

Quantified Student

C4 Models

Walter Sajtos (438989)



Table of Contents

Table of Contents	2
Version History	3
1. Overview	4
1.1 Preface	4
2. What has been done?	5
2.1 Context Diagram	5
2.3 Components Diagram Gateway	7
3. Result	8
3.1 KPI's	8

Version History

Version	Date	Author	Comment
0.1	April 26, 2022	J. Maas	Initial document setup and write conclusion
0.2	May 10, 2022	J. Maas	Write introduction and elaboration for the C4 model
0.3	May 23, 2022	J. Maas	Update context and container diagrams
0.4	May 30, 2022	J. Maas	Update context diagram to show extendibility of data sources
0.5	June 9, 2022	J. Maas	Update diagram
1	02-01-2022	W. Sajtos	Update C4 models to include the gateway
1.1	18-01-2022	W. Sajtos	Include Data Sharing middleware in the C4 model

1. Overview

1.1 Preface

During the standups and meetings of the team, there needed to be more clarity about the responsibilities of each part of the system. The goal of this document is to create a clear and concise overview of the Quantified Student system.

Assignment

Create C4 models for the QS system.

What's it for?

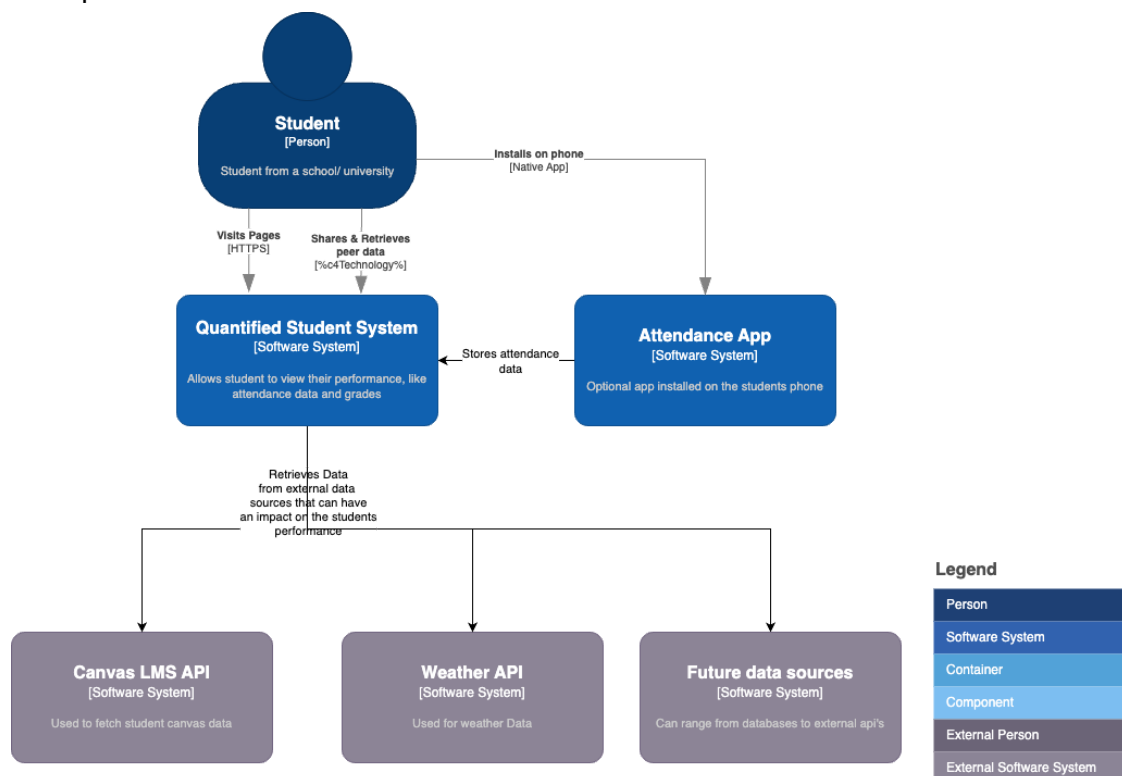
Create more clarity of the responsibilities of each part of the system and prevent confusion in the future. It also allows outsiders to understand how the parts of the system interact together.

