# System Requirements Specification

## For

# Library Services

Version 1.1

**IIHT Pvt. Ltd.**
fullstack@iiht.com

# Table of Contents



# Library Services

## System Requirements Specification

## 1 PROJECT ABSTRACT

Library Management (Spring Boot + Kafka) use case

Use case is designed to implement Kafka messaging service to interact between a producer and consumer:

1. Producer: Producer is a Spring Boot RESTful Application, exposing REST endpoint to add and edit a book for issue. The information received in Producer is pushed to Kafka service queue, where it is available for Consumers to use it.
2. Consumer: Consumer is a part of same Spring Boot application, which will fetch message from Kafka and displays the same on console.

**Following is the requirement specifications**:

| | Library Service |
|---|---|
| | |
| | |
| Producer | |
| | |
| 1 | Add a Book |
| 2 | Update a Book |
| | |

# 2 Producer

**REST Endpoints**

| URL Exposed | | Purpose |
|---|---|---|
| 1. /books | | Add a new book |
| Http Method | POST | |
| Parameter | Book | |
| Return | Response Entity Status | |
| 2. /books/{id} | | Updates the book data |
| Http Method | PUT | |
| Parameter | Id, Book | |
| Return | Response Entity status | |
| | | |

**Model**:

Book: Used to represent Book Model. Add required getter/setter methods.

**Service:**

KafkaBookProducerService: A Service Class That needs to be implemented for pushing kafka message whenever adding and updating the book request is received. This class must be used by Controller method.

**Kafka Config**:

**KafkaProducerConfig** : A config class that requires to add configuration for :

a. Representing the location of Kafka Server
b. Configuration related with key serializer.
c. Configuration related with value serializer.
d. Send the Book Entity as a part of message

**Topic:**

Topics to be used must be as follows:

a. For adding a new book: “addBook”
b. For updating the book: “updateBook”

# 3 Consumer

**Model**:

Book: Used to represent Book Model. Add required getter/setter methods.

**Kafka Config Class**:

KafkaConsumerConfig: Implement the following methods:

a. consumerFactory: Implement to add configuration and expose the Consumer Factory Bean.
b. kafkaListenerContainerFactory: Implement to expose the ConcurrentKafkaListenerContainerFactory Bean object.

Add required annotations wherever needed.

**Kafka Consumer Service:**

KafkaBookConsumerService: Implement the following methods:

a. listenAddBook: Implement the method to receive the add book message from kafka and display the received book information on console.
b. listenUpdateBook: Implement the method to receive the update book message from kafka and display the received book information on console.

Add required annotations wherever needed.

**In main method of application call the Service methods to consume kafka message and display same on console**

**Kafka Config:**

In application.properties file add required configuration to

a. Represent the location of Kafka Server
b. Configuration related with key serializer.
c. Configuration related with value serializer.
d. Set the offset reset to earliest.
e. Configure your respective group ID.

# 4 EXECUTION STEPS TO FOLLOW

1. **All actions like build, compile, running application, running test cases needs to be done using respective options of eclipse.**
2. **Mandatory: Before final submission run test cases for both producer and consumer**
3. **Kafka Installation can be found in C drive**