



Sri Lanka Institute of Information Technology

B.Sc. Honours Degree in Information Technology

Specialized in Software Engineering

Final Examination
Year 3, Semester 1 (2023)

SE3040 – Application Framework

Duration: 3 Hours

May/June 2023

Instructions to Candidates:

- ◆ This paper has 5 questions.
- ◆ Answer all questions in the booklet given.
- ◆ The total mark for the paper is 100.
- ◆ This paper contains 5 pages, including the cover page.
- ◆ Students are not allowed to use the internet.
- ◆ Necessary documentation will be provided in the netexam.

Question 01**(10 marks)****Spring boot & Node.js**

John is a senior developer at a startup company that is building a web application to help people find and book fitness classes online. The company has already decided to use React for the frontend development, but now John needs to decide on the best backend development technology to use.

He is considering two options: Spring Boot and Node.js. Both technologies have their pros and cons, and John needs to weigh them carefully before deciding.

- a) Compare the factors that should be considered when you select a technology stack for backend development.

(5 Marks)

- b) Suggest a suitable technology stack for the given scenario and justify.

(5 Marks)

Question 02**(20 marks)**

The team at ABC Company has been tasked with developing a new REST API for their e-commerce platform. The API will allow third-party developers to access and manipulate data from the platform, such as product information, customer orders, and inventory levels.

The team begins by conducting a thorough analysis of the existing platform's data structures and business logic. They identify the key data points that need to be exposed through the API and determine the most efficient way to organize and present that data.

The company has selected the NodeJS framework Express JS to develop the API.

The code given below contains two end points.

```
const express = require('express');
const app = express();

app.use(express.json());

app.get('/products/:id', (req, res) => {
  const productId = req.params.id;
  const product = db.get('products').find({ id: productId
}).value();
  if (product) {
    res.status(204).json(product);
  } else {
    res.status(403).json({ error: 'Product not found' });
  }
});
```

```
app.post('/orders', (req, res) => {
  const order = req.body;

  if (!order || !order.items || !order.customer) {
    res.status(401).json({ error: 'Invalid order data' });
    return;
  }

  const orderId = db.get('orders').push(order).write().id;
  res.status(200).json({ id: orderId });
});

app.listen(3000, () => {
  console.log('Server listening on port 3000');
});
```

- a) Asses the usage of the middleware in the given code.
(5 Marks)
- b) What is your opinion on http status code usage in the given code. Critically evaluate.
(8 Marks)
- c) Suggest a REST API Authentication method for the above project and justify your suggestion.
(3 Marks)
- d) Considering the SOLID principles, assess the design patterns or techniques that need to be implemented to ensure the code was maintainable, extensible, and easy to test?
(4 Marks)

Question 03**(20 marks)**

Answer the following questions below on Maven, Spring and Spring Boot.

- a) Defend how Maven simplifies the process of managing project dependencies and building applications in Java-based projects. (3 Marks)
- b) Discover the role of the pom.xml file in Maven and how does it help to manage project dependencies and build settings. (3 Marks)
- c) Interpret Dependency Injection (DI). (4 Marks)
- d) Explain how you would use the features of Spring Boot to create a RESTful AP (3 Marks)
- e) Justify how Spring Boot simplify the development and deployment of microservices (4 Marks)
- e) Assess the challenges of microservices that addresses by Spring Boot. (3 Marks)

Question 04**(25 marks)**

You are working on a project that requires building a backend system for an online bookstore. The system should allow users to browse, search, and purchase books. As a backend developer, you are tasked with building the backend system using either Express.js or Spring Boot.

Build a RESTful API using either Express.js or Spring Boot that allows users to browse, search, and purchase books. The API should be able to perform the following actions:

- a) List all available books with their title, author, price, and availability. (10 Marks)
- b) Search for books by title, author, or genre. (10 Marks)
- c) Purchase a book by ID and process the payment. (5 Marks)

You should be able to perform all these actions using HTTP requests. No need to connect the app with MongoDB.

Question 05**(25 marks)**

You have been tasked with building a frontend application for an online bookstore using React. The backend system for the bookstore has already been built and you will be using the RESTful API to connect to the backend. The API provides the following endpoints:

- GET /books - Returns a list of all available books with their title, author, price, and availability.
- GET /books/search?q={query} - Returns a list of books matching the search query by title, author, or genre.
- POST /books/purchase/:id - Purchases a book by ID and processes the payment using a third-party payment gateway.

Your task is to build a frontend application that allows users to browse, search, and purchase books from the online bookstore. The application should have the following features:

- a) A home page that displays a list of all available books.
(5 Marks)
- b) A search page that allows users to search for books by title, author, or genre.
(5 Marks)
- c) A book details page that displays the details of a selected book and allows users to purchase the book.
(5 Marks)
- d) A shopping cart page that displays the books that have been added to the cart and allows users to checkout and complete the purchase.
(5 Marks)
- e) Coding standards are evaluated as well.
(5 Marks)

You should use React to build the frontend application and connect to the RESTful API to retrieve and manipulate data. The application should be responsive and user-friendly with appropriate error handling and validation. You may use any additional libraries or frameworks as needed.