Geofeelings

Anthony Madhvani

January 16, 2016

Contents

Build instructions

```
npm install
npm start
```

Links

Deployment: young-dusk-9003.herokuapp

Repository: octagonal/Geofeelings

Deployment

A simple `git push heroku master` should suffice. Please create a new issue if this is not the case for you.

Default users

Username	Password
ikbeneendocent	test123

The MongoDB & Admin credentials will be given during the presentation.

Performance

- Consistently 60FPS
- Network load is reduced to a minimum by using very efficient MongoDB calls

Task division

Name	BE	FE	Task in team
Anthony Madhvani	Y	N	Backend, frontend

Workload

In hours

Estimated

```
                    Map view:  5.00
                    Timeline:  4.00
Authentication & authorization:  3.00
               Statistics view:  5.00
               Entry filtering:  4.00
                 Activity chat:  5.00
```

Actual

```
                    Map view:  10.00
```

```
                     Timeline:                          8.00
Authentication & authorization:          4.00
               Statistics view:                               10.00
               Entry filtering:          4.00
                 Activity chat:                 6.00
```

Technology

This list is not exhaustive. Refer to `/package.json` for every package used.

Frontend

Name	Usage
Jade	Non-interactive views
React	Stateful, interactive views
Sentiment	Calculating sentiment
Moment	Date calculations
lodash	Throttling function calls, filtering & mapping the entry list
d3	Showing graphs

Backend

Name	Why
qs	Parsing querystrings
Passport	Helps with auth
Node.js	Mandatory
connect-flash	Flash important messages
express-session	Manage sessions

Conclusion

Possible enhancements

- Additional filtering
- Graphic the amount of tags
- Showing the trending tags

Difficulties

- JavaScript

Being used to strongly typed languages, coming to grips with JavaScripts dynamism was quite difficult at first. It requires a great deal of discipline to keep code structured since the compiler won't do it for you.

- React

React does not play along nicely with most "classic" UI libraries such as Jquery or Bootstrap.js, especially when those libraries update state as well. Some nasty hacks were required.

Successes

- Isomorphic UI

Rendering on both the server and in the browser was largely a success. This is a substantial improvement over my previous projects since the UI has the best of both worlds: caching, Google indexing, interactivity, etc.

- Map & Chat

The main views of the project were (IMO) implemented very well. State updates instantly across all components, chat works as expected and the timeline is really snappy.

Personal remarks

By doing this project I came to the realization I would've gladly taken on the FE module as well. Nevertheless, it was a great experience for me because it was my first real attempt at making a (non-tutorial level) project with a JS based stack. If I had to take a guess I would say that the technological skills I learned by doing this project will definitely come in handy in my career.

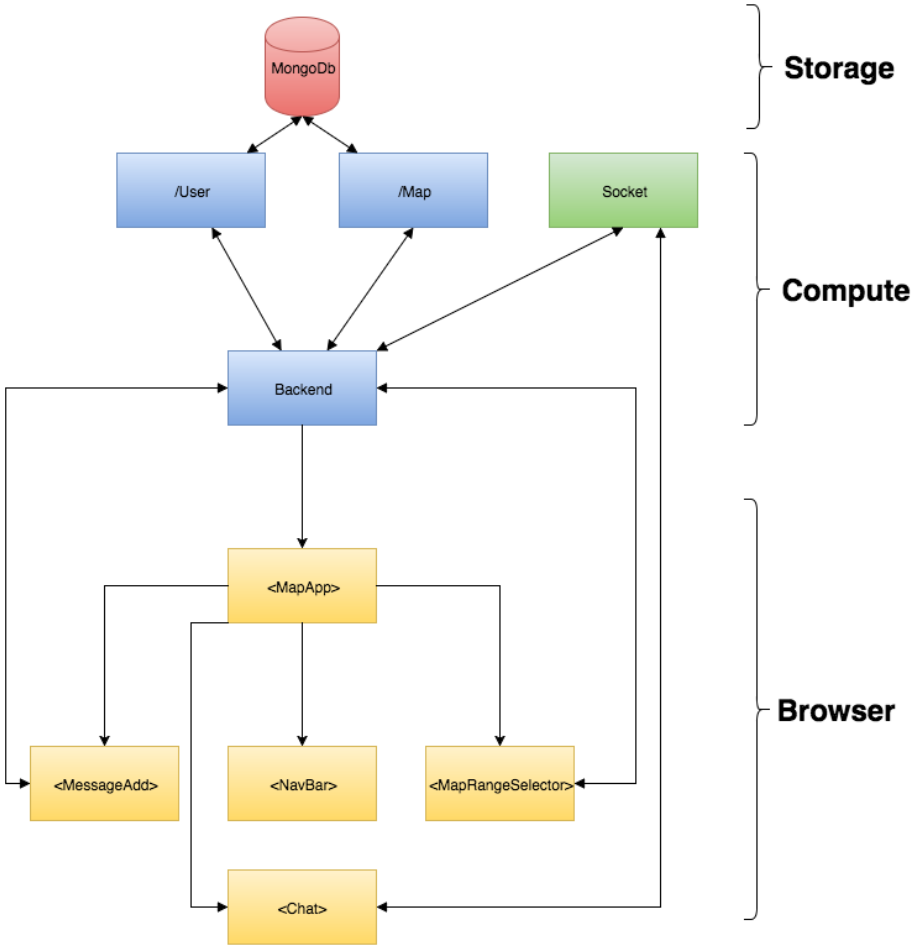

Figure 1: Architecture