Architectural Design Document

Describing the architectural, C4 & UML designs of the Care platform

Name: Victoria C. A. Fong

Student number: 4576993

Class: S-A-RB-CMK 4

Introduction

This document describes the architectural design of the Care Web-application in multiple diagrams to get an understanding of its structure. Such diagrams are the architectural diagram, c4 diagram & UML diagram.

The document describes each diagram of the relations and tools/technologies used to get an overall understanding of how they communicate and work with one another.

Table of Contents

Architectural Diagram
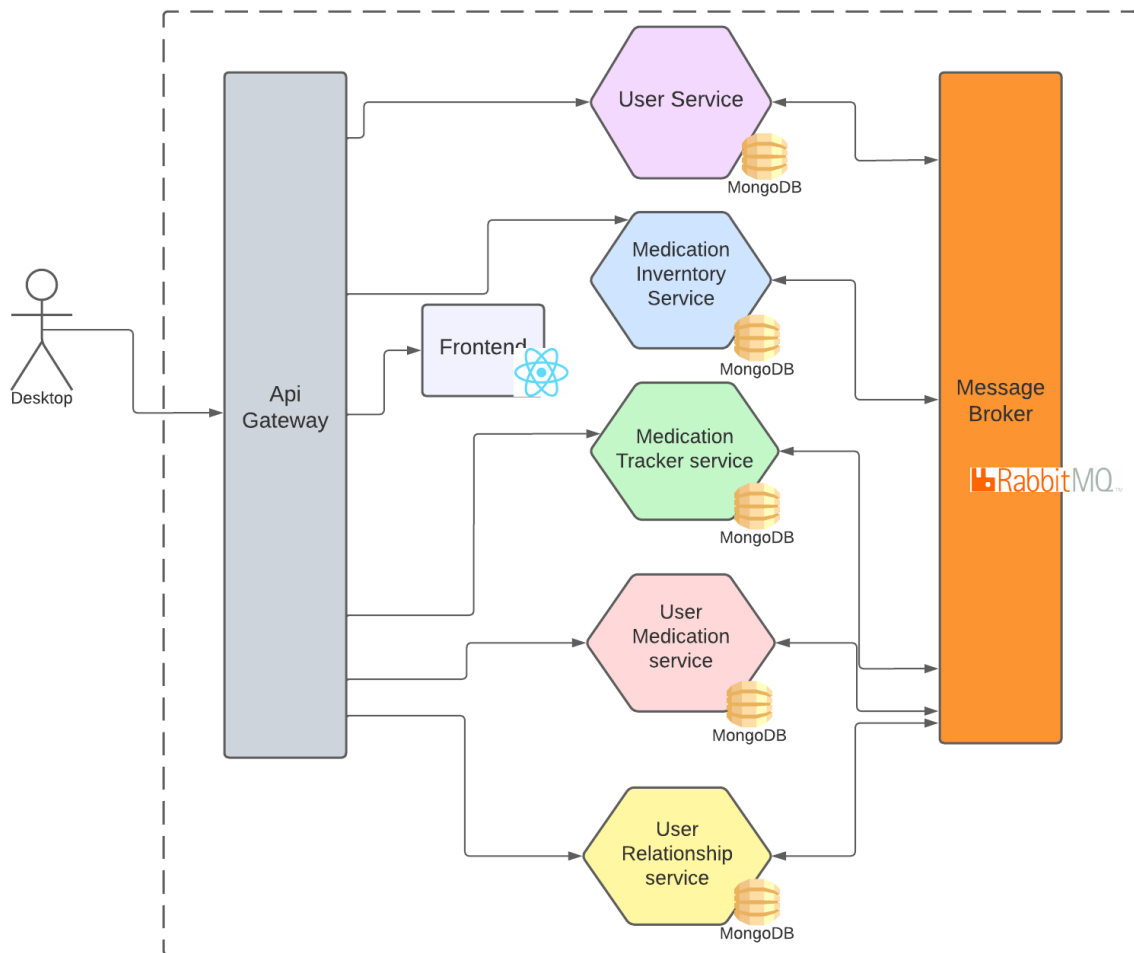
Diagram 1: "Architectural Diagram of the Care Web-application"

Link to GitHub
Architectural Diagram on GitHub

Description

This diagram shows the microservices and communication tools used for the Care Web-application.
I will describe a bit more of each component that is shown in image 1 of the diagram.

