HPC platform

C4 model supplement

Jelle Maas (479146)

Koen Janssen (451394)



# Context

Soon the students from Delta will have their own high powered computer, also known as the number cruncher. The goal of the number cruncher system is to give all Delta students the ability to easily send their resource-intensive workload, e.g. AI model training and media rendering, to this powerful computer. This could potentially save lots of time for the Delta students, for the number cruncher can do resource-intensive tasks much more efficiently.

The general idea is thus, to create an easy-to-use system for Delta students so they can submit their workloads to the number cruncher and finally get back their results. This while having a system that automatically manages the workloads on the infrastructure and software side.

This document contains the software architecture for the initial concept of the HPC platform. The design outlines the required containers and components for the HPC platform. Please note that the design is still subject to change.

# 

# Contents

[**Context**](#_jsw6m6kl7bru) **1**

[**Contents**](#_8uau9kelaqt6) **2**

[**Security**](#_f1bhgodx6pjr) **3**

[HTTPS](#_9q8tszy7emet) 3

[OAuth](#_o9ds347xn9d3) 3

[**Model**](#_pyu180ofghjo) **4**

[C1 - Context](#_uww65rsvklak) 4

[C2 - Container HPC Platform](#_ck9xc6qqy5ai) 5

[C3 - Component HPC Controlcenter](#_ii0f77ny7tky) 6

[C3 - Component HPC API](#_p98spepumdmy) 7

[**Conclusion**](#_xb6i4tfbpba) **8**

# 

# Security

The intention is to host both the control center and the API on a publicly accessible domain name. For this reason, setting up the API securely with proper encryption and authentication is important. This is to prevent an unauthorized user from interacting with the physical computer system.

## HTTPS

All HTTP requests are protected by SSL, also known as HTTPS. This ensures that the connection between the client and server is encrypted. This is to avoid manipulation or interception when data is sent over a network.

