует.
  - b. **Авторизация:** не нужна.
  - c. **Ответ (при коде 200):** массив со всеми страницами 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": "4de2427e-95da-49e4-ba6e-842462af1598",     "projectId": "341d3c39-0a73-422b-b9f2-80b056774116",     "userId": "c53d6f50-5259-47d7-9f8f-4cb31e1bc4df",     "name": "tab1",     "timeStamp": "2023-08-17T14:23:23+03:00"   },   {     "id": "60eab929-386e-405e-89fa-ced94ef997bc",     "projectId": "341d3c39-0a73-422b-b9f2-80b056774116",     "userId": "c53d6f50-5259-47d7-9f8f-4cb31e1bc4df",     "name": "tab2",     "timeStamp": "2023-08-17T14:23:32+03:00"   },   {     "id": "ab471d96-ad25-4201-9966-07318751b416",     "projectId": "5df0a309-de61-4435-a4f6-8a8e150315a6",     "userId": "b575ad32-5925-41a0-9391-01b0b87ddccf",     "name": "tab2",     "timeStamp": "2023-08-17T14:25:34+03:00"   } ]</pre>

2. **GET** api/Tabs – получение всех страниц указанного проекта.

- a. **Тело запроса:** отсутствует.
- b. **Параметры:** “projectId” (string) – Id проекта.
- c. **Авторизация:** нужна.
- d. **Ответ (при коде 200):** массив объектов с полями:
  - i. “id” (string)
  - ii. “name” (string)
  - iii. “timeStamp” (datetime)
- e. **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.
- f. **Пример ответа:**

Request URL

`https://localhost:7023/api/Tabs?projectId=341d3c39-0a73-422b-b9f2-80b056774116`

Server response

Code	Details
200	<div>Response body</div> <pre>[   {     "id": "4de2427e-95da-49e4-ba6e-842462af1598",     "name": "tab1",     "timeStamp": "2023-08-17T14:23:23+03:00"   },   {     "id": "60eab929-386e-405e-89fa-ced94ef997bc",     "name": "tab2",     "timeStamp": "2023-08-17T14:23:32+03:00"   } ]</pre>

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

### 3. **POST** api/Tabs – создание страницы

- a. **Тело запроса:** объект с полями:
  - i. “projectId” (string)
  - ii. “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. Так же возможны ошибки валидации полей самим ASP.NET.
- e. **Пример ответа:**

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

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

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

4. **GET** `api/Tabs/<Id>` - получение страницы 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)
- d. **Другие ответы:** 400 код (объект с полем `"message"` (string)) – ошибка запроса (Не найден пользователь. (Возможно если JWT токен не обновлен.) Не найдена страница.). 500 код – ошибка сервера.
- e. **Пример ответа:**

Request URL	
<code>https://localhost:7023/api/Tabs/a8ee2d17-54f2-4353-8ba1-145bb119062b</code>	
Server response	
Code	Details
200	<div>Response body</div> <pre>{  "id": "a8ee2d17-54f2-4353-8ba1-145bb119062b",  "name": "test_create",  "timeStamp": "2023-08-19T12:28:38+03:00"}</pre>
400	<div>Error: response status is 400</div> <div>Response body</div> <pre>{  "message": "Tab not found"}</pre>



5. **PUT** api/Tabs/<Id> - обновление страницы 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 найдена страница. Страница другого пользователя.). 500 код – ошибка сервера.
- e. **Пример ответа:**

Code	Details
200	<div>Response body</div> <div><pre>{   "message": "Tab 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 tab." }</pre></div>

Code	Details
400 <i>Undocumented</i>	<div>Error: response status is 400</div> <div>Response body</div> <div><pre>{   "message": "Tab 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/Tabs/<Id> - удаление страницы 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 найдена страница. Страница другого пользователя.). 500 код – ошибка сервера.
- e. **Пример ответа:**

Code	Details
200	<div>Response body</div> <div><pre>{   "message": "Tab 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 tab." }</pre></div>

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

# Tiles

1. **GET** api/Tiles/all – получение всех тайлов (блоков на странице) страниц в БД.
  - a. **Тело запроса:** отсутствует.
  - b. **Авторизация:** не нужна.
  - c. **Ответ (при коде 200):** массив со всеми блоками 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": "16cd0526-f0ad-4851-b4f8-d4564dedf1f3",     "tabId": "ab471d96-ad25-4201-9966-07318751b416",     "userId": "b575ad32-5925-41a0-9391-01b0b87ddccf",     "name": "tile2",     "x": 0,     "y": 0,     "h": 0,     "w": 0,     "timeStamp": "2023-08-17T14:35:21+03:00"   },   {     "id": "ee6b048d-8352-45f9-ad53-5205f40e6cd9",     "tabId": "ab471d96-ad25-4201-9966-07318751b416",     "userId": "b575ad32-5925-41a0-9391-01b0b87ddccf",     "name": "tile1",     "x": 0,     "y": 0,     "h": 0,     "w": 0,     "timeStamp": "2023-08-17T14:35:18+03:00"   } ]</pre>

2. **GET** api/Tiles – получение всех блоков указанной страницы.

