

Building Mathhub using React

Bachelor Thesis

Johannes-Sebastian See
Supervisor: Michael Kohlhase
Co-supervisor: Tom Wiesing
Friedrich-Alexander University, Erlangen Nürnberg, Germany
January 18, 2019

Contents

1	Introduction	1
1.1	Mathhub	1
1.2	Drupal Attack	1
1.3	Building an interactive Frontend - State of the art	2
2	React	2
2.1	Core concept	2
2.2	Building new components	3
3	The Architecture of Mathhub	4
3.1	Mathhub Routes	4
3.2	Mathhub Library API	5
4	Mathhub Components	6
4.1	Libraries	6
4.2	Archives	6
4.3	Modules: Theories and Views	6
4.4	Statistics	6
5	The Applications of Mathhub	6
5.1	Glossary	6
5.2	Math Dictionary	6
6	Communication with the backend	6
7	Conclusion	6
8	Future Work	6
8.1	TGView	6
8.2	MathWebSearch	6
8.3	Subset Frontends	6
8.4	Issue report: Mathhub and content	6
9	Sources	6

1 Introduction

1.1 Mathhub

1.2 Drupal Attack

Up until April 2018 Mathhub was build with Drupal. Drupal is an open source content-management framework used by millions of different websites. But in April 2018 a critical security flaw in the versions 6 to 8 went

public. The problem was that the Drupal core in these versions accepts request parameters without any validation. This means the core processes any input from anybody. To exploit this weakness an attacker doesn't even need to log in or have any other privileges on a vulnerable website. With this flaw it is possible to inject malicious code and compromise a website in multiple ways. This can be used to access, change and delete private data and create backdoors to make future attacks possible. The Drupal community called this weakness "Drupalgeddon2" while its official name was "CVE-2018-7600". Some code that was injected installed the program XM-Rig Monero miner, which is a cryptocurrency mining program, as well as deleting other mining programs on the compromised system. The National Institute of Standards (NIST) and Technology gave Drupal a "Highly Critical" Rating because of this vulnerability. After this flaw was discovered a patch was published and a warning to update every website that used a vulnerable version was given.

Since this was not the first detected flaw in Drupal the decision to stop using it and rebuild Mathhub from the ground up to not be affected by future attacks was made.

1.3 Building an interactive Frontend - State of the art

2 React

2.1 Core concept

React is an open source JavaScript library owned and maintained by Facebook. It was created to build interactive user interfaces (UI). For example it is used for Facebook and Instagram. What makes React unique is its use of a virtual Document Object Model (DOM). The concept of the virtual DOM is that when updating a website not everything is rendered again. React computes the differences between the last and the next page and only changes the necessary parts. On top of that it has conditional rendering which means that an item will only be rendered if it is shown. The advantage of virtual and conditional rendering is that this makes updating a website fast, but it comes with high RAM costs. The actual interface is made up of many different elements and components. Since a website that uses React can have many different features it is helpful to build new components.

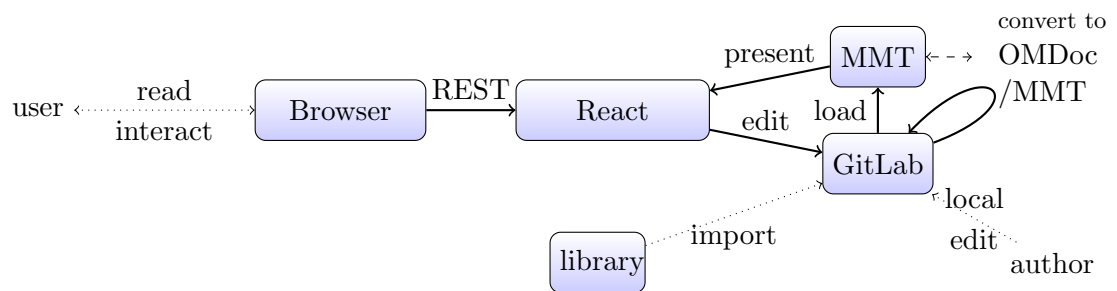
- React is a JavaScript Library owned and used by Facebook and Instagram
- open source
- build the frontend (UIs)

- on change: compute and only render changes (virtual DOM - Document Object Model)
- conditional rendering
- UI is made up of multiple elements / components
- make own components
- (We use JSX)

2.2 Building new components

React already has a large library with a lot of different components, but it is often necessary to make new ones that have the desired functionality. In JavaScript new components can be implemented by creating either a function or a class. Their input variables are called props and can only be read. Components return React elements that are ready to be rendered. Naturally a component can grow big rather quickly. Luckily it is possible to use components inside other components. This comes with the advantage that they can be reused in many different locations. The difference between creating a new component as a function and as a class is that a class can have a private internal state, which can be updated an any time. Since props are read-only, updating the state can only affect lower components. If it is necessary to also change something in a higher component it is possible to "lift up" the state. This means adding the state that causes the change to the state of the component on a higher level and giving it back to the lower levels as a prop. If the update should affect a component on the same level creating a new component with that state that consist of all the one that are affected will make this possible.

