

## Assignment Title: Real-Time Order Management System with AWS and CI/CD

---

### □ Objective:

Build and deploy a full-stack Order Management System with the following stack:

- **Frontend:** React.js
  - **Backend:** Spring Boot (Java) / Node.js
  - **AWS Services:** DynamoDB, S3, SNS
  - **CI/CD:** GitHub Actions or Jenkins
- 

### □ Backend Requirements (Spring Boot/Node.Js)

#### □ Entities:

##### Order

- `orderId` (UUID)
- `customerName` (String)
- `orderAmount` (Double)
- `orderDate` (ISO Timestamp)
- `invoiceFileUrl` (String, S3 URL)

#### □ Endpoints:

- `POST /orders` – Create new order (store in DynamoDB, upload PDF to S3, send SNS notification)
- `GET /orders/{id}` – Fetch order by ID
- `GET /orders` – List all orders

#### ♣ AWS Integration:

- **DynamoDB:** Save order data
- **S3:** Upload invoice PDF and store URL
- **SNS:** Publish message on order creation (to simulate email/SMS)

#### • Optional:

- Swagger UI
- JWT Authentication

---

## ☐ Frontend Requirements (React.js)

### ☐ Pages:

#### ☐ Dashboard (/)

- Display all orders in a table
- Columns: Order ID, Customer, Amount, Date, Invoice Link

#### ☒ Create Order (/create)

- Form fields:
  - Customer Name
  - Order Amount
  - Upload Invoice (PDF)
- Submit form via `POST /orders`

#### ☐ Order Detail (/orders/:id)

- View order detail
- Download invoice

### ☐ UI Features:

- Responsive design
- Toast notifications
- Clean design with Bootstrap, Tailwind, or Material UI

---

## ☐ ☐ CI/CD Pipeline

### Option A: GitHub Actions

`.github/workflows/deploy.yml` should:

- Checkout repo
- Build with Maven
- Run tests
- Deploy to AWS (Elastic Beanstalk, EC2, or ECS)

### Option B: Jenkins

- Git clone + Maven build
  - Test and deploy using AWS CLI commands
- 

## ☐ Deliverables

1. Spring Boot project (/order-service)
  2. React.js frontend (/order-ui)
  3. CI/CD config (GitHub Actions or Jenkinsfile)
  4. README.md with:
    - AWS setup instructions
    - API references
    - How to run the app locally
  5. Swagger UI
- 

## ☐ Evaluation Criteria

Criteria	Weight
Spring Boot API + AWS Integration	25%
React.js Functionality + UI/UX	20%
CI/CD Workflow	20%
Documentation & Code Quality	10%
SNS Email Subscription for Notification	5%
Deployment to any Cloud VPS	10%

---

## ☐ Bonus Tasks

- JWT Authentication for Order APIs
  - Export order logs to S3 as JSON
  - Sales analytics chart on dashboard
-