Realtime Collaborative Mapping

Mapdraw is a web application that allows users to place markers, lines, and polygons on a map - simultaneously and in real time.

Open source and free to use. The source code is available on GitHub. Built with Leaflet, Socket.io, Node.js, Express, Alpine.js, and sakura.css.

Full code available at www.github.com/mabafaba/mapdraw

Features

User Management

New users can register with a username and password. Existing users can log in with their credentials.

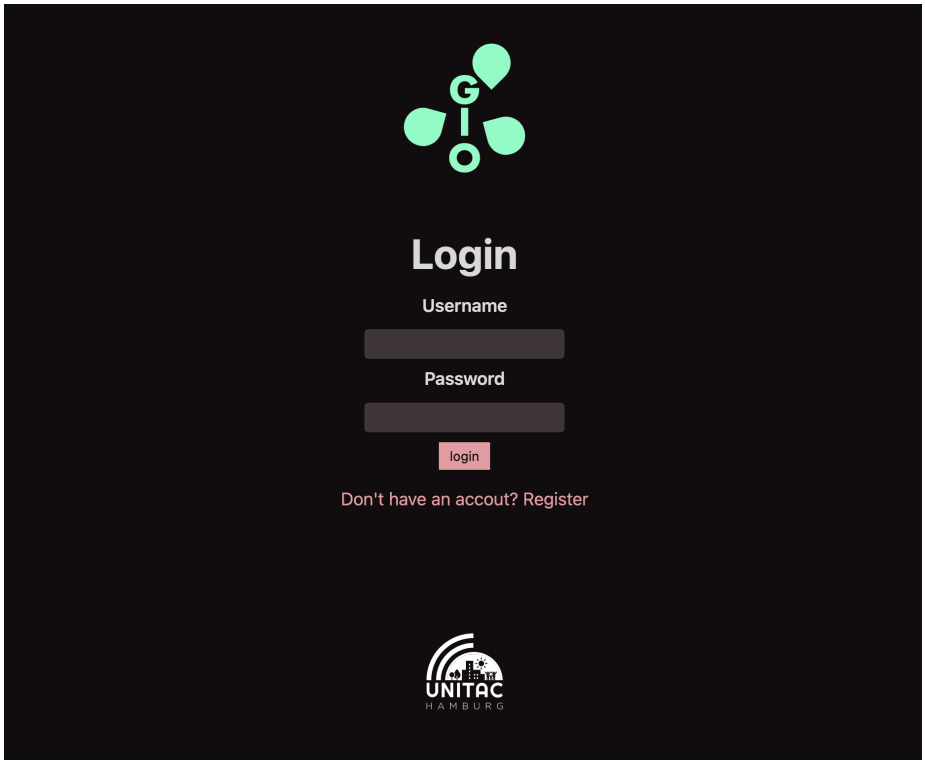
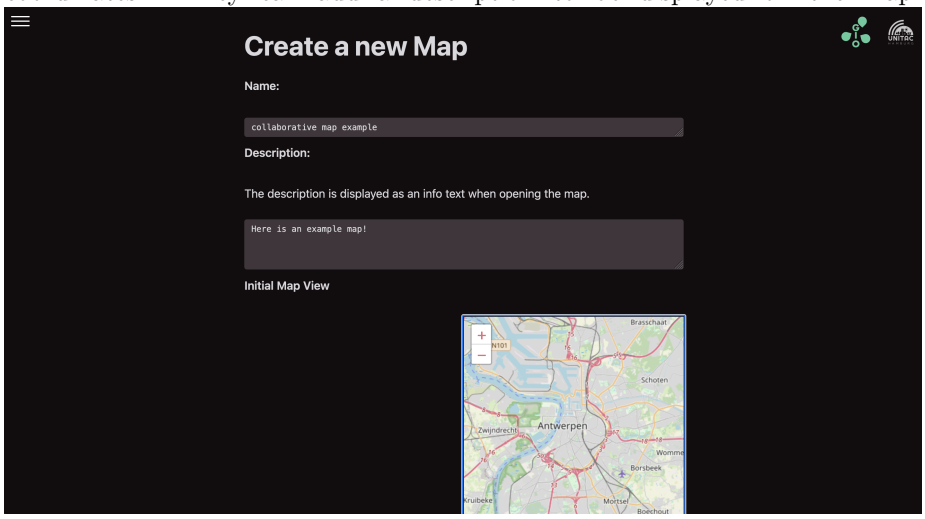


Figure 1: Login Form

Creating a new map

Users can create a new map by clicking on the “New Map” button. They can choose a name for the map and set the initial zoom level and center coordinates. They can add a description to be displayed on the map.



