**Assignment completion details – Arun Kumar**

1. I am using Spring initializr to get started with project setup as shown below:

A screenshot of a social media post

Description automatically generated

1. Download and Install Postgres DB version 12.3.1 along with pgadmin 4
2. Add below settings in **application.properties**

#DB driver class

spring.jpa.properties.hibernate.dialect = org.hibernate.dialect.PostgreSQLDialect

#hibernate settings - to create table or columns if it does not exist

spring.jpa.hibernate.ddl-auto = update

# to view SQL statements in logs

spring.jpa.hibernate.show-sql = true

#DB connection strings

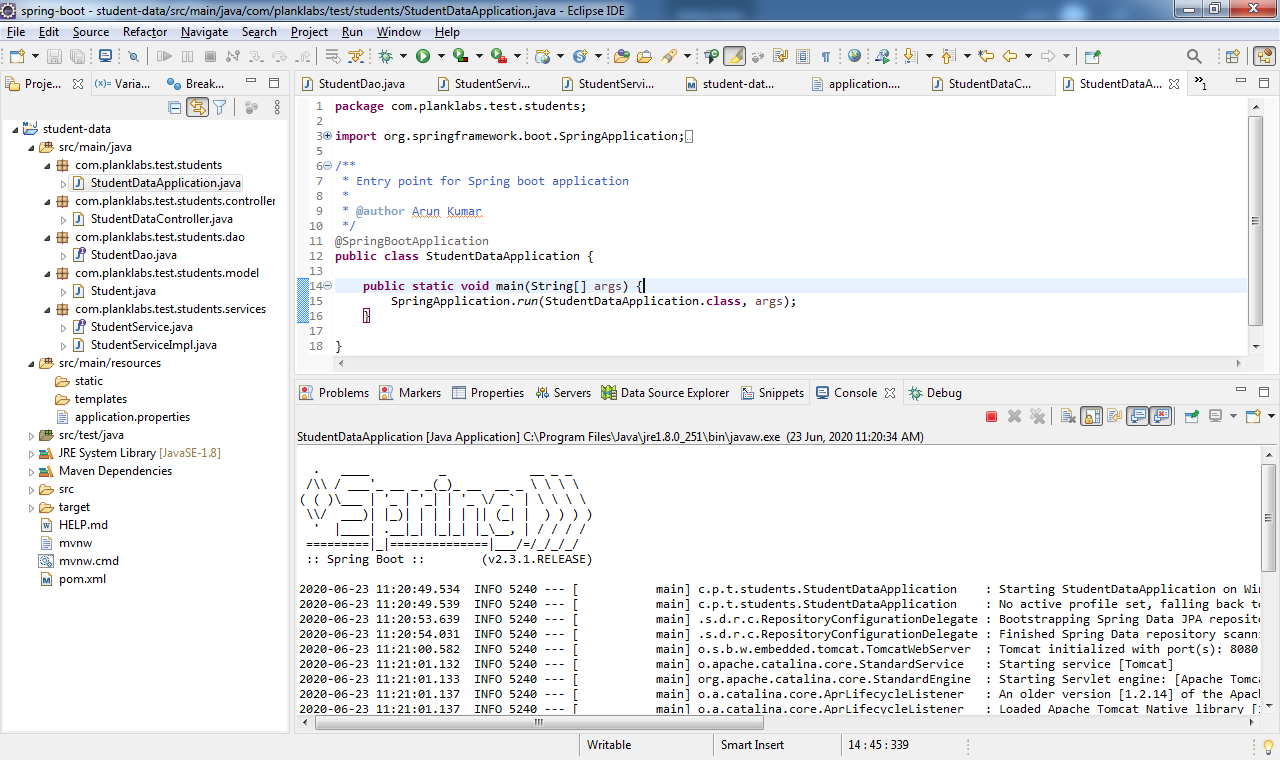
spring.datasource.url = jdbc:postgresql://localhost:5432/postgres

spring.datasource.username = postgres

spring.datasource.password = postgres

# use string representation of DOB instead of milliseconds

spring.jackson.serialization.write-dates-as-timestamps = false

1. Add RestController, Service and Dao dependency
   1. In Rest controller, create 2 endpoints with “planklabs” as base URL
   2. “insertStudent” as POST request endpoint
   3. “getStudentById/{studentId}” where “studentId” is a path variable
   4. Create a Service class which has Dao dependency
   5. Create a Dao interface (no implementation as this interface extends CrudRespoitory interface which has inbuilt implementations)
2. Start the application using StudentDataApplication class which has main method by running as Java Application
3. 
4. Download Postman for sending requests
   1. Send POST request as below to endpoint <http://localhost:8080/planklabs/insertStudent>

