**Flexso Intelligent Water Consumption**

SAP Challenge – Hack the Future 2022

Technical Documentation

# Introduction

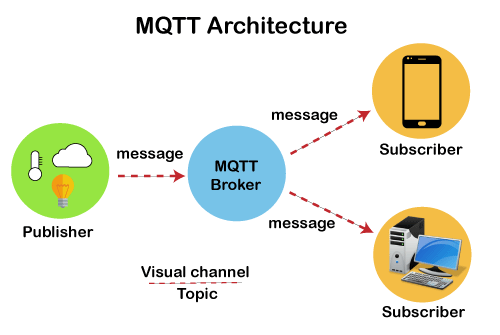
Flexso is a highly innovative company that makes use of the newest and finest hard- and software. They use sensors to monitor different processes to analyse and improve them. Almost everything is digitalized, and many processes are automized, such as starting the dishwasher remotely and limiting the amount of water it uses.

Since Flexso really wants to focus on being a sustainable, eco-friendly company, they want us to develop an application to monitor their water consumption using the magic of IoT. It needs to include both real-time and historical data, it needs different kinds of overviews to monitor the water consumption over the past months or weeks and most importantly, and it needs to be user friendly so they can make data-comparisons and conclusions very easily.

# Technical Setup

## IOT Service

We will deliver a fully working IoT Service that we integrate with the CAP middleware. To combine and deliver mocked historical data and live-data from the IoT device. The IoT service uses the MQTT network protocol, a lightweight and popular publish-subscribe protocol that is commonly used for IoT applications.



How does it work? In an MQTT setup you have 2 main components, a broker and clients. The broker is the central hub were all the messages pass through from clients and gets send out to clients. We chose the HiveMQ cloud broker as its easy to setup without any cost or need to host the broker by yourself. ( <https://www.hivemq.com/mqtt-cloud-broker>). We have 2 clients in our setup with different roles, a publisher and a subscriber, the IoT device that measures our water consumption and the UI5 app that represents that consumption. These two have to connect to the broker first before they could send or subscribe messages. To send data you need to publish that data under a specific subject and to receive that data you need to subscribe to that subject.

You will need to complete some code regarding the MQTT client of the UI5 app so you can find a basic code example in the README file under ‘MQTT JS client example’.

## The HTF-2022 application

The HTF-2022 application consists of a middleware application built in Cloud Application Programming (CAP) model, and a Fiori frontend application. Here’s how it’s structured:

* App
  + Webapp
    - Controllers -> JavaScript code & eventhandlers behind the views
    - Views -> XML views & frontend structure
    - Img -> all local images used in the application
    - Services -> API-handling
    - State -> Business logic
    - Model -> Object properties & Object related functions
* Db
  + Datamodel.cds -> Database structure
  + Data folder -> test data
* Srv
  + Service.js -> API-handling

You can find this structure in the README, aswell as how to run the application.

In short:

* Open terminal and run following commands:
  + Run npm install
    - This will add all necessary dependencies
  + Run local:db
    - This will start a local SQLite database and register test-data
  + Start the app via npm run start
    - Or run the middleware in debug mode via F5
    - To debug the frontend, use the developer tools via F12

## SAP AppGyver

If you want to create a Low-Code / No-Code project, we’ll recommend SAP AppGyver!

In here, there are several drag & drop options to create your interfaces very quick & easy.

Since this is an opportunity to show your creativity & innovative ideas, we won’t be giving you a template, but we will provide you with a step-by-step [**roadmap**](https://github.com/HTF-2022/SetupFioriFrontend/blob/main/appgyver_instructions/SAP%20AppGyver%20Instructions.pdf) towards an interesting interface. We will also provide some demo API’s to help you on your way!

**Demo API’s:**

* <https://fb65b981trial-dev-setupfiorifrontend-srv.cfapps.us10.hana.ondemand.com/v2/service/htf2022/FlowStream>
* <https://fb65b981trial-dev-setupfiorifrontend-srv.cfapps.us10.hana.ondemand.com/v2/service/htf2022/GandalfQuote>
* <https://fb65b981trial-dev-setupfiorifrontend-srv.cfapps.us10.hana.ondemand.com/service/htf2022/FlowHint>

# Deployment

When you’re done programming, your application can be deployed to the cloud using SAP Business Technology Platform (BTP).

Follow these steps to get your app up and running in the cloud in no time!

* Setup a free [BTP-Trial account](https://developers.sap.com/tutorials/hcp-create-trial-account.html).
  + Be sure to select US as region.
* Run cf login -a https://api.cf.us10.hana.ondemand.com and follow the process
  + This will log you in to SAP BTP Trial
* Run npm run deploy:dev
  + This will deploy your app to the cloud!
* Setup[SAP HANA Cloud](https://developers.sap.com/tutorials/btp-app-hana-cloud-setup.html)
  + Only **Step 3** is necessary!
  + Use **SetupFioriFrontend** as name