Assessment for Implementation of REST API using java with Amazon RDS Database integration.

**Summary:**

Created REST API services using Spring boot as per the requirement and the required structure. The application used was IntelliJ IDEA 2020.2.2 (Community Edition) for the implementation of the project. Integration with Oracle Database as per the Credentials provided. This project is uploaded in the GITHUB profile and can be downloaded easily. Just basic requirement for this will be IntelliJ IDEA 2020.2.2 and maven. The project can directly be imported and executed on the local.

**Basic Structure**:

1.**Connectivity**: Application properties included with Oracle database connectivity and credentials. Dependencies added in pom.xml ojdbc7.

2.**Folder Structure**:

1) DepartmentEntity: Contains the definition of the table.

2) DepatmentRepository: To perform actions required as per entity.

3) DepatmentController: Holds the structure for Requestmapping and API calls.

This Structure is common for Department, Badge, Job\_titile, and Employee with different query conditions as per requirement.

4. **Tds\_Exception**: Hold the Runtime Exceptions.

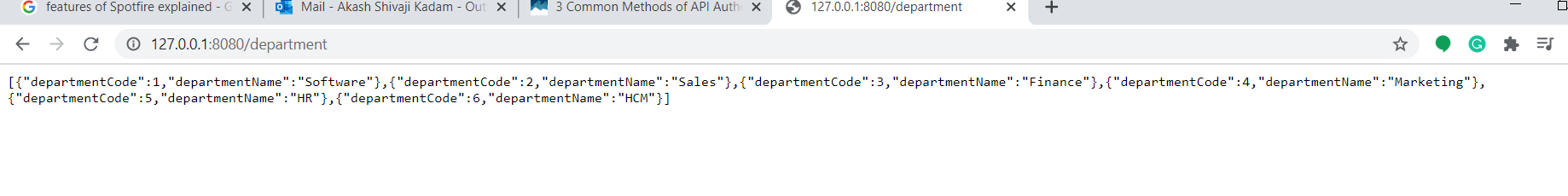
5. **CountryList**: This class contains the method to extract the name of the country when passed the URL code using EmployeeController.

**Implementation and Unit Testing Results:**

**1) Entity: Department**

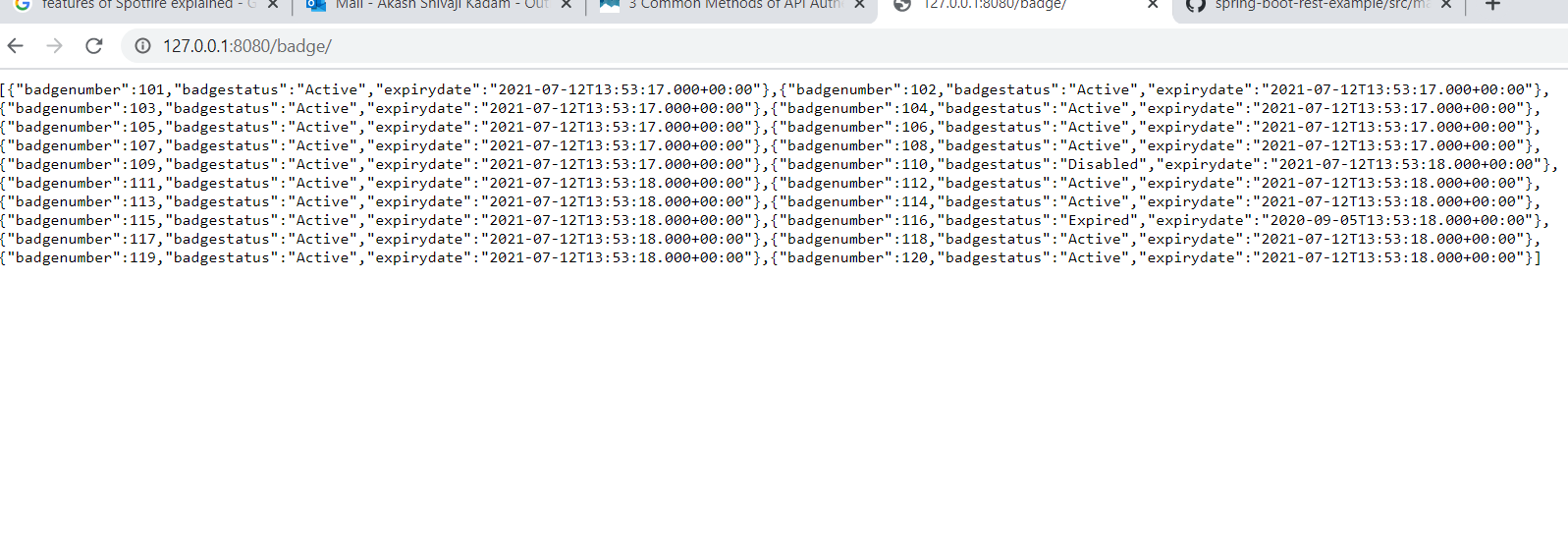
Request: GET (http://127.0.0.1:8080/department/)