Component	Description	Technology used
API Gateway	the API gateway takes calls from the client and routes it to the correct service.	
Frontend	the frontend serves as the GUI for the Web-application for the client to communicate with.	React
User-Service	the user service is based for holding the most personal data of users. This service holds for example the name, password, email, etc.	.Net & C#
Medicine-Inventory-Service	The service holds all medicine registered to the web-application. Currently this is done by hand but the goal is to connect it to the government open source registered medicine database.	.Net & C#
Medicine-Tracker-Service	This service keeps alerts the user when to take their medicine with keeping track of what time and intake of the medicine.	.Net & C#
User-Medicine-Service	The service is based on the medicine a user is taken with extra information of the medicine such as the amount of medicine, start and end date of intake and more.	.Net & C#
User-Relationship-Service	The platform uses this service to find the relation between users to who is and is not allowed to access/see the user's information without their consent.	.Net & C#
Message Broker	The message broker helps to send and receive messages between services as well as placing messages that have not yet been received in a safe place.	RabbitMQ
Enterprise Service Bus	Enterprise Service Bus is used for the service to use subscribe and publish methods of data. The platform uses Mass Transit for its easy to use and implementation in .Net for which the platform is built upon.	Mass Transit

Choices made

The entire web-application architectural design started with three services which were the user service, medicine service and user-medicine service. After feedback from teachers and critically thinking of the purposes of the platform and what will be needed in the long run. This end design has come to place where it is more flexible for change and is not restrained on other services.
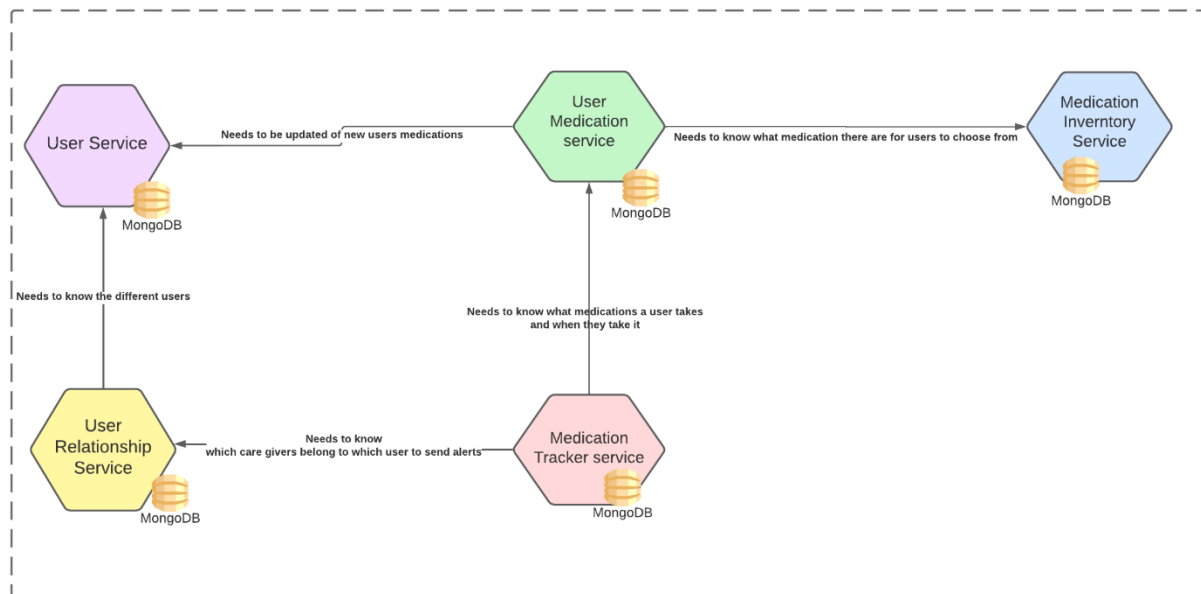
Relation Diagram between services

Diagram 2: "Relationship between microservices of the Care Web-application"

Link to GitHub
Relation Architectural Diagram on GitHub

Description
Diagram 2 shows the relationship between services and what data they store from one another. To make is clearer, a detail description is written below. The services that uses data from other services will be described of its purpose:

User-Medicine-Service

Services the User-Medicine-Service communicates with	Description of Data shared
User -Service	The user medicine service needs to know the user ids that is registered to the platform so that it can store this in their database. This way of storing a part of the user service data makes it faster and not tightly coupled to the service in the sense that when the user service is down, the user medicine service is still able to function.
Medicine-Inventory -Service	The user medicine service needs to be able to know which medicines the platform provides for the user to choose out of. The medicines data is all stored in the medicine inventory service. To be able to not have duplicated data of this service in the user medicine service and to be able to function without the medicine service. A small portion of the data such as the name and id are stored in the user medicine service.