Adding Geometries

All users can then add markers, lines, and polygons to the map. They can also edit and delete existing markers, lines, and polygons. All changes are saved in real time and are visible to all users.

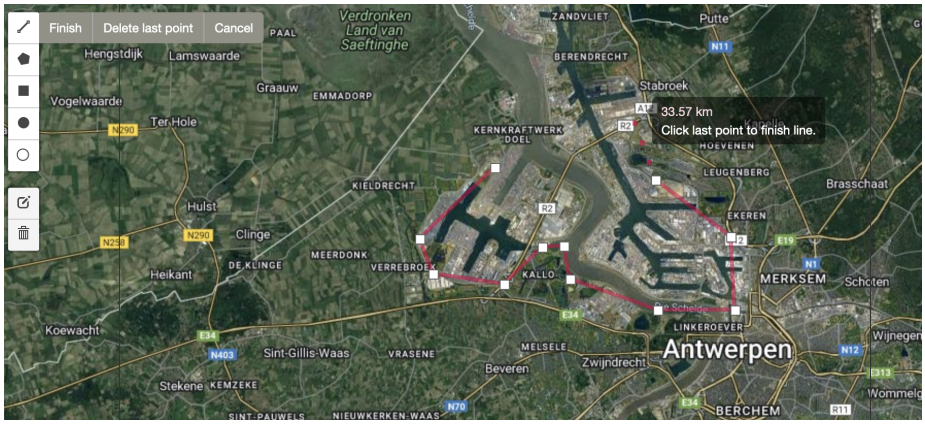


Figure 2: Adding Geometry

Geometry Properties

Users can add properties to markers, lines, and polygons. Properties can include a name and a description.

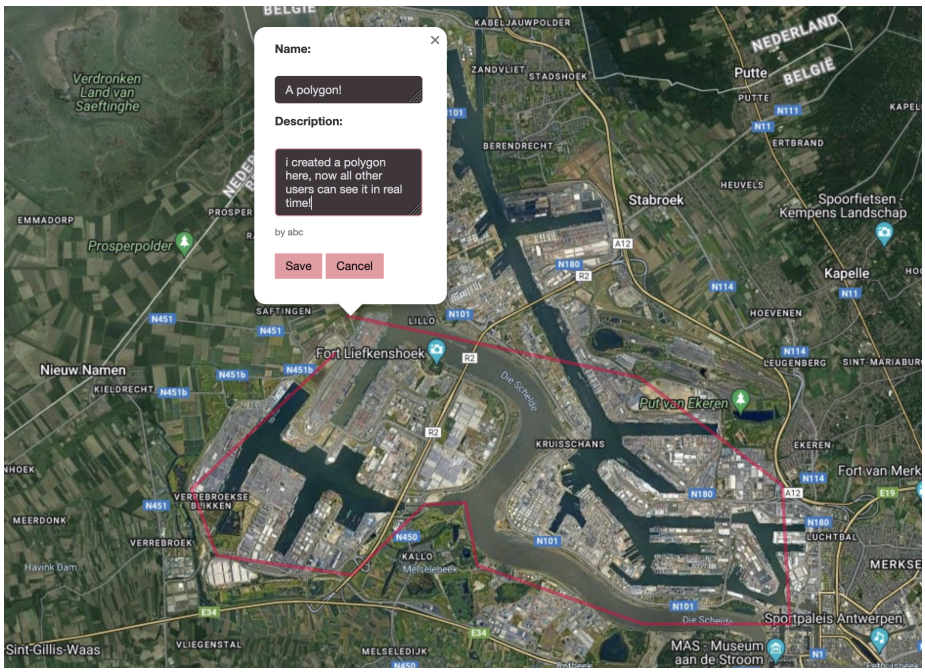


Figure 3: Adding Properties

Sharing a map

Each map has a unique URL that can be shared with other users. Users can access the map by entering the URL in their browser.