Requirement: To Get all the records from the department table using API.

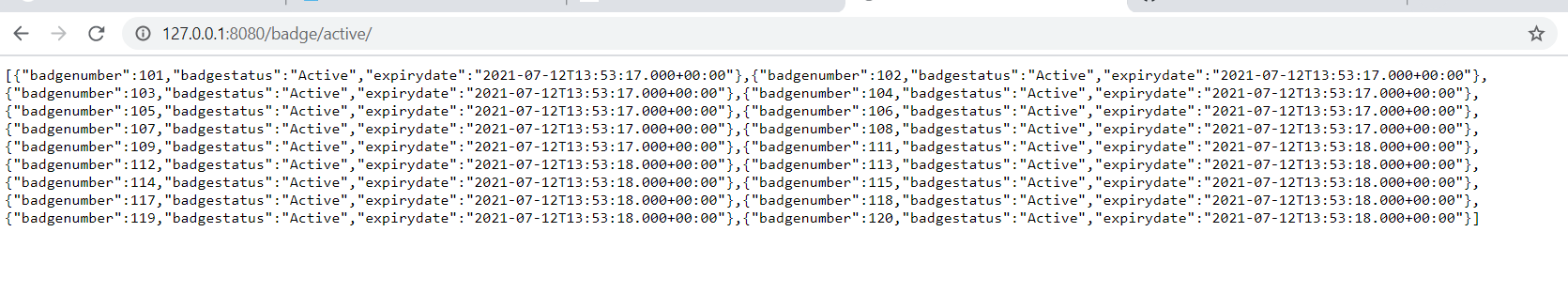


**2) Entity: Badge**

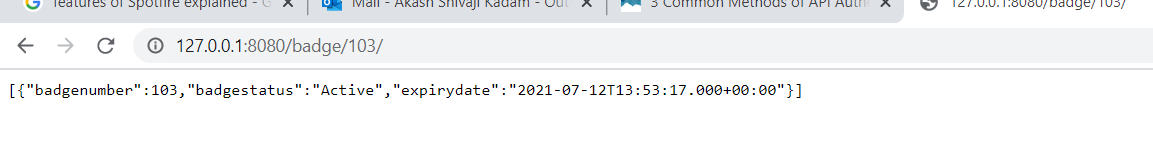
* Request: GET (<http://127.0.0.1:8080/badge/>)
* Requirement: To Get all the records from the Badge table using API.



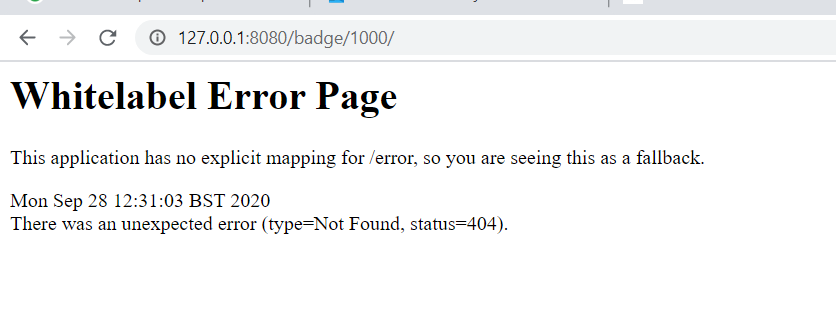
* Request: GET (<http://127.0.0.1:8080/badge/active>)
* Requirement: To Get all the records from the Badge records which are active and expiry date less than the current system date table using API.
* Query: “select \* from badge b where b.badge\_status = 'Active' and to\_char(b.badge\_expiry\_date,'dd/mm/rrrr') < to\_char(sysdate,'dd/mm/rrrr') ;”



