Containerisation Task Report

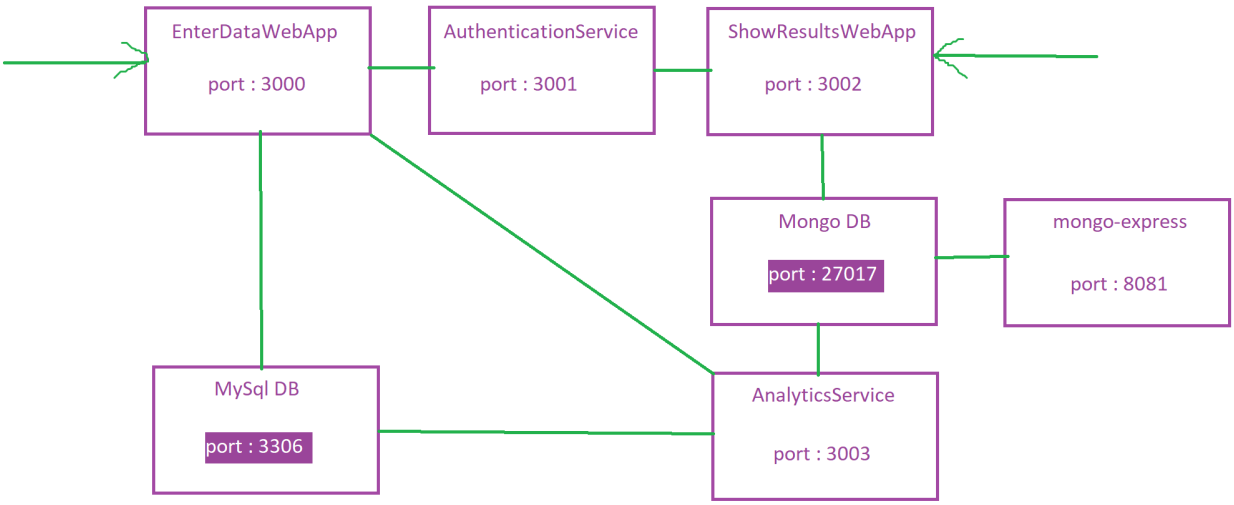
ESSAM ABDO

Contents

Overview

My project is basically a system that get **ages** from authorized users , using **EnterDataWebApp** , and shows the **average of ages** using **ShowResultWebApp** .

The figure below demonstrates the structure of the whole system :



Enter Data Web Application

A web Application , hosted on **port 3000** , that is used to get the age of the user , but before getting the age from the user , a login page is displayed to make sure that the user is authorized .

After taking the input from user , the credentials will be sent to **authentication service (port 3001)** to make sure that the user is authorized , if the credentials are right , the user will be directed to **Enter age page** , otherwise , the user will be directed to login page again .

After taking the age from the user , the age will be sent to **port 3306 (MySQL DB)** to be inserted in the database .

This app is developed using Node JS .

Authentication Service

It is a service that used to authenticate credentials , it takes email and password and redirect the result back depending on the credentials .

It validates the credentials statically .

It is hosted on **port 3001** . The app is developed using Node Js .

MySQL Database

This container **is not mapped** on any port . It is connected to other containers through the **default network of docker-compose** . Other containers can access the DB in this network using **port 3306** .

The database has one table called **Ages** with two columns **id** and **age** .

Analytics Service

This is a service to make the some analysis on **MySqlDB data** , this service is triggered whenever **EnterDataWabApp (port 3000)** makes a change on **MySqlDB** container .

This service is hosted on **port 3003** .

This application is made in Node JS .

MongoDB

This container has **MongoDB** in it , and like **MySqlDB** , **it in not mapped to any port** for the host . It is connected to other containers through the

default network of docker-compose . Other containers can access the DB in this network using **port 27017** .

The database consists of one collection called **age_avg ,** each document has two attributes , **id** and **avg** .

Show Results Web Application

This app is used for displaying the results of **analytics service (port 3003)** , before viewing the results the user must enter his / her credentials . Credentials are sent to **authentication service (port 3001)** , to check if the user is authorized , and if so , the user will be directed to a page to see the results . Results are brought from **MongoDB (port 27017)** .

The application is **mapped to port 3002** .

The application was developed using Node JS .

Mongo-Express

Mongo Express is an interactive lightweight Web-Based Administrative Tool to effectively manage MongoDB Databases . Written with Node Js , Express , and Bootstrap3 , Mongo Express can be used to simplify several MongoDB Admin tasks . Using Mongo Express , you can add , delete or modify databases , collections , and documents .

I used this tool to easily view the content of **MongoDB** database .

This container was mapped to **port 8081** .