Medicine-Tracker-Service

Services the Medicine-Tracker-Service communicates with	Description of Data shared
User-Medicine-Service	The medicine tracker service uses stored parts of the user medicine service data to keep track and alert the user to when to take their meds. This data is the intake time, user id and medicine.
User-Relationship-Service	The medicine tracker is to alert not only the user (patient to whom is taken the medicine) but also close members they have allowed to receive notifications to remind the user.

User-Relationship-Service

| Services the User-Relationship-Service communicates with	Description of Data shared
User -Service	The relationship service determines the relationship between users. This can be friends, family, specialist and more. When a new user is registered the user relationship service stores the user id. This way the user relationship service is not reliant on the user service.

C4 Diagram

C1 Diagram
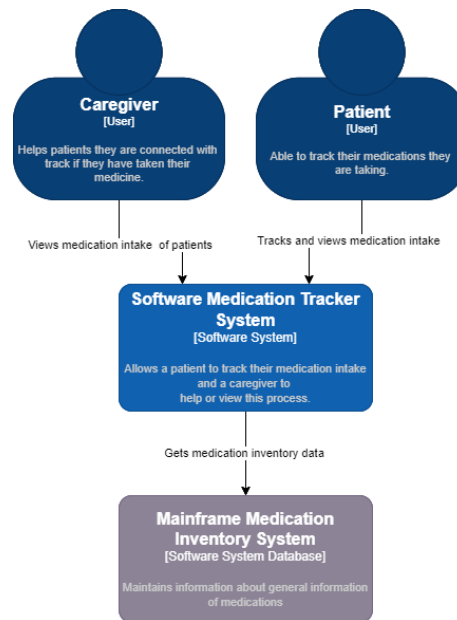
Diagram 3: "C1 Diagram of the Care Web-application"

Link to GitHub
[C1 Diagram on GitHub](#)

Description
The C1 diagram also known as the "System Context Diagram" is the starting point to showcase how the software system fits in the scope. This gives a very high-level perspective that can be an easy way before diving deeper into more technical aspects.