2. What has been done?

2.1 Context Diagram

The goal of the Context Diagram is to provide a starting point, showing how the system interacts with the 'outside world'

The students interact with the system. They access the QS system through the browser. The QS system mostly retrieves data from external APIs. However, not all data can be fetched from external APIs. Some data needs to be aggregated. Aggregating data allows for a 'dumb' front end and faster response times as we don't need to fetch all the data, just the valuable parts.

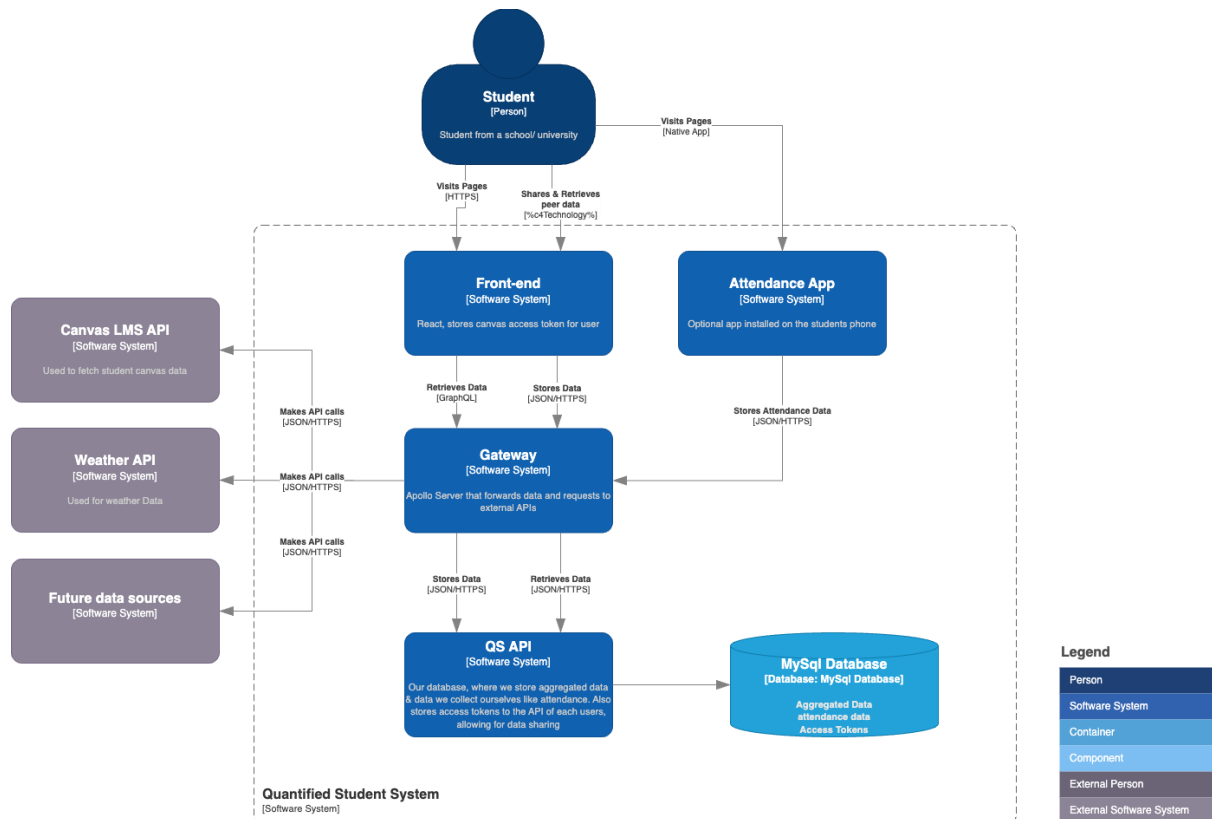


2.2 Container Diagram

The goal of the Container Diagram is to show the 'Building blocks' of the system and show how they interact with each other.

The gateway is the center point of the system. All data flows through the gateway, which is then altered (if needed) and forwarded to the required destination. The same goes for retrieving data.

The QS API is the place where all aggregated data is stored. It also stores data that is not available in the external APIs, like attendance data.



