

# PROG 2200 ASSIGNMENT 03

## REST API ENDPOINTS

**Due Date: 08<sup>th</sup> Nov** (See due date designated on the assignment on D2L.)

Late submissions will receive the standard late submission penalty as stated in the course outline. (5% overall deduction per day late).

### Assignment Instructions:

Use IDE to create Java classes for REST API Endpoints using Spring boot framework along with entity classes as per the database schema for the given problem. **You must create .java files in this assignment.**

- Use the REST Resource Naming Conventions to design the endpoint URLs and methods.
- Ensure that the endpoints adhere to the principles of RESTful API design, including the use of nouns for resource names, plural nouns for collection names, hyphens to separate words and lowercase letters.

### Submissions:

When you are finished, commit your all files (java files) to GitHub as your submission for Assignment 3. Also, share the screenshot of the outcome on the word document file name like [YourName]\_PROG2200\_Assignment3\_Output\_[ScenarioX].docx.

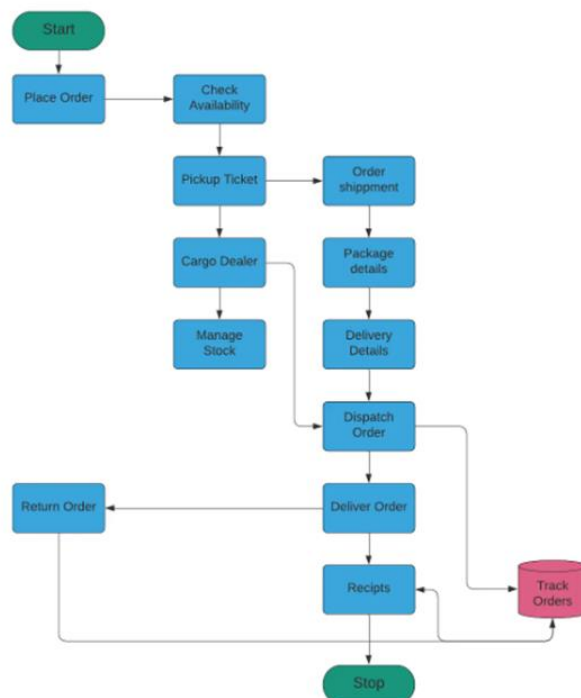
## Program 1 – REST APIs Endpoints

### Background

Online Courier systems have improvised the shopping experience to an exponential level. Be it your lunch or your favorite electronic gadget, you can place an order for it online, and the product will be delivered in no time. This kind of application is used by any product and service-based company like Amazon, Uber, and much more. Experience in designing, developing, and deploying such an application could make you stand out and make it easy for you to get into your dream company.

Online Courier Service project deals with the 'Courier Service Management System.' The system will be used for day-to-day activities.

- Product
- Buy a Product
- To return a product
- Payment Window
- company details
- hub rates
- Delivery Details
- Non-delivery Details
- Reviews



## Able to create REST API endpoints

We need to make our own rest APIs to make the entities and repos, connect it to a database, and use post/get request to return or insert data.

```
@RestController
public class StoreController {
    3 usages
    private final ShipmentRepository shipmentRepository;

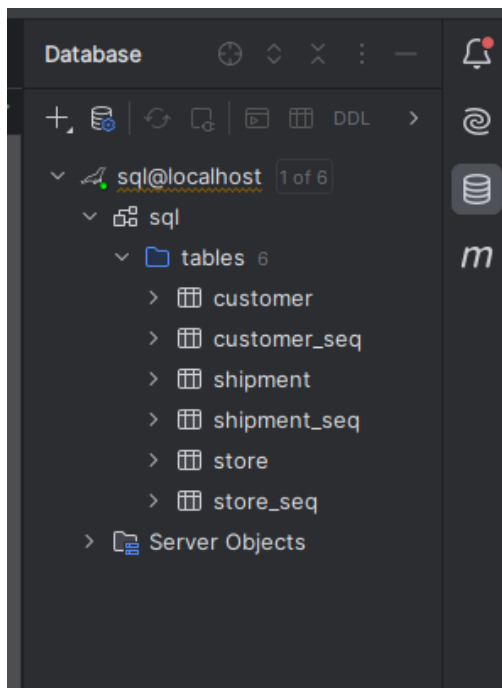
    public StoreController(ShipmentRepository shipmentRepository) { this.shipmentRepository = shipmentRepository; }

    @PostMapping(value = "/shipment/new")
    public ResponseEntity<Shipment> createPurchase(@RequestBody Shipment shipment) {
        shipmentRepository.save(shipment);
        return new ResponseEntity<>(shipment, HttpStatus.CREATED);
    }

    @GetMapping(value = "/shipment/list")
    public ResponseEntity<List<Shipment>> listPurchase() {
        List<Shipment> shipment = shipmentRepository.findAll();
        return new ResponseEntity<>(shipment, HttpStatus.OK);
    }
}
```

## Able to create database schema

**What is a database schema?** A database schema defines how data is organized within a relational database; this is inclusive of logical constraints such as, table names, fields, data types and the relationships between these entities. (<https://www.ibm.com/topics/database-schema>)



## Read Me File & Presentation

Refer to the ReadMe File.

GitHub Link: <https://github.com/Fall2024-NSCC-ECampus/assignment-3-rest-api-endpoint-cadalac-don.git>