

# Coding Assessment: JAVA

## 1 GUIDELINES

---

You are required to develop a Java application which will address the requirement indicated below.

- Language: Java 1.8 or higher
- Submission: A zip folder contains project files and other documents mentioned in the assessment or a GitHub public repo
- Deadline: Refer to the email

## 2 ASSESSMENT DETAILS

---

### 2.1 INTRODUCTION

"SmartPark" is a technology company developing an intelligent parking management system for urban areas. The company is aiming to optimize the use of parking spaces and facilitate easy navigation for drivers.

A **Parking Lot** has:

- Lot ID (Unique identifier, 50 characters max)
- Location
- Capacity (Total number of parking spaces)
- Occupied Spaces (Number of currently occupied spaces)

Each **Vehicle** has:

- License Plate (Unique identifier, allowed only letters, numbers, and dashes)
- Type (Car, Motorcycle, Truck)
- Owner Name (Letters and spaces only)

Develop a service exposing a REST API which allows clients to communicate with the parking management system.

The service should be capable of doing,

- Registering a parking lot
- Registering a vehicle
- Checking in a vehicle to a parking lot
- Checking out a vehicle from a parking lot
- Viewing current occupancy and availability of a parking lot
- Viewing all vehicles currently parked in a lot

\* You are allowed to make any assumptions in the design approach.

### **2.1.1 Functional Requirements**

- Prevent parking a vehicle in a full parking lot.
- Update the available spaces when a vehicle is checked in or out.
- Ensure that a vehicle can only be parked in one lot at a time.

### **2.1.2 Non-Functional Requirements**

- Input / Output data of APIs should be JSON.
- Your project must be buildable and runnable.
- Your project must have a README file with build/run/test instructions.
- If database is required, use in memory database type (Ex: H2).
- Any data required by the application to run must be preloaded in the database (Ex: reference data, dummy data).
- Junit tests are optional but advisable.
- Postman collection with APIs or document with curls to test the APIs.

### **2.1.3 Out Of Scope**

- API authentication is out of scope. All APIs can be no auth.
- Design documents are not required.