- a. **Тело запроса:** отсутствует.
- b. **Параметры:** “tabId” (string) – Id страницы.
- c. **Авторизация:** нужна.
- d. **Ответ (при коде 200):** массив объектов с полями:
  - i. “id” (string)
  - ii. “name” (string)
  - iii. “x” (ushort)
  - iv. “y” (ushort)
  - v. “h” (ushort)
  - vi. “w” (ushort)
  - vii. “timeStamp” (datetime)
- e. **Другие ответы:** 400 код (объект с полем “message” (string)) – ошибка запроса (Не найден пользователь (Возможно если JWT токен не обновлен). Не найдена страница). 500 код – ошибка сервера.
- f. **Пример ответа:**

Request URL

```
https://localhost:7023/api/Tiles?tabId=ab471d96-ad25-4201-9966-07318751b416
```

Server response

Code	Details
------	---------

200

Response body

```
[
  {
    "id": "16cd0526-f0ad-4851-b4f8-d4564dedf1f3",
    "name": "tile2",
    "x": 0,
    "y": 0,
    "h": 0,
    "w": 0,
    "timeStamp": "2023-08-17T14:35:21+03:00"
  },
  {
    "id": "ee6b048d-8352-45f9-ad53-5205f40e6cd9",
    "name": "tile1",
    "x": 0,
    "y": 0,
    "h": 0,
    "w": 0,
    "timeStamp": "2023-08-17T14:35:18+03:00"
  }
]
```

Code	Details
------	---------

400

Undocumented

Error: response status is 400

Response body

```
{
  "message": "Tab not found"
}
```

### 3. **POST** api/Tiles – создание блока

a. **Тело запроса:** объект с полями:

- i. "tabId" (string)
- ii. "name" (string)
- iii. "x" (ushort)
- iv. "y" (ushort)
- v. "h" (ushort)
- vi. "w" (ushort)

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 найдена страница. Страница другого пользователя.). 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": "Tile created" }</pre>

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

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

4. **GET** api/Tiles/<Id> - получение блока по Id.
- a. **Тело запроса:** отсутствует.
  - b. **Авторизация:** нужна.
  - c. **Ответ (при коде 200):** объект с полями:
    - i. "id" (string)
    - ii. "name" (string)
    - iii. "x" (ushort)
    - iv. "y" (ushort)
    - v. "h" (ushort)
    - vi. "w" (ushort)
    - vii. "timeStamp" (datetime)
  - d. **Другие ответы:** 400 код (объект с полем "message" (string)) – ошибка запроса (Не найден пользователь. (Возможно если JWT токен не обновлен.) Не найден блок.). 500 код – ошибка сервера.
  - e. **Пример ответа:**

Request URL	
<code>https://localhost:7023/api/Tiles/16cd0526-f0ad-4851-b4f8-d4564dedf1f3</code>	
Server response	
Code	Details
200	<div>Response body</div> <pre>{   "id": "16cd0526-f0ad-4851-b4f8-d4564dedf1f3",   "name": "tile2",   "x": 0,   "y": 0,   "h": 0,   "w": 0,   "timeStamp": "2023-08-17T14:35:21+03:00" }</pre>

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

5. **PUT** api/Tiles/<Id> - обновление блока по Id.

a. **Тело запроса:** объект с полями:

i. "name" (string)

ii. "x" (ushort)

iii. "y" (ushort)

iv. "h" (ushort)

v. "w" (ushort)

vi. **Каждое поле является необязательным.**

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 блок. Блок другого пользователя.). 500 код – ошибка сервера.

e. **Пример ответа:**

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

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

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

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

6. **DELETE** api/Tiles/<Id> - удаление блока 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 блок. Блок другого пользователя.). 500 код – ошибка сервера.
- e. **Пример ответа:**

Code	Details
200	<div>Response body</div> <div><pre>{   "message": "Tile 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 tile." }</pre></div>

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



# Tiles Items

1. **GET** `api/TilesItems/GetTypes` – получение всех доступных типов объектов блока в БД.
  - a. **Тело запроса:** отсутствует.
  - b. **Авторизация:** не нужна.
  - c. **Ответ (при коде 200):** массив строк.
  - d. **Другие ответы:** 500 код – ошибка сервера.
  - e. **Пример ответа:**

Code	Details
200	<div>Response body</div> <pre>[   "Note",   "Task" ]</pre>

2. **GET** api/TilesItems/all – получение всех объектов блоков в БД.