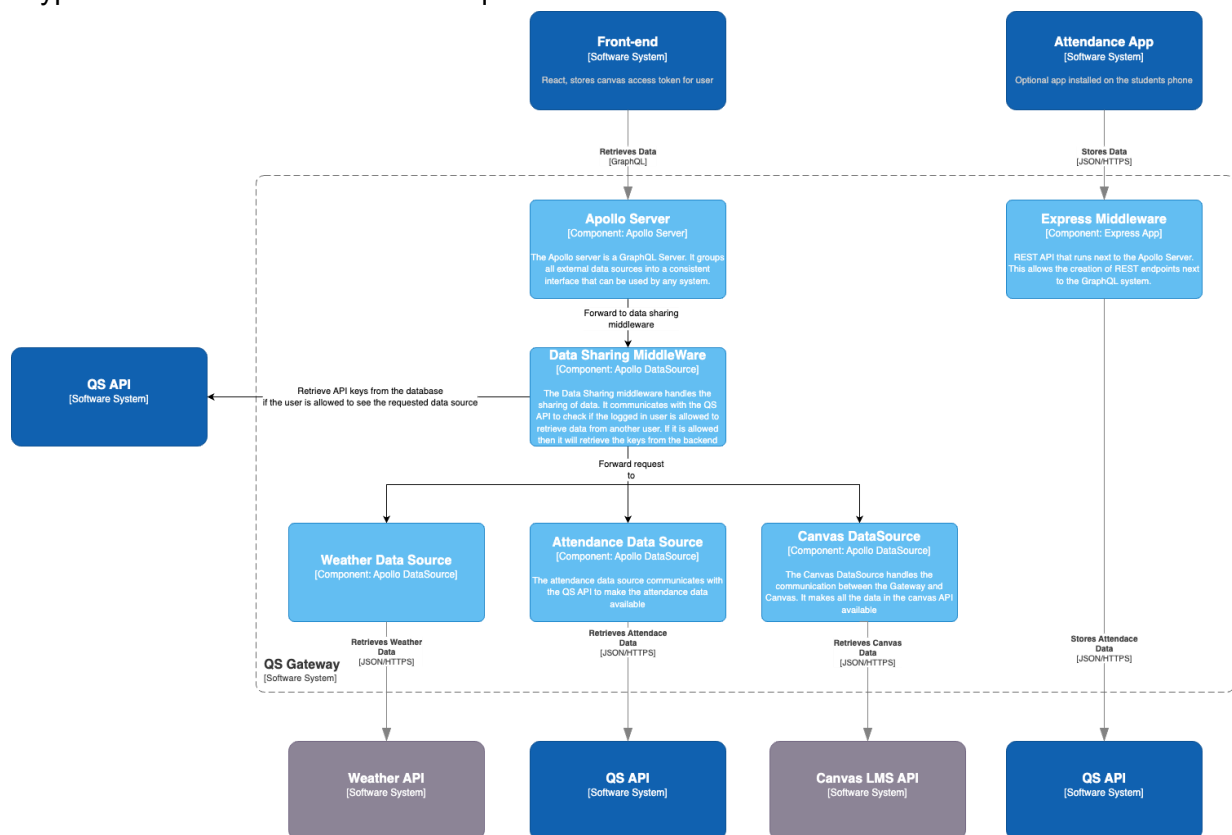
2.3 Components Diagram Gateway

The goal of the components Diagram is to zoom in on the gateway and show how it is built up internally. The gateway consists of 2 types of components; data sources and resolvers.

The Apollo server is a GraphQL Server. It groups all external data sources into a consistent interface that any system can use.

Data sources consist of one or more resolvers. The resolver describes to the GraphQL Client what types of data it can query.

A resolver is responsible for fetching the data from the external data source. This can be a REST API, GraphQL API, or a database. Resolvers can also be nested into other resolvers if a type of data needs more than 1 request.



3. Conclusion

After designing the c4 models, the team can come back to the models to see if what they are designing/ realizing align with the rest of the application. As a first next step the advice is to create a first prototype with the structure that is laid out in this document. When the first prototype is ready, these designs should be revised to make sure everything works as expected.