A screenshot of a social media post

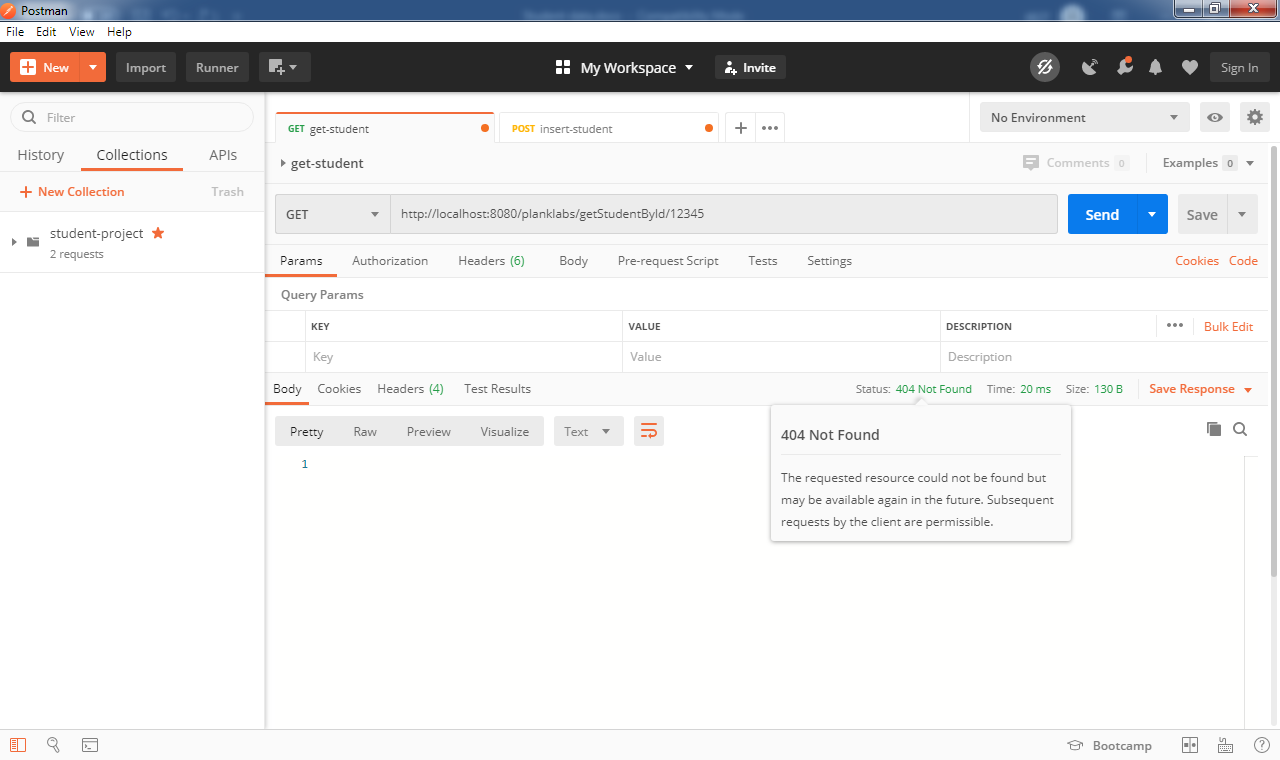
Description automatically generated

1. Send a GET request to read the student data by sending the ID got in previous request:

A screenshot of a social media post

Description automatically generated

1. Negative test case:
   1. Send a GET request with invalid student ID, the server returns 404 (Not Found)



* 1. Send POST request with invalid data, server returns 400 (Bad Request)