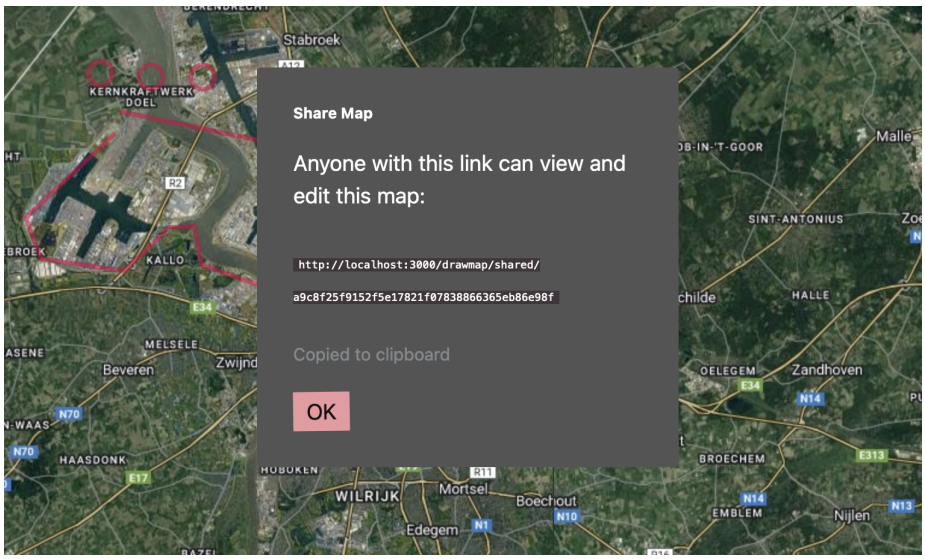


Figure 4: Map sharing

Export Data

Users can export the map data as a GeoJSON file. The file contains all the markers, lines, and polygons on the map, along with their names and descriptions.

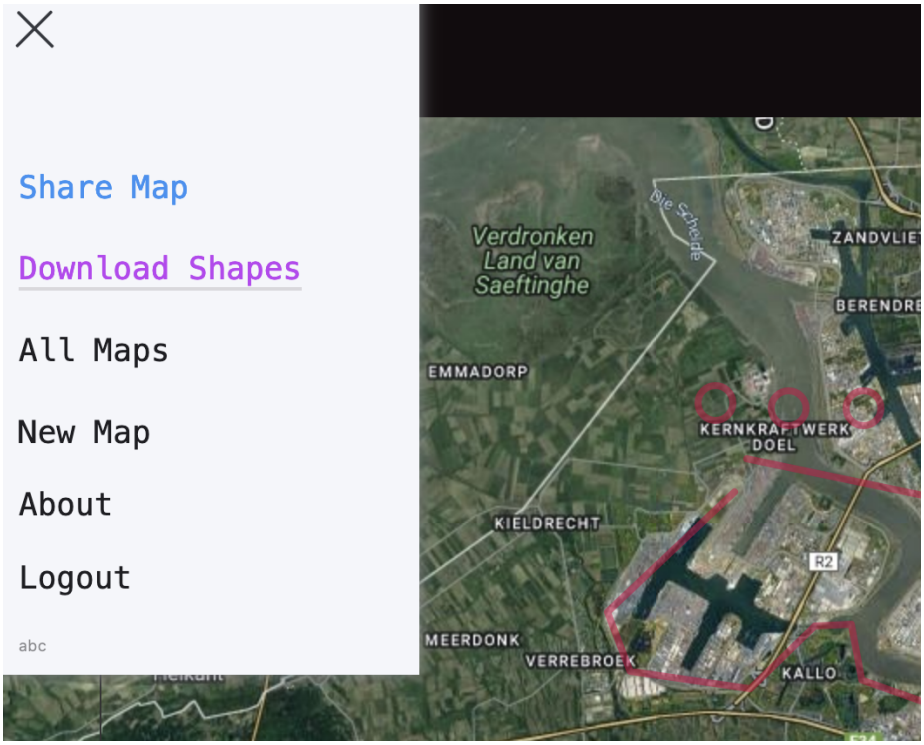


Figure 5: Data Export

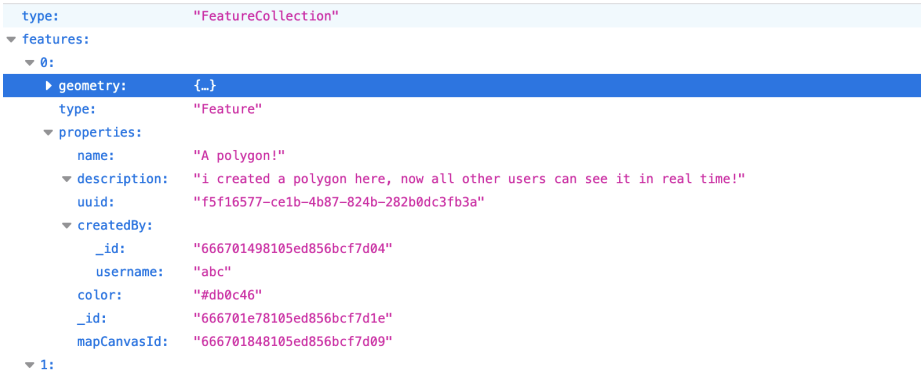


Figure 6: Exported GeoJson File

Import Data

Users can import a GeoJSON file to add markers, lines, and polygons to the map. The file must be in geojson format.

