CSCI 4145 / 5409 – Summer 2020

Assignment 5

Technologies used: As before plus Model View Controller (MVC) pattern, used in building web applications through Sailsjs framework, and ORM using either Waterline framework or an ORM framework of your choice .

In this assignment you will be using the Sailsjs framework to build a web application based on the MVC pattern. Your task is to create a web service for management of jobs (as was done in the previous assignment) and also the front-end. If you did not use the Sailsjs framework thus far, it is strongly recommended that your first follow the tutorial provided on that topic. Data should be stored in a relational DB, which may be local or may be on cloud (managed or not), while access to the DB needs to be through *Waterline* ORM framework that is already a part of *sailsjs*. However, instead of the *Waterline* ORM, you may use ORM of your choice as long as you indicate clearly which one and briefly point to a couple of examples in your code how it is used.

As you will be required to use Sailsjs in your subsequent assignment(s), in which you will also be required to have your software running on clouds, to satisfy *basic functional requirements*, this assignment may be executed locally only (e.g., your desktop/laptop), i.e., you need not deploy your software in a container or on cloud. However, to receive a grade of A, besides your submission begin in top shape, you also need to deploy your software in a container(s) and/or on cloud. For A+, you need the software being deployed in a container on cloud.

Submission Requirements

- *Code and Data Requirements*
  - Your JS code must be such that
    - Any identifier in your code (including function names for non-anonymous functions) needs to end with the last three digits of your Student ID. Exceptions are identifiers such as i in looping constructs. For instance, my Banner/Dal ID ends with "007" and therefore any name of an identifier needs to end in "0007".
    - Any values for IDs of parts or jobs must end with the three digits of your ID. For instance, when creating an array containing the job information with two objects would be:
  - let jobs007 = [{ 'j1007', 1007, 55 }, { 'j2007', 2007, 66}] … where 'j1007' and 'j2007' are job IDs; 1007 and 2007 are part IDs; and 55 and 66 are quantities
  - If the above is not satisfied – the submission gets F

- *Submission file name(s) requirement*: When submitting a file, its name should be "A-X-Team#-StudentID-LastName.zzz", where
  - X … is the assignment number (5 in this case)
  - Team# … is a two digit ID/number of your group/team (enter "00" if the group/team has not been formed as yet)
  - StudentId … is your Dalhousie student (B00…) number
  - LastName … is your last name as it appears on you on Brightspace
  - zzz … is a file extension, e.g., 'zip' for a zipped file or pdf, docx, …)

- In a separate section, description of any deficiencies related to the assignment's functional requirements.

- Optionally, in a separate section, brief description of what you did in excess of the basic requirements. Use this to highlight what you did particularly well or in excess of the requirements. In particular, if you are proud of your work and feel it deserves more than satisfying requirements (B+), highlight here why. If you are striving for the A or A+, here is where you highlight your container /cloud deployment.

- Brief description of the major steps you performed to achieve your task – as an example of major steps, see tutorials. Your description should be supported by judiciously taken screenshots that demonstrate the results of these steps (and that they worked). For each step that you took, if appropriate refer to the documentation that you followed. (Also include in your appendix your code for web service routers/endpoints (in a textual form, not as a screenshot)). BTW, your description should refer to any resources that you consulted and used/followed to perform your major steps. If you were a regular sailsjs user prior to the course and did not need to consult any references on its (sailsjs) use, note it here while including a brief (2-3 – lines long) description of context of your previous experience with sailsjs. Of course, if you deployed your software in a container on cloud, major steps followed and references to resources you followed also need to be described.

- Screenshot of your project directory similar to as it appears in Figure 2 of the tutorial *A4-Azure-4145-5409sum2020.pdf*. If you used containers and/or cloud for deployment, screenshots evidence need to be provided as in previous assignments.

- The results of testing supported by screenshots that demonstrate that your software works.

- Make sure that you are prepared to test your software after your submission as your TA/marker may ask you for a live demo/testing of your assignment, including running and invoking your software. In such a case your TA/marker would contact you and arrange for a mutually convenient time to meet on MS Team call to start the demo (of course, the mutually convenient time would have to be within a reasonable time-frame (i.e., you may not delay the demo too much). She/he may also ask you to examine/review your environment pertinent to the assignment, such as content of some of your directories.

Once the grades for this assignment are released, you will have 3 days to file a request for a grade review of your assignment.