

## **Assignment 4 - 3-Tier application with React**

This assignment has five parts and focuses on the development of a simple posting system based on nodejs, React & mysql. In this assignment, you will train:

- the development of a stateful React application,
- use of asynchronous fetch
- testing of your code

Please provide the Dockerfile and/or docker-compose.yml you used in completing the assignment. Failure to provide the needed Dockerfile and/or docker-compose.yml will make it impossible to test the solution you developed and result in 0 marks for this assignment.

### **Part A) 10 Points**

Develop a post method that will create a database in mysql designed for storing posts in a table called posts. A post consists of ID, Topic and Data. Make sure that the ID is assigned automatically by the database (should be the primary key). In case the database already exists the method should delete all existing entries in the table

### **Part B) 20 Points**

Develop a post and a get method that allows the addition of a new post and the retrieval of all existing posts.

### **Part C) 60 Points**

Using React develop a website that enables a user to

- Initialize/Create the database and the table posts.
- Create a new post consisting of topic and data.
- List all existing posts

Make sure you provide a Post, PostID, PostTopic and PostData tag.

To improve usability design your system, make sure to offer:

- A landing page
- A page for showing all postings
- A page for creating new posts

Feel free to use `react-router-dom`.

**Part D)            10 Points**

Write a test report that shows how you verified/tested that parts A, B and C work as intended,

**What to hand in?**

*The docker-compose.yml file that you used in the assignment and any additional dockerfiles you developed. Failure to provide this/these files will result in a zero (0) grade for the assignment.*

Parts A&B&C) -> The modified App.js file, all .js & .css files **you** developed/created.

Part D) -> one file called report.pdf that contains your report.