

TradeTerra

Fullstack course project overview

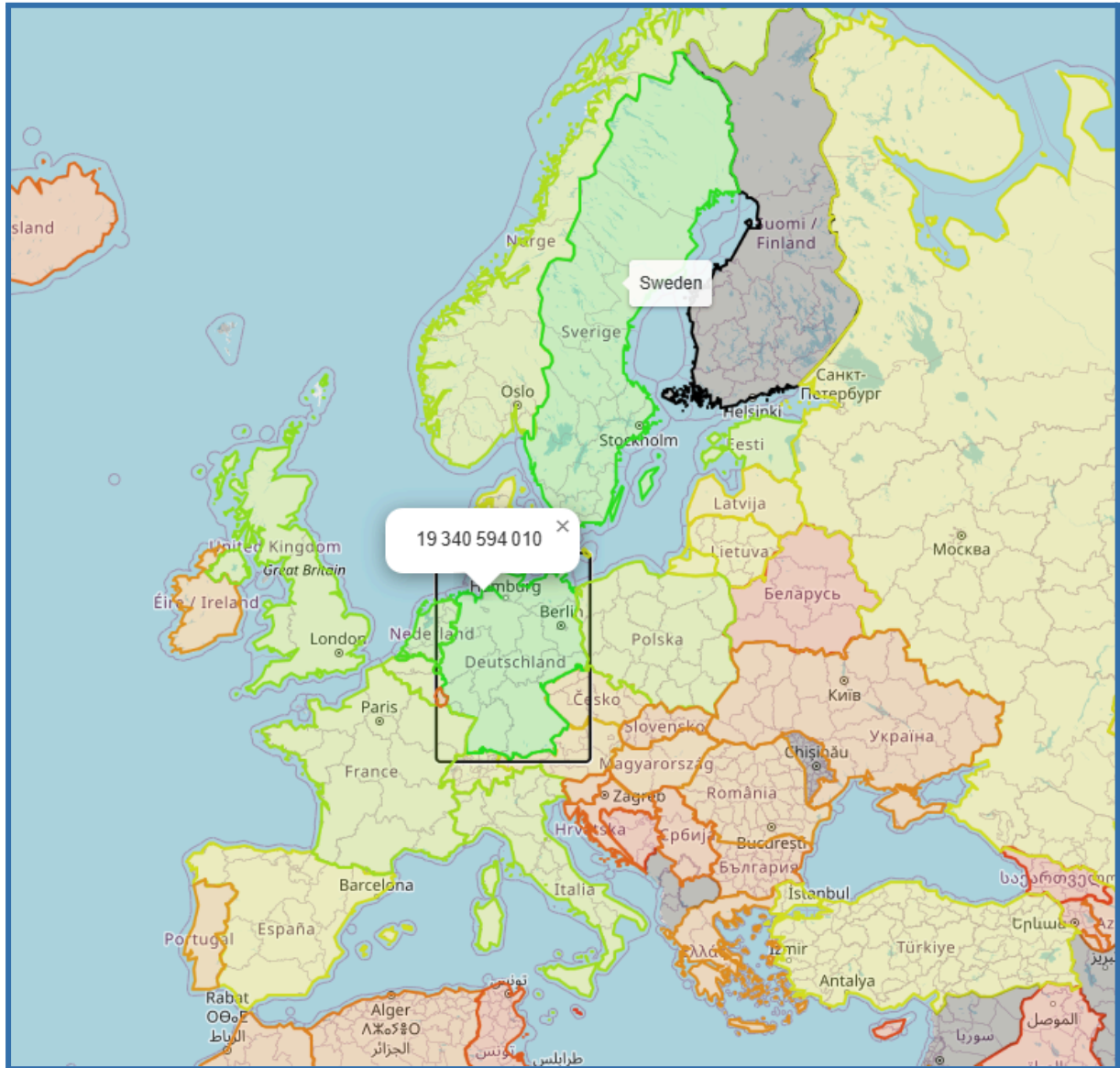


Figure 1. Map visualization prototype

Weronika Klatka

weronika.klatka@student.lut.fi

github.com/Wirlaa

Description

Summary

Global trade statistics visualization webapp with educational gaming elements.

Rationale

The goal is to build a simpler, easier to understand and more educational version of the Observatory of Economic Complexity website (oec.world/en). That website contains lots of trade statistics from around the world, but it is difficult to understand without expert knowledge and generally feels overwhelming to read. Additionally, it features already popular games like Tradle, but they have some shortcomings that I'd like to avoid. My idea is to create a more casual and fun version of that site that encourages curiosity and serves educational value.

Detailed description

The webapp will have two main parts: the visualization and the gamified education. The visualization will include an interactive world map and various charts, while the gamified part will include at least one simple game and user accounts for tracking progress.

Tech stack

Main language

I'll use **JavaScript**, even though I believe TypeScript would generally be better long-term. I have little experience with both and I don't want to be bogged down with specific syntax.

Front end

I've already developed a demo using the **Leaflet** framework for map visualization. I may use a similar small framework for charts, i.e. **Frappe**. I have a bit of experience with **React** and **Bootstrap**, so I plan to use either of them for the polished user interface version, but so far I've been using plain **JavaScript** and **HTML**. I may use **Phaser** for game implementation.

Back end

As I have no experience with building backend for a website, I'll use **Node.js** as suggested in the lecture.

Database

Similarly, due to lack of experience, I'll go for **MongoDB** as suggested in the lecture.

IDE

I'm a big fan of JetBrains products, so I'll be using **WebStorm** IDE.

Definition of done

The webapp contains:

- an interactive map that shows imports and exports for a selected country
- at least three different charts showing interesting trade data
- interpolation of at least one chart
- at least one simple game
- user account database for storing game scores

It is primarily a desktop application, but mobile support would be nice to have.

Timeline

Since I work better when given a rough timeline, I thought I'd include it. This doesn't mean I'll strictly follow through, it's just something that I can use to motivate myself and later refer to.

01.09 - 19.10

Take part in Web Programming course to learn the basics of front-end

13.10

Build a prototype

14.10 (today)

Create project outline

19.10

Complete data visualization part (lines up with Web Programming deadline)

24.10

Follow the tutorial and set up backend with user accounts

31.10

Create simple game(s)

07.11

Make the UI user friendly

14.11, 21.11, 28.11

Extra weeks in case I fall behind schedule

03.12

Final deadline