Healthcare mobile application using MQTT

Course: Internet of things

Professors:

Sami Melki – Habib Smei

Manel Kacem

University Year 2021-2022

Plan

1. Introduction
2. Presentation and goals
3. Conceptual analysis
4. Implementation
5. Conclusion

Introduction

Healthcare is an industry that has been greatly impacted by the rise of the internet of things (IoT). With the use of connected devices and sensors, medical professionals are now able to monitor and manage patients' health in real-time. This has led to the development of mobile health care applications, which provide users with user-friendly interfaces and advanced features.

Presentation & goals

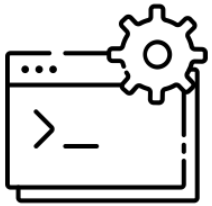
As part of an IoT course, we created a healthcare mobile application that helps users track and monitor their most important health indicators. The app allows users to follow their number of steps, heart rate, blood pressure, blood oxygen levels, and body temperature. With this information at their fingertips, users can better understand their overall health and make any necessary changes to improve their wellness.

Conceptual analysis

When choosing a backend for a healthcare mobile application that uses the MQTT protocol, scalability is an important factor to consider. The backend should be able to handle an increase in the number of users and devices without experiencing performance issues. Two databases were compared for this project, and the most suitable technology was selected based on key differences.

Key Difference Between MongoDB and MySQL

	MySQL	MongoDB
Use case	Applications that use a moderate number of join operations.	Real time analytics, content management, internet of things, mobile apps.
Data structure	Structured data with clear schema.	No schema definition defined
Risk	Risk of SQL injection attack.	Less risk of attack due to design.
Analysis	A great choice for structured data and need a traditional relational database.	A great choice if you have unstructured and/or structured data with the potential for rapid growth.



Choice of database

In conclusion, we have chosen to use MongoDB for this project because we anticipate a large amount of data coming from the MQTT broker.

MongoDB is a powerful document-oriented database that is well-suited for storing and managing large amounts of data. Overall, we believe that MongoDB is the best choice for our project due to its versatility, performance, and scalability.

MQTT Protocol

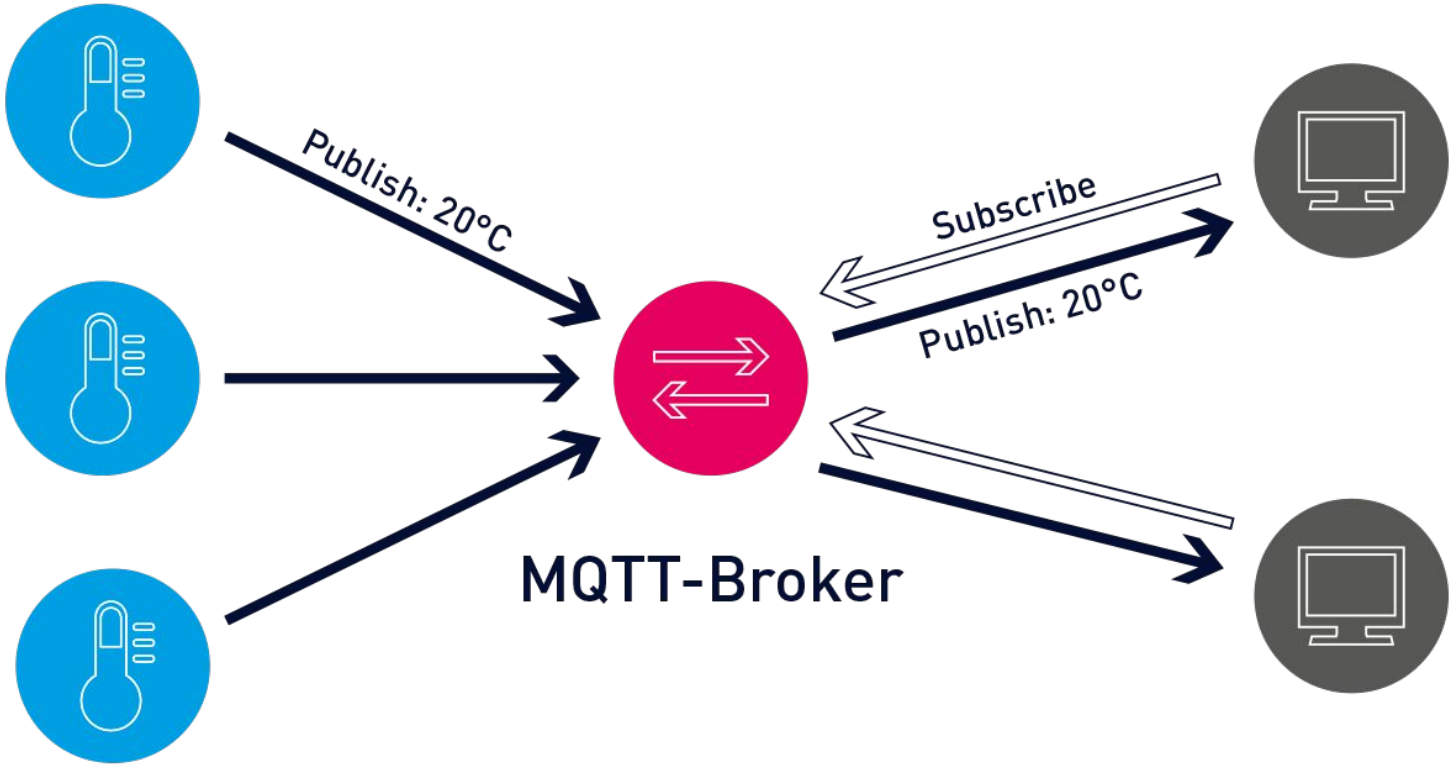
MQTT (Message Queuing Telemetry Transport) is a lightweight messaging protocol designed for low-bandwidth, high-latency, or unreliable networks. It is commonly used in IoT applications to transmit data between devices and platforms in a reliable and efficient manner. The MQTT protocol uses a publish-subscribe model, where clients can publish messages to a broker, and other clients can subscribe to the broker to receive those messages.