* Request: GET ([http://127.0.0.1:8080/badge/{badge\_number}](http://127.0.0.1:8080/badge/%7bbadge_number%7d))
* Requirement: To Get all the records from the Badge table based in the badge\_number using API.

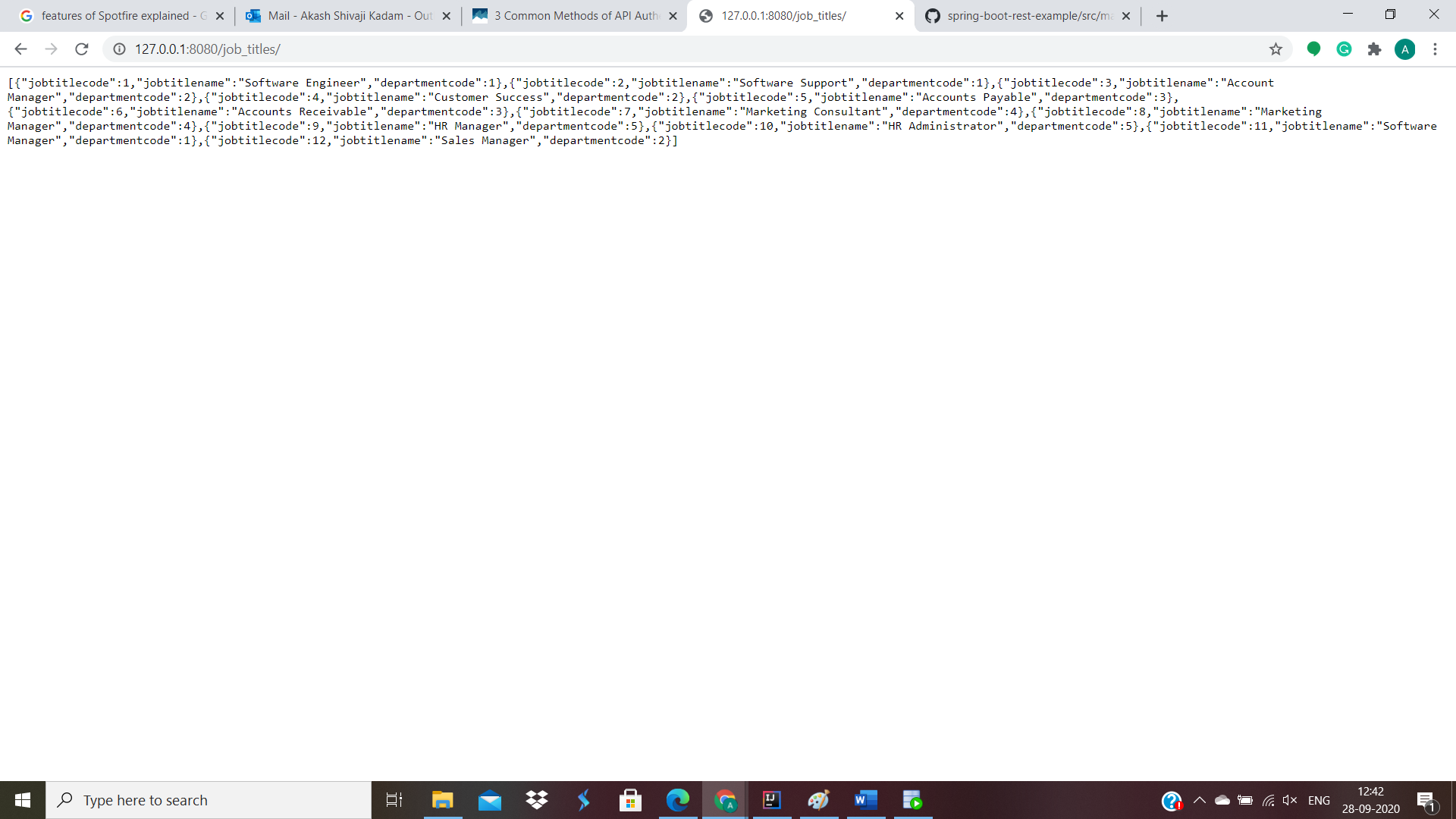


* Exception when badge\_number does not exist in the database:



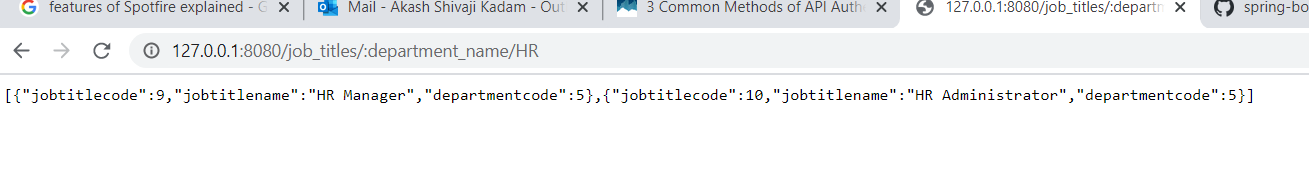
3) **Entity: Job\_Title**

* Request: GET (<http://127.0.0.1:8080/job_title/>)
* Requirement: To Get all the records from the job\_title table using API.



* Request:GET ([http://127.0.0.1:8080/job\_title/:department\_name/{department\_name}](http://127.0.0.1:8080/job_title/:department_name/%7bdepartment_name%7d))
* Requirement: To Get all the records from the job\_title table using API
* Query: “select \* from job\_title jt,department d where jt.department\_code = d.department\_code and d.department\_Name=:departmentname”

.



4) **Entity: Employee**

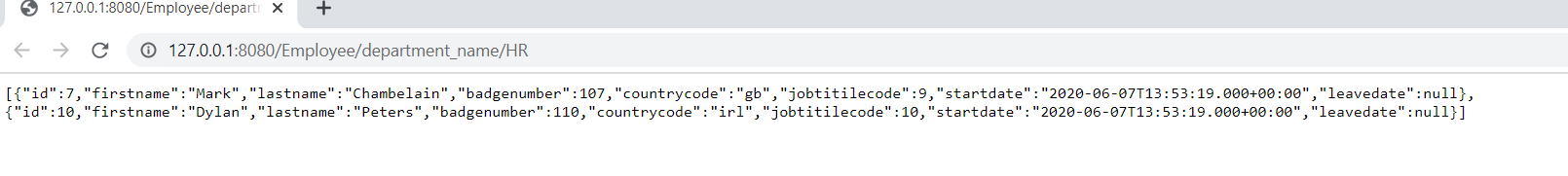
* Request: GET (<http://127.0.0.1:8080/Employee/>)
* Requirement: To Get all the records from the employee table using API. Additional, Country code to be replaced with Country name by consuming API to fetch the country name.



* Request: GET (<http://127.0.0.1:8080/Employee/active>).
* Requirement: To Get all the records from the employee table using API.
* Query: “select \* from employee e where e.LEAVE\_DATE is null or e.LEAVE\_DATE < sysdate”.



* Request: GET (<http://127.0.0.1:8080/Employe/department_name/>)
* Requirement: To Get all the records from the employee table using API.
* Query : “select e.\* from employee e , department d , job\_title jt where e.job\_title\_code = jt.job\_title\_code and d.department\_code=jt.department\_code and d.department\_Name=:departmentname”.



**Note:**

The code has been uploaded yo the GITHUB profile: **Kadamakash111**

**Link**: https://github.com/Kadamakash111/RESTAPI\_TDS\_ASSESSMENT

Please feel free if there are any queries.

**Gmail**: kadamakash111@gmail.com