- a. **Тело запроса:** отсутствует.
- b. **Авторизация:** не нужна.
- c. **Ответ (при коде 200):** массив со всеми объектами 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": "29e44f8f-fbab-43fc-b2d2-e859789e4996",     "tileId": "ee6b048d-8352-45f9-ad53-5205f40e6cd9",     "userId": "b575ad32-5925-41a0-9391-01b0b87ddccf",     "content": "tileitem1",     "type": "Note",     "timeStamp": "2023-08-17T14:40:48+03:00"   },   {     "id": "d98887a9-a586-44cd-96ce-cda1138e11ea",     "tileId": "ee6b048d-8352-45f9-ad53-5205f40e6cd9",     "userId": "b575ad32-5925-41a0-9391-01b0b87ddccf",     "content": "tileitem2",     "type": "Task",     "timeStamp": "2023-08-17T14:41:16+03:00"   } ]</pre>

3. **GET** api/TilesItems – получение всех объектов указанного блока.

- a. **Тело запроса:** отсутствует.
- b. **Параметры:** “tileId” (string) – Id блока.
- c. **Авторизация:** нужна.
- d. **Ответ (при коде 200):** массив объектов с полями:
  - i. “id” (string)
  - ii. “content” (string)
  - iii. “type” (string)
  - iv. “timeStamp” (datetime)
- e. **Другие ответы:** 400 код (объект с полем “message” (string)) – ошибка запроса (Не найден пользователь (Возможно если JWT токен не обновлен). Не найден блок.). 500 код – ошибка сервера.
- f. **Пример ответа:**

Request URL

```
https://localhost:7023/api/TilesItems?tileId=ee6b048d-8352-45f9-ad53-5205f40e6cd9
```

Server response

Code

Details

200

Response body

```
[
  {
    "id": "29e44f8f-fbab-43fc-b2d2-e859789e4996",
    "content": "tileitem1",
    "type": "Note",
    "timeStamp": "2023-08-17T14:40:48+03:00"
  },
  {
    "id": "d98887a9-a586-44cd-96ce-cda1138e11ea",
    "content": "tileitem2",
    "type": "Task",
    "timeStamp": "2023-08-17T14:41:16+03:00"
  }
]
```

Code

Details

400

Undocumented

Error: response status is 400

Response body

```
{
  "message": "Tile not found"
}
```

#### 4. **POST** api/TilesItems – создание объекта

a. **Тело запроса:** объект с полями:

- i. “tileId” (string)
- ii. “content” (string)
- iii. “type” (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 блок. Блок другого пользователя.). 500 код – ошибка сервера. Так же возможны ошибки валидации полей самим ASP.NET.

e. **Пример ответа:**

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

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

5. **GET** `api/TilesItems/<Id>` - получение объекта по Id.

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

Request URL	
<code>https://localhost:7023/api/TilesItems/42765d79-a6c3-4d44-a031-f13c82f36a7c</code>	
Server response	
Code	Details
200	<div>Response body</div> <pre>{   "id": "42765d79-a6c3-4d44-a031-f13c82f36a7c",   "content": "item",   "type": "Task",   "timeStamp": "2023-08-24T14:00:35+03:00" }</pre>
400	<div>Error: response status is 400</div> <div>Response body</div> <pre>{   "message": "Tile Item not found" }</pre>

6. **PUT** api/TilesItems/<Id> - обновление объекта по Id.

a. **Тело запроса:** объект с полями:

i. "content" (string)

ii. "type" (string)

iii. **Каждое поле является необязательным.**

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 объект. Объект другого пользователя.). 500 код – ошибка сервера.

e. **Пример ответа:**

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

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

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

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

7. **DELETE** api/TilesItems/<Id> - удаление объекта 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 объект. Объект другого пользователя.). 500 код – ошибка сервера.
- e. **Пример ответа:**

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

Code	Details
400 <i>Undocumented</i>	<div>Error: response status is 400</div> <div>Response body</div> <div><pre>{   "message": "Tile Item not found" }</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 tile item." }</pre></div>