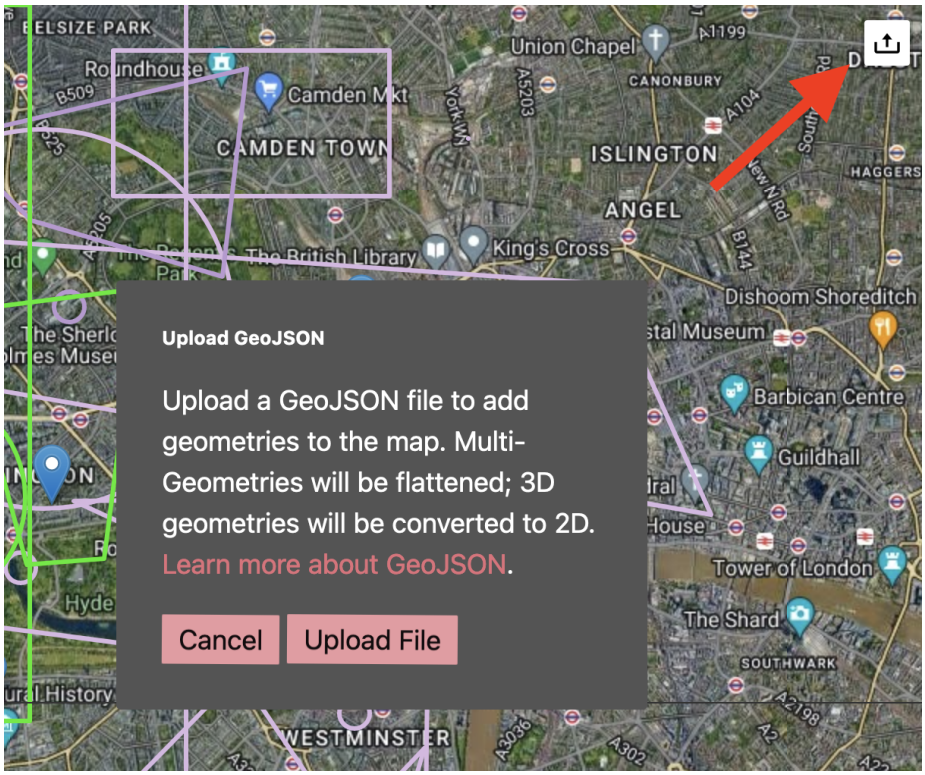


Figure 7: Data Import

Development Details

Installation

Via Docker

prerequisites

- Docker Installed
- Docker Compose Installed

If you have node/npm installed:

```
git clone https://github.com/mabafaba/mapdraw
cd drawmap
npm run docker
```

Without node/npm installation:

```
git clone https://github.com/mabafaba/mapdraw
cd drawmap
docker build -t drawmap .
docker-compose up
browser http://localhost:3000
```

Manually

Prerequisites

- Node.js
- npm
- Git
- MongoDB

Install MongoDB

See MongoDB installation instructions.

Install node app

```
git clone https://github.com/mabafaba/mapdraw
cd drawmap
npm install
```

Set environment variables

Generate a secret key for JWT authentication. Run the following command in the terminal:

```
node -e "console.log(require('crypto').randomBytes(256).toString('base64'));"
```

Create a file named `.env` in the root directory of the project with the following content:

```
JWT_SECRET=secret
```

Where `secret` is the secret key generated in the previous step.

Usage

For production:

```
npm install
npm run start
```

For development:

```
npm install
npm run dev
```

App runs at localhost:3000 by default.

NGINX configuration

```
location /drawmap {
        proxy_pass http://localhost:3000/drawmap;
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection 'upgrade';
        proxy_set_header Host $host;
        proxy_cache_bypass $http_upgrade;
}

location /drawmap-socket-io {
        proxy_pass http://localhost:3000/drawmap-socket-io;
        proxy_http_version 1.1;
```

```
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection 'upgrade';
        proxy_set_header Host $host;
        proxy_cache_bypass $http_upgrade;
}
```

Stack

- Server: Node.js, Express, Socket.io
- Database: MongoDB
- Client state management: Alpine.js
- Map: Leaflet
- Styling: sakura.css