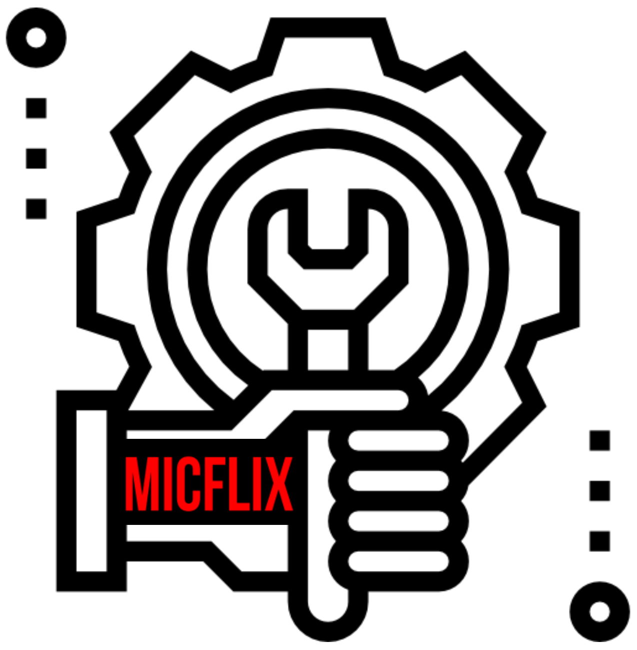
TECHNICAL DESIGN – MICFLIX

FONTYS SEMESTER 6 SOFTWARE PROJECT



Maintenance free icon: Flaticon.com

Date	**07-03-2022**
Version	**0.2**
State	**In progress**
Author	**Mickey Krekels**
Class	**RB04**

VERSION HISTORY

Version	Date	Author(s)	Changes	State
0.1	07-03-2022	Mickey Krekels	Added the main structure of the document.	In progress
0.2	24-03-2022	Mickey Krekels	Updated C2 model and added chapter 2	In progress

PREFACE

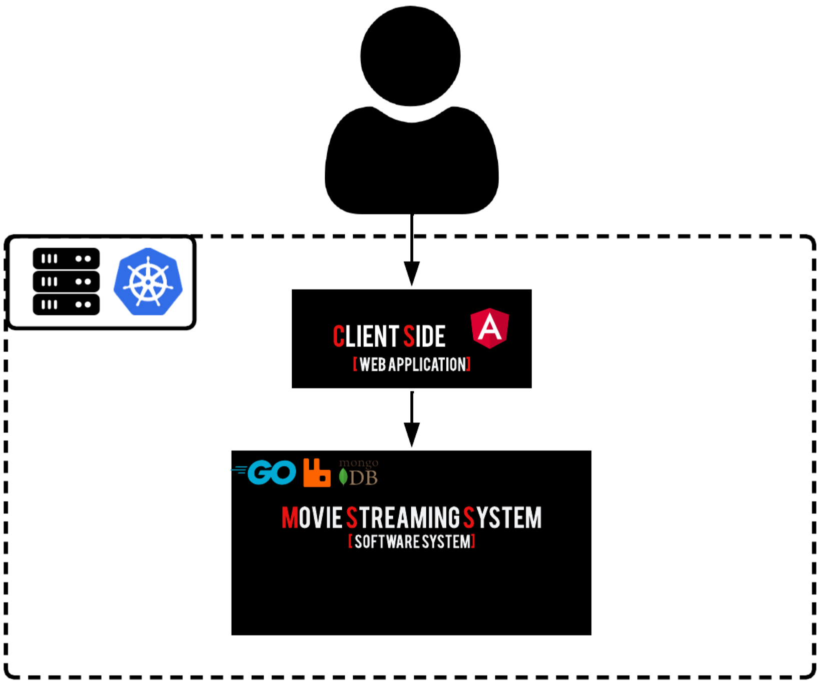
This project is made for the Fontys semester-6 personal assignment. The goal is to make a streaming service clone with a strong inspiration from the platform Netflix. This clone will be called Micflix and the final version of the project will be of enterprise quality.

This document contains all the technical designs of this assignment. The design will be split up into 4 different C model levels (1). Where each of the levels represents a different layer in the architecture, specifically context, container, component and code layer. In chapter, 2 the reason will be explained for each of the architectural choices. At the end of the document, there will be a short conclusion.