Choices made
There are primarily 2 users that the application is based for which is a patient and a caregiver. Further, the platform will be connected to a mainframe database of the local government to keep track of medicines registered for users to choose from.
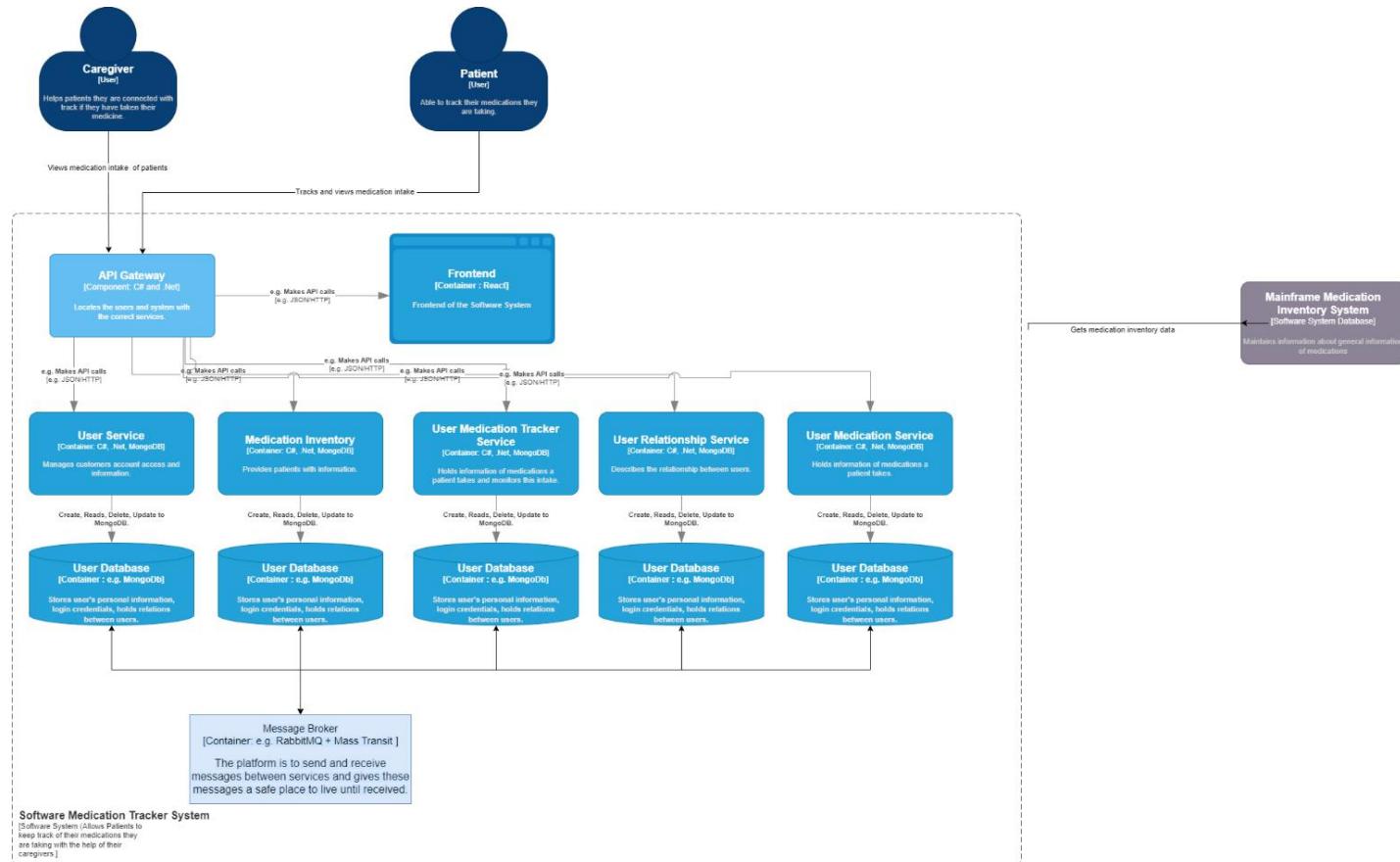
C2 Diagram

Diagram 4: "C2 Diagram of the Care Web-application"

Link to GitHub
C2 Diagram on GitHub

Description
The C2 diagram also known as the "Container Diagram" zooms into the software scope of the project. It is used to give high-level perspective of the technical building blocks.

Choices made
The C2 diagram shows the relationship of the entire service with more detail description compared to the architectural diagram.

UML – Class Diagram

Context

UML Diagrams help visualize the different types of objects within a system and the relationship among other classes. Further, the diagrams help to illustrate the operations and attributes of the classes.
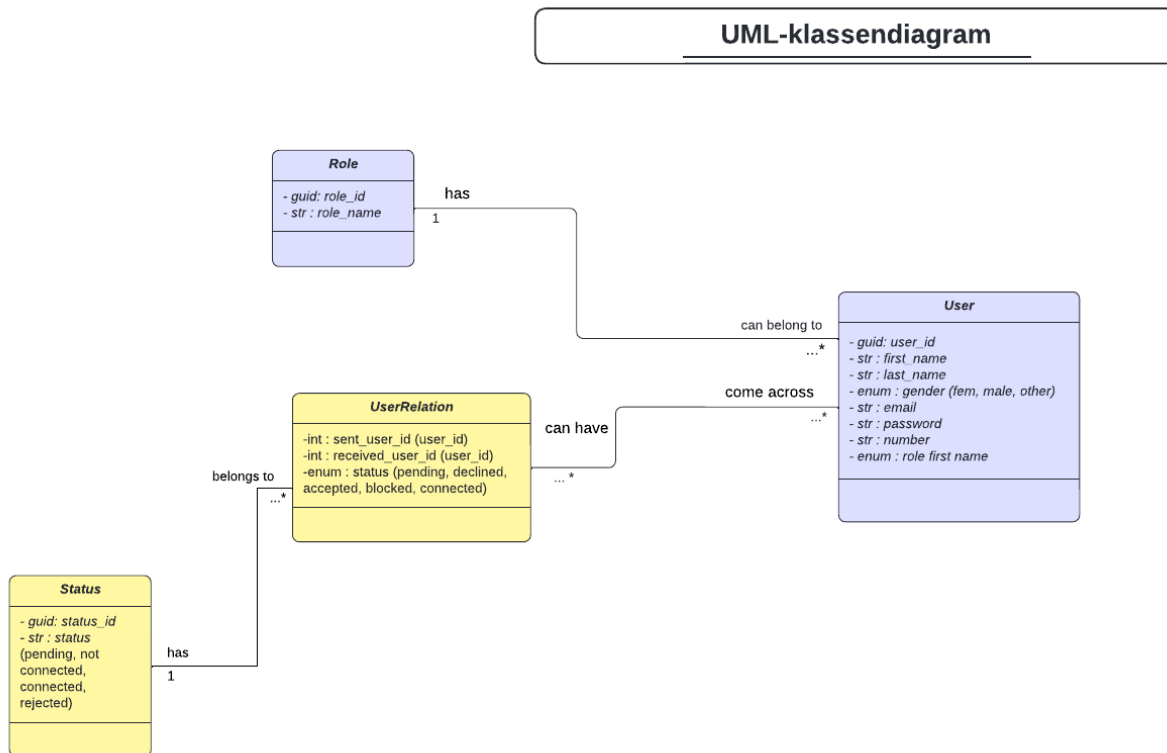
User v5 diagram

