Sri Lanka Institute of Information Technology

B.Sc. Honors Degree in Information Technology

Specialized in Software Engineering

Final Examination
Year 3, Semester 1 (2021)

SE3040 – Application Frameworks

Duration: 4 Hours

June 2021

Version – Student ID ends with 0 (zero)

Instructions to Candidates:

- This paper has 1 question.
- The total mark for the paper is 100.
- This paper contains 4 pages, including the cover page.
- Please see the instructions at the end of the paper

You have been asked to develop a web application for a Hotel. After the discussions with related parties, the product team has come up with the following noticeable points.

1. The system is built mainly around categories and rooms. Categories and rooms share a many-to-many relationship. See the example below.
    a. Rooms- RM102, RM203
    b. Categories- Sporty, Romantic, Nature, Rustic
2. A web interface is required to display all categories and rooms belongs to a category. You can come up with your own design.
3. 3rd party vendors are interested in accessing following data.
    a. Get all categories.
    b. Get all rooms.
    c. Add new room.
    d. Get rooms in each category.

    All the above services and not CPU heavy (please consider this when you are selecting the technology).
4. Another special service is required to calculate the total amount of given set of rooms. This is calculated by adding all rooms amount together. This service is expanding and will be a CPU heavy task, please consider this when you are selecting the technology.

Note the following points as well.

1. RESTful services are required for the following.
    a. Get all categories.
    b. Get all rooms.
    c. Get rooms by category.

d. Add new room- Pass all the categories that a room belongs to as a list of IDs.
   Ex: {
   code: “RM102”,
   amount: 40000,
   wing: “west”,
   pax: 3
   categories: [Category ID]
   }
e. Get rooms in each category.

2. Room and Category data structure should be flexible. Please note the below example data structures.
   a. Room
      i. Code- RM102
      ii. Amount- 40000
      iii. Wing- Possible values (west, east, north, south)
      iv. Pax- Possible values (2,3.4)
   b. Category
      i. Name- Rustic
      ii. Description- Rustic experience
3. There is no concern for transaction control and consistency in this system.

Architect suggests from following technologies, and you must select the best suitable technology depending on the requirements being provided **(You do not need to use all the technologies).**

1. ReactJS
2. Node JS/Koa JS/Express JS
3. MongoDB
4. Spring Boot

Deliverables

Please follow the steps to upload the deliverables to the submission link.

1. Create a folder named your student ID number.
2. Create two folders name Frontend and Backend inside the created folder above.
3. Start the frontend and backend development in the relevant folder.
4. Make sure there are no compilation or run time issues in your ReactJS project.
5. Remove the node_modules folder from the project frontend folder structure.
6. Make sure there are no compilation or runtime issues in the **backend** project.
7. Remove the node_modules folder from the project backend folder structure.
8. If you develop the backend application using Spring Boot, **keep only the relevant** files including the source folder and the pom file.
9. Create a PDF document including the following,
    a. Screenshots of all user interfaces of the frontend application.
    b. Screenshots of all MongoDB collections.
    (Please see the template report provided.)
10. Put the PDF file in the frontend project folder.
11. Zip the project folder with the student id number (including both frontend and backend) and upload using the provided submission link.
12. Note that the zip file must be less than 30 MB. If the size is more than that try removing assets in the frontend project if you used any.

Marking Rubric

1. Implementation of User Interfaces
   - 24 marks
2. Implementation of get categories, get rooms, add rooms, and get rooms belongs to a category service endpoint.
   - 30 marks
3. Implementation of the calculation service.
   - 24 marks
4. Suitable technology selection (Student will be eligible for this mark if only he/she select the correct technologies for the different parts of the application).
   - 10 marks
5. At least one-unit test for either part of the application (UI / services).
   - 5 marks

6. Styling of the UI.
   - 2 marks.
7. Coding standards and quality
   - 5 marks

Note: Following are considered when awarding marks for Coding standards and quality.

1. Following the REST architecture (resource paths).
2. Variable, function, and class naming.
3. Clear code and directory structure.

--- End of the Paper ---