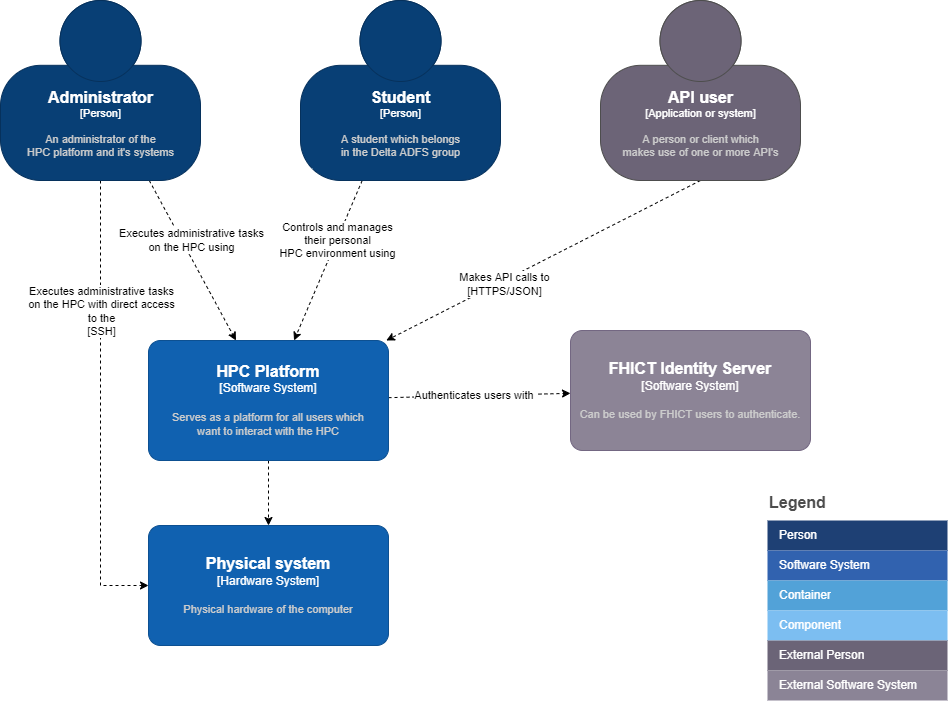
## OAuth

The endpoints are protected with a JSON Web Token (JWT). This token is granted to the user when they log in with the FHICT identity server. The JWT contains specific user data, known as claims, which can be used to, among other things, filter through specific groups within the [Active Directory Federation Services](https://docs.microsoft.com/en-us/windows-server/identity/active-directory-federation-services) (ADFS). This allows the system to, for example, only give access to (specific) Delta students.

# Model

The C4 model is an "abstraction-first" approach to diagramming software architecture, based upon abstractions that reflect how software architects and developers think about and build software. The small set of abstractions and diagram types makes the C4 model easy to learn and use. For more information, please refer to <https://c4model.com/>.

## C1 - Context



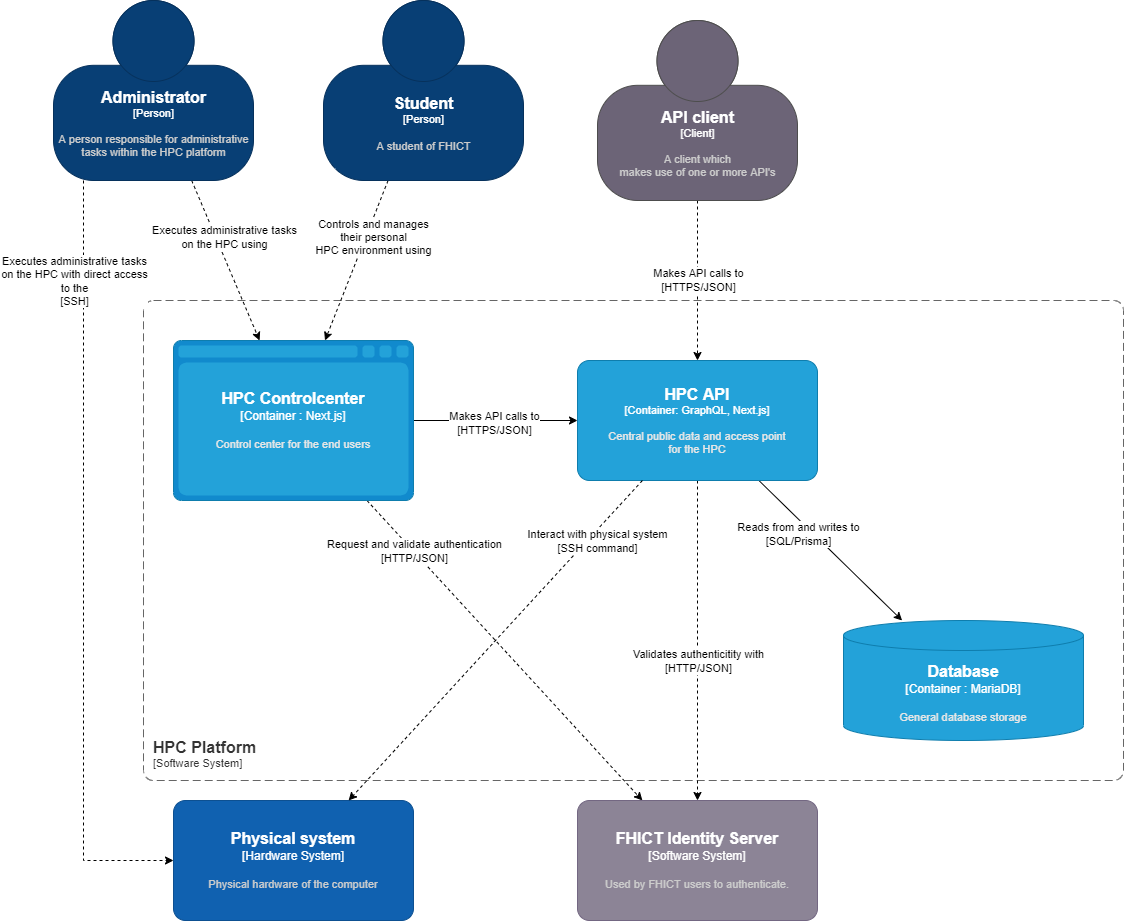
In the context diagram above three users are shown, these are the administrator, student and API user. In this case, the API user can be an external user, e.g., a CI/CD pipeline. These three users interact with the HPC platform. The HPC platform authenticates these users using the FHICT identity server.