3 The Architecture of Mathhub

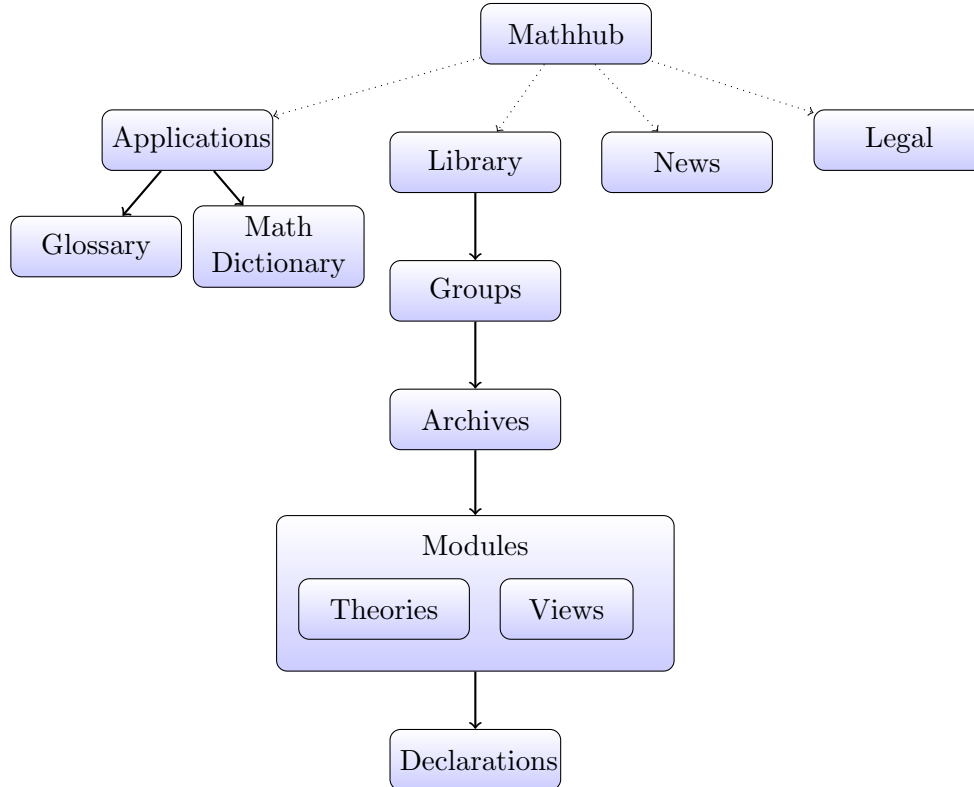


1

¹EdNOTE: look for author and user pictures

EdN:1

3.1 Mathhub Routes



Routes are used to navigate through the many different uses of Mathhub. Every application and every component has its own route. At the top of the Mathhub frontend there is a menu which contains the links to home under "Mathhub". The math dictionary and the glossary can be accessed through the dropdown menu "Applications". The "News" tab has the route to the news. The dropdown menu "Help" has three external links "Documentation", which leads to the the "Mathhub Documentation wiki" on github.com, "Browse Sources", where the different project of the KWARC group can be found and "Contact a Human" to write an E-Mail to the KWARC staff. To learn more about Mathhub itself the tab "About" links to "About Mathhub.info" in the Mathhub wiki.

At the bottom of the frontend the routes to the Imprint, Privacy Policy and Licenses can be found.

The most interesting part are the library routes. Every step in the library hierarchy is its own route. Starting with the different groups to their archives with their modules, which can be either theories or views.

3.2 Mathhub Library API

The library is build with `IApiObjectItems`. An `IApiObjectItem` consists of an ID, a name, a parent if one exists and statistics if available. Further

attributes are dependent on its kind. The possible kinds are group, archive, document, opaque, module, declaration, component and tag.² Since most objects have many children it would be unnecessary to load every information for every child at once. To reduce the cost for loading a page there exists a smaller version of every object called a reference. These reference objects only have the necessary information like ID, name, parent, statistics and in some cases a short teaser of the content. For example when a group is opened it only loads a list of references to the archives contained in this group instead of the complete archives. EdN:2

4 Mathhub Components

4.1 Libraries

4.2 Archives

4.3 Modules: Theories and Views

4.4 Statistics

5 The Applications of Mathhub

5.1 Glossary

5.2 Math Dictionary

6 Communication with the backend

7 Conclusion

Don't forget Notes³

EdN:3

8 Future Work

8.1 TGView

8.2 MathWebSearch

8.3 Subset Frontends

8.4 Issue report: Mathhub and content

9 Sources

Drupal Attack:

²EdNOTE: What's up with Theory and View?

³EdNOTE: I am a Note

- https://nvd.nist.gov/view/vuln/search-results?cpe=cpe%3A%2Fa%3A Drupal%3A Drupal%3A7.38&page_num=0&cid=3
- https://www.theregister.co.uk/2018/03/28/drupal_urgent_security_software_patch/
- <https://www.zdnet.com/article/update-drupal-asap-over-a-million-sites-can-be-e>
- <https://thehackernews.com/2018/04/drupal-cryptocurrency-hacking.html>
- <https://thehackernews.com/2018/04/drupal-site-vulnerability.html>

React: <https://reactjs.org/docs/getting-started.html>