Diagram 5: "UML Class Diagram based on User of the Care Web-application"

Link to GitHub

[UML-Class-Diagram regarding users on GitHub](#)

Description

The UML Diagram based solely on any user is shown in diagram 5. Here we see what data is stored in the user-service (purple boxes) and the user relationship service (yellow boxes). We also see what attributes they share with one another which has been touched upon for describing the architectural diagram.
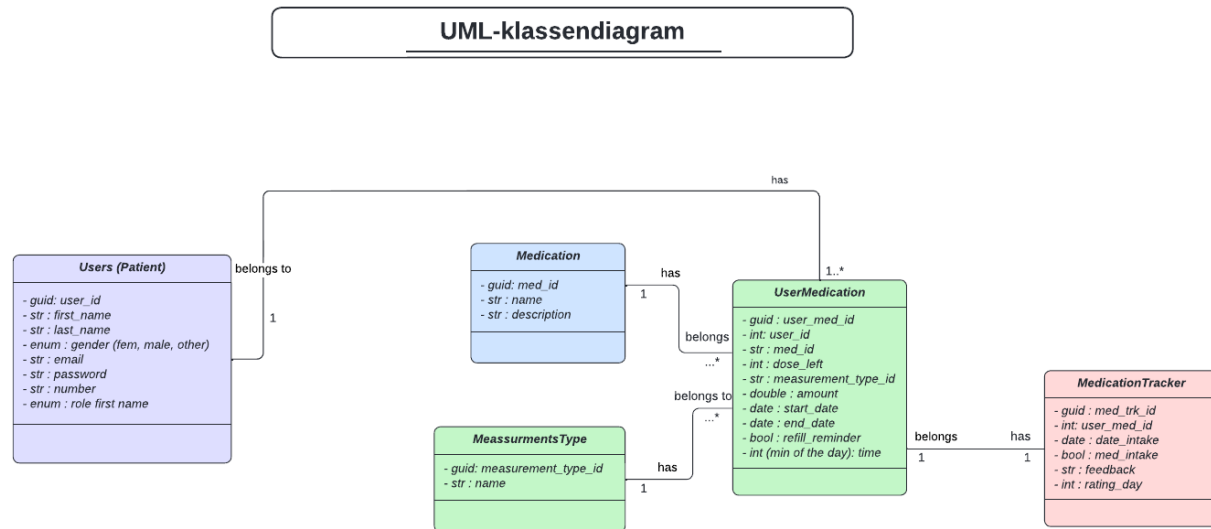
Patient v5 Diagram

Diagram 6: "UML Class Diagram based on Patient of the Care Web-application"

Link to GitHub
UML-Class-Diagram regarding patients on GitHub

Description
The UML Diagram above is based on the services that is used for user's as patients (the users who intake medicine). Here we see what data is stored in the user-service (purple boxes), the medicine service (blue box), the user medicine service (green boxes) and the medicine tracker service (pink box). We also see what attributes they share with one another which has been touched upon for describing the architectural diagram.