

The background image shows a serene lake scene. On the left, a large, leafy tree stands on a grassy bank. In the middle ground, several small boats are docked along the shore, most of which are covered with blue tarps. One boat has a green tarp. In the foreground, a white swan is swimming in the water. In the background, a large, modern building with a reddish-brown facade and many windows is visible. The sky is overcast with grey clouds.

Challenge Task

Advanced Distributed Systems and Blockchain, HS 2020

Florian Bitterlin, Andreas Baumgartner, Pascal Schürmann

Agenda

- Goal
- Software Design
- Implementation
- Fulfillment of requirements
- Demonstration

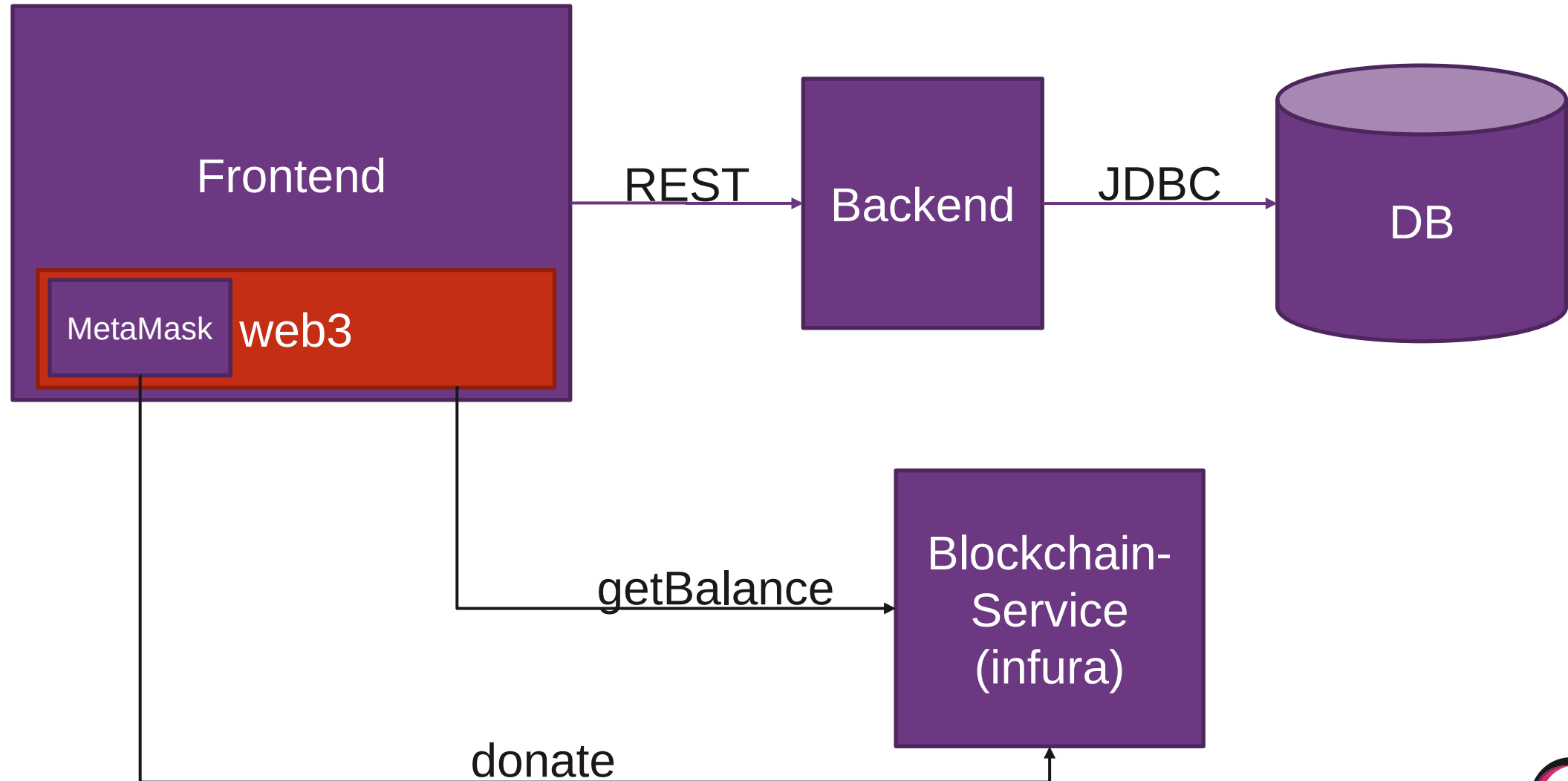
Goal

- „A tool to donate to non-profit projects for the greater good and keep track of donations with the help of a special contract.”

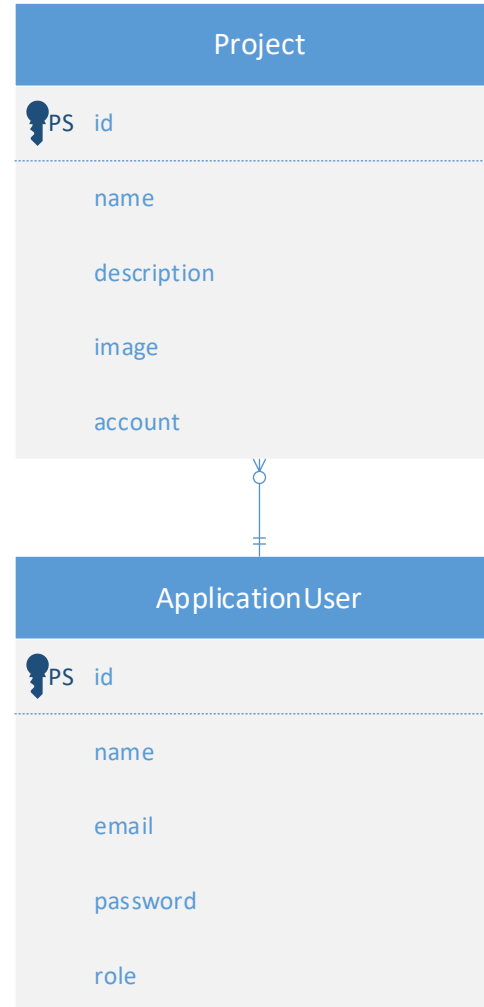
Software Design

- Simple client server design
 - Frontend, Backend, Database
- Scalable through Docker
 - All deliverables are docker images
- Ethereum
 - All financial processes are handled in the browser
 - Usage of a blockchain service
 - Keep track of donation through a proxy contract

System Diagramm



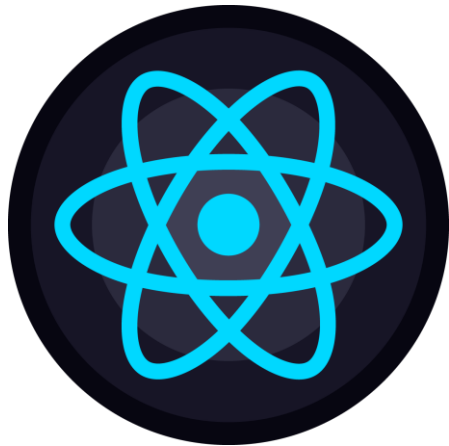
Database



Implementation

React

- React
- Redux
- MetaMask
- Web3

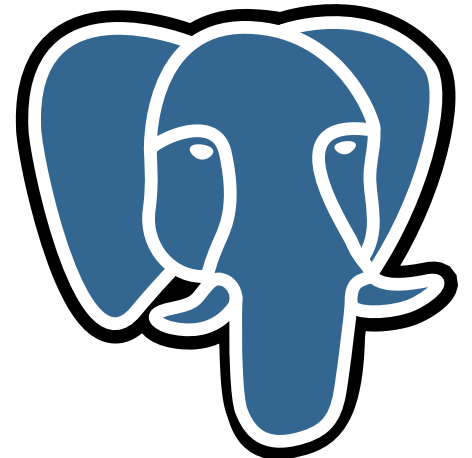


Spring Boot

- Flyway
- Hibernate
- JWT



Postgres



Implementation

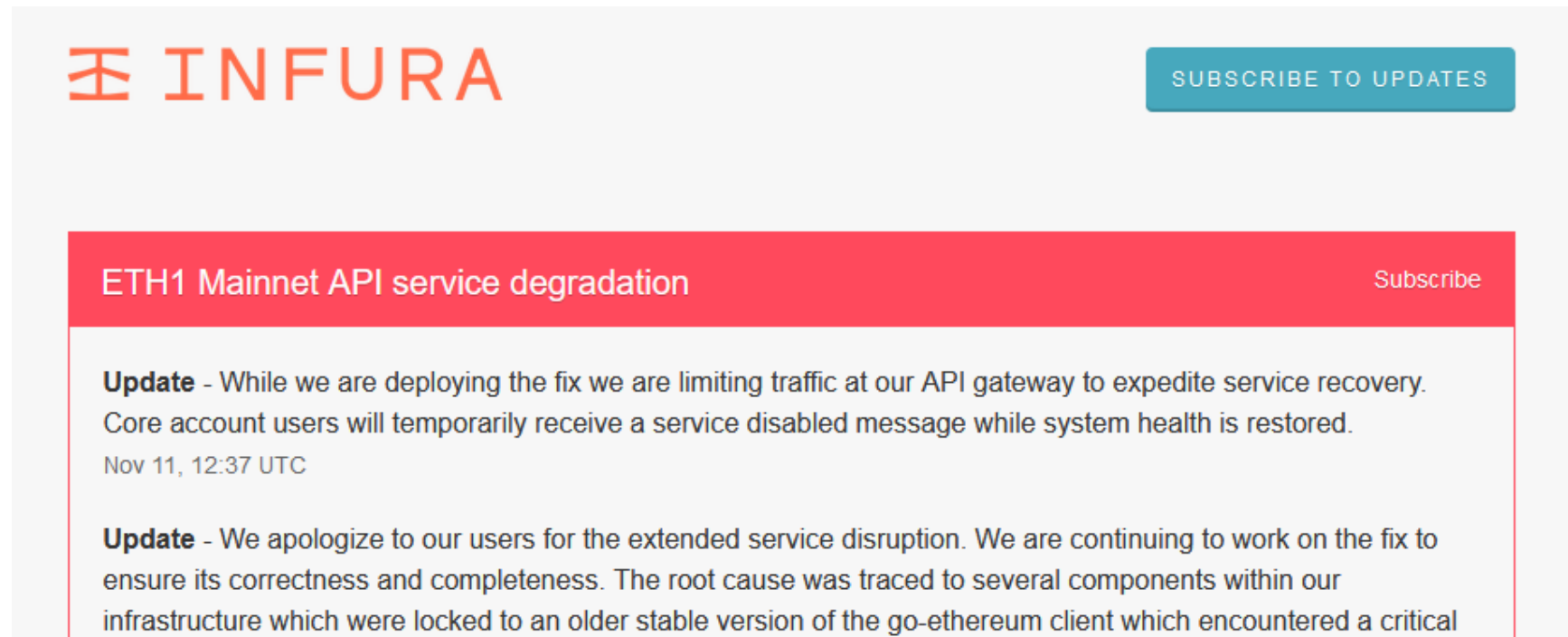
Contract

```
contract donateTo{  
    mapping(address => uint256) recipients;  
  
    function donate(address payable recipientId) external payable {  
        recipientId.transfer(msg.value);  
        recipients[recipientId] += msg.value;  
    }  
  
    function getBalanceOf(address recipientId) external view returns(uint256) {  
        return recipients[recipientId];  
    }  
}
```


Implementation

Trade offs

- Requires MetaMask browser add on to donate
- Not fully distributed
 - We use Infura in the background to interact with the blockchain
 - Infura can fail



The image shows the Infura logo at the top left and a "SUBSCRIBE TO UPDATES" button at the top right. Below this is a red banner with the text "ETH1 Mainnet API service degradation" and a "Subscribe" link. The main content area contains two updates: the first states that traffic is being limited during a fix deployment and that core account users will receive a service disabled message; the second update apologizes for the extended disruption and explains that the root cause was a locked version of the go-ethereum client.

ETH1 Mainnet API service degradation [Subscribe](#)

Update - While we are deploying the fix we are limiting traffic at our API gateway to expedite service recovery. Core account users will temporarily receive a service disabled message while system health is restored.
Nov 11, 12:37 UTC

Update - We apologize to our users for the extended service disruption. We are continuing to work on the fix to ensure its correctness and completeness. The root cause was traced to several components within our infrastructure which were locked to an older stable version of the go-ethereum client which encountered a critical

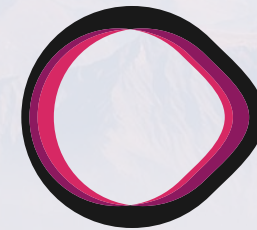
Fulfillment of Requirements

Requirement

- Scalability
- Graphical Frontend
- Public Block Chain

Fulfillment

- ✓ Docker (Load Balancing)
- ✓ We got that
- ✓ Ethereum with own Smart Contract



OST
Ostschweizer
Fachhochschule

Demonstration