MQTT Protocol Architecture

MQTT runs on top of TCP/IP using a PUSH/SUBSCRIBE topology. In MQTT architecture, there are 2 types of systems: clients and brokers.

1. The **broker** is responsible for routing the messages to the appropriate clients based on their subscriptions.
2. **Clients** connect to the broker using a TCP connection, and communicate using a simple, binary-based protocol.

MQTT Protocol Architecture



Node JS

Node.js is a JavaScript runtime environment that allows developers to build server-side applications with JavaScript. It uses an event-driven, making it lightweight and efficient for building real-time, scalable, and high-performance applications.

Hive MQ

HiveMQ is a popular MQTT broker, which is a central server that receives messages from clients and routes them to other clients based on their subscriptions.

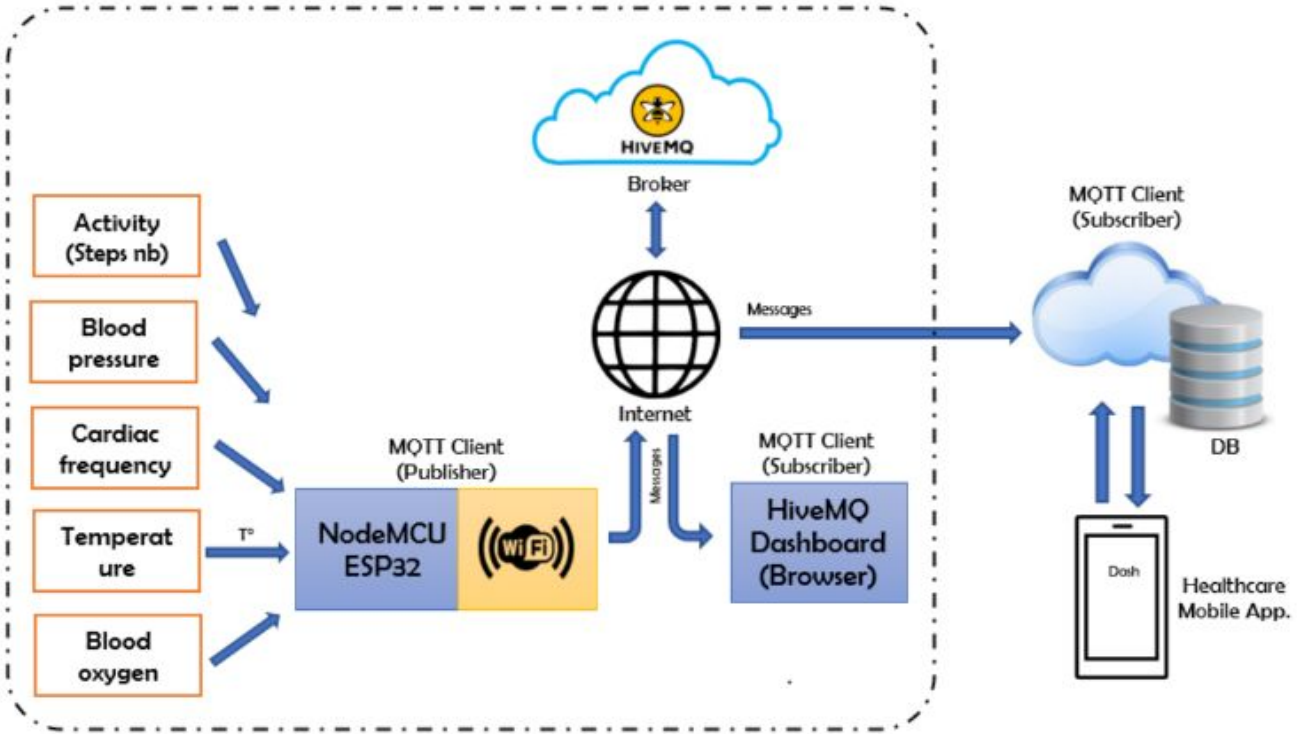
Frontend of choice

In this section, we will discuss our choice for the frontend of our mobile application. There are many options available for developing the user interface of a mobile application, including native development with Android and iOS, cross-platform frameworks such as React Native and Flutter. In this project we opted for Flutter technology.