When the users are authenticated, they are allowed access to the HPC platform. After which they can interact with the physical system by using the HPC platform as a man in the middle.

Contributors:

* J. Maas
* K. Janssen

## C2 - Container HPC Platform

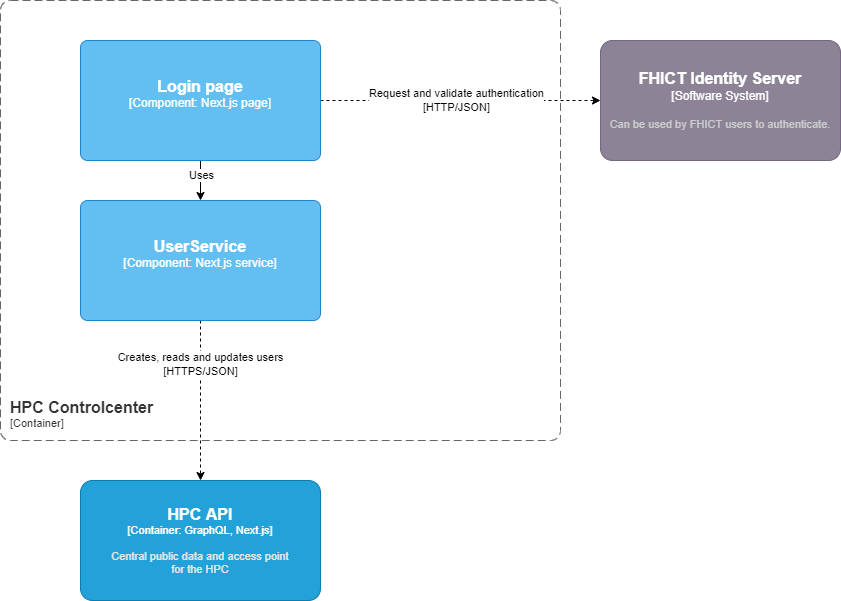


The administrator and student interact with the HPC Control center for controlling their own virtual work space on the HPC after authenticating with the FHICT Identity Server. The HPC Control center makes use of the HPC API as the Physical HPC system is controlled solely by the API.

Contributors:

