Toll Parking Engine 1.0.0 API

Documentation

Table of Contents

[API Details 1](#__RefHeading___Toc378_3740649519)

[Swagger Details 2](#__RefHeading___Toc380_3740649519)

[Java Doc Path 2](#__RefHeading___Toc382_3740649519)

[Api Dependencies 2](#__RefHeading___Toc384_3740649519)

[How to run 2](#__RefHeading___Toc386_3740649519)

[Sequence Diagrams 5](#__RefHeading___Toc388_3740649519)

[Packages 6](#__RefHeading___Toc390_3740649519)

[Hierarchy 7](#__RefHeading___Toc392_3740649519)

## API Details

|  |  |  |
| --- | --- | --- |
| **Resource** | **Request Method** | **Function** |
| /parking/vehicle/ | POST | Park vehicle in right slot |
| /parking/vehicle/{vehicleRegistration} | DELETE | Unpark vehicle from slot and generate bill |
| /parking/status | GET | Fetch current status of all praking slots |

## Swagger Details

<http://localhost:9090/v2/api-docs>

<http://localhost:9090/swagger-ui.html>

## Java Doc Path

toll-parking-engine/build/docs/javadoc/index.html

## Api Dependencