It's an online Shopping Cart Application, the application allows customers to shop online electronic product and perform the payment online, it also provide features like managing products by price and by category.

Key features:

1. User authentication and Authorization
2. Product management: Create, update, delete and retrieve books from the database.
3. search and filtering
4. Shopping cart
5. Order management; which allow customer to view their orders
6. Restful APIs to support client server communication for retrieving and ordering products.

Technology to use:

- Java
- Spring boot
- Spring Cloud (Eureka service registry, API gateway, ...)
- Spring Security
- Spring Data JPA
- OAuth2 security
- MySQL, PostgreSQL, Mongo DB
- Maven for project management
- Git for version control and Github as a repository to host the source code.
- Spring Tool Suite IDE
- Zipkin
- Micrometer
- Prometheus

- Spring boot Actuator

- Docker for containerization

The application will be built using microservice architecture, it consists of 3 microservices (Product Service, Order Service, Payment Service) and each has its own database, I will use MySQL for Product Service, PostgreSQL for Order Service and MongoDB for Payment Service, it's per service database/database management system design pattern. I create a product using the product service and place an order from the order service then perform the payment using the payment service.

A user service that has the information about the user which is stored in the database, Oauth2 security service for authentication and authorization which will be done in the API Gateway service, and a config server where the common configuration will be stored remotely in a Github repository.