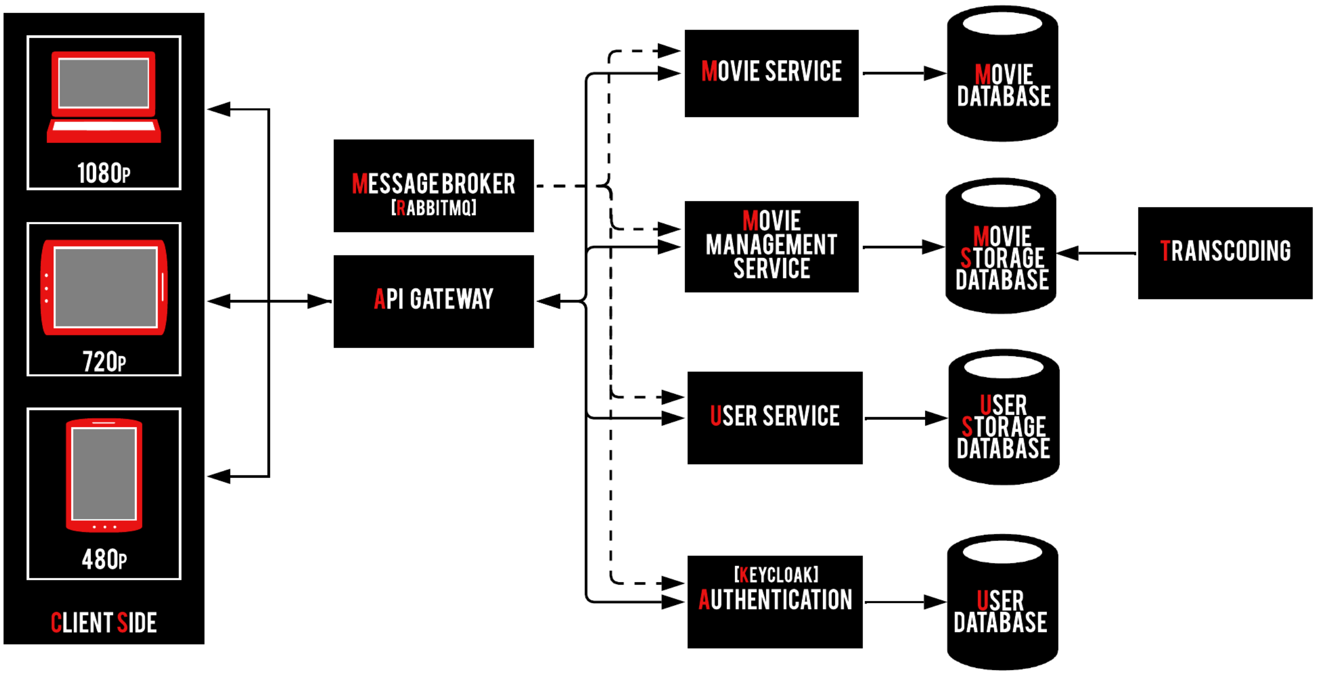
TABLE OF CONTENT

1 PROJECT ARCHITECTURE

1.1 C1 MODEL



1.2 C2 MODEL



1.3 C3 MODEL

1.4 C4 MODEL

2 DECISION JUSTIFICATION

In this part of the document, I will describe the overall decision justification of my Technology choices. This information will be described in a table with the pros and cons of each of the research choices.

2.1 WHY USE THE MONGODB DATABASE?

Types	MongoDB	Microsoft SQL Server	PostgreSQL
Description	NoSQL type database	SQL type database	SQL type database
Pros and Cons	+ Supports JSON + Data structure can be stored easily + Fast operations and queries + Free Cloud-hosted cluster + Atlas Operator for Kubernetes + Lots of documentation - Query language is not SQL - Difficult to setup	+ Fast and stable + Ability to adjust and track performance levels - Heavy on resources - Enterprise pricing is high	+ Supports JSON + Scalable - low on documentation - Slow on large bulk operations or read queries
Estimated cost	[**HIGH**] Short term [**LOW**] Long term	[**Medium**] Short term [**HIGH**] Long term	[**LOW**] Short term [**Medium**] Long term

The Biggest pro of MongoDB is the ability to host a free cluster of **512MB** on the cloud. This allows for good testing in the prototype phase of the project, and if scaling up is needed it can be easily done so. There is also a lot of documentation on MongoDB and how to use it in a Kubernetes structure effectively.

For these reasons, I think that the **MongoDB** database type is perfect for this project.

(All the information on the pros and Cons were found on MongoDB (2) website and Keycdn (3))

2.2 WHY USE THE RABBITMQ MESSAGE BROKER?

Types	RABBITMQ	KAFKA	Redis
Description	NoSQL type database	SQL type database	SQL type database
Pros and Cons	+ Ease of Integration + Fault Tolerance + Lots of documentation + Library support for Golang - Can be Hard to maintain	+ Scalability + Performance + Quick data transfer - Logging can be confusing	+ Cache speed + Lacks UI + Clustering and sharding - Single-threaded - Difficult to use - Heavy on resources
Estimated cost	[**LOW**] Short term [**Medium**] Long term	[**Medium**] Short term [**Medium**] Long term	[**HIGH**] Short term [**HIGH**] Long term

For the current project, I use Golang as a Back-end language for creating the services. The API template I use has already a working library intergraded into the service. Therefore the initial connection is easy to set up. The user interface also provides a good way to debug problems during development.

For these reasons, I think that the **RABBITMQ** message broker is perfect for this project.

(All the information on the pros and Cons were found on Trustradius (4))

3 CONCLUSION

4 BIBLIOGRAPHY

1. **Brown, Simon.** The C4 model for visualising software architecture. *c4model.* [Online] https://c4model.com/.

2. **mongodb.** ***The MongoDB Kubernetes Operators.*** **[Online] https://www.mongodb.com/compatibility/kubernetes.**

3. The Pros and Cons of 8 Popular Databases. ***keycdn.*** **[Online] 20 4 2017. https://www.keycdn.com/blog/popular-databases.**

4. Software Reviews You Can Trust. ***trustradius.*** **[Online] https://www.trustradius.com/.**