Flutter

Flutter is a cross-platform UI toolkit that is designed to allow code reuse across operating systems such as iOS and Android. The goal is to enable developers to deliver high-performance apps that feel natural on different platforms, embracing differences where they exist while sharing as much code as possible.

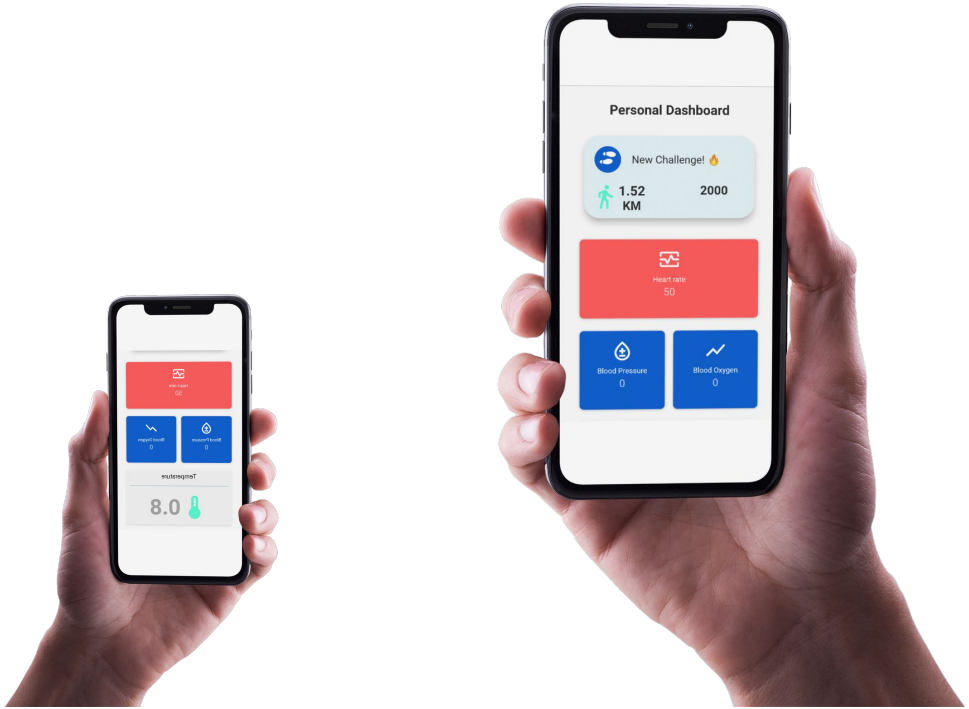
Overall architecture of healthcare architecture



Implementation

Wrap up of technologies used:

- Flutter
- HiveMQ (Broker)
- Node.js
- MQTT Protocol
- MongoDB



Conclusion

The Internet of Things (IoT) is changing the way we approach healthcare, allowing medical professionals to monitor and manage patients' health in real-time. Together, IoT, MQTT, and Flutter enable the development of powerful, scalable, and reliable applications that can collect, transmit, and analyze data from connected devices. These technologies have the potential to revolutionize industries and improve the quality of our lives.

Thank you for

Your attention!