* J. Maas
* K. Janssen

## C3 - Component HPC Controlcenter

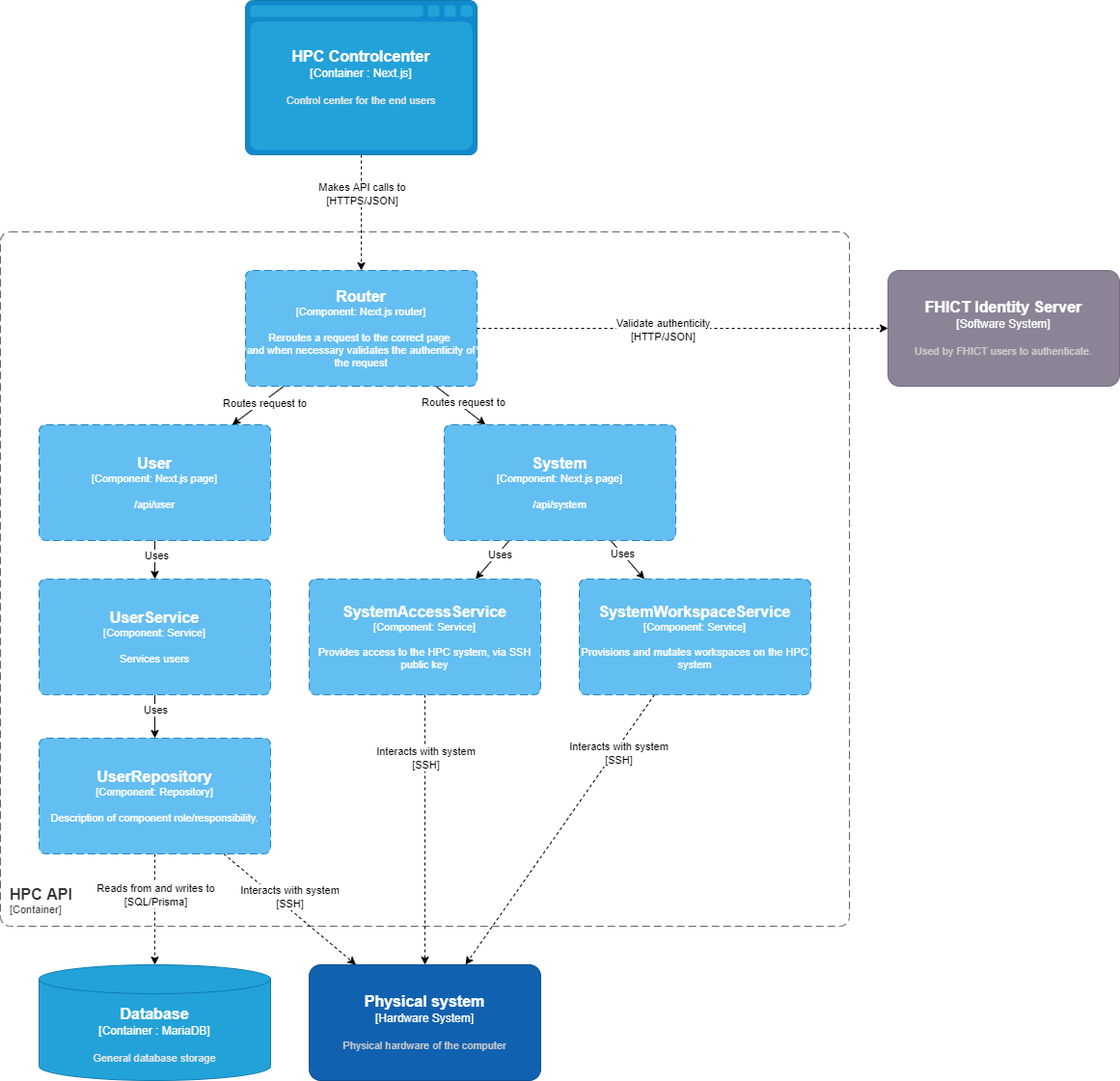


This component layer goes more in-depth on the HPC Control center container. On the next.js login page, the administrator or student can press login with FHICT to authenticate with the FHICT Identity Server. Here they can log in with their Fontys FHICT account. If the logged-in user is in the delta group they are logged into the system and forwarded to the control panel. Here they can interact with their virtual workspace on the HPC server.

Contributors:

* K. Janssen

## C3 - Component HPC API



The HPC control center interacts with the HPC API using API requests (HTTPS/JSON). The requests are received by the internal router which, when necessary, validates the request by using the FHICT identity server. The router then proceeds to reroute the request to the appropriate Next.js page. These pages each correspond to their endpoints, of which currently there are two: user and system.

Each endpoint has its own dependencies, which usually come in the form of services. These services can interact with both the database and the physical system according to their specifications.

Contributors:

* J. Maas

# 

# Conclusion

Now it is sure that most components of the software architecture are well-designed and thought out. The C4 model gives a clearer picture of the software architecture and allows us to plan accordingly. The next step is to start with the project development and finally produce a proof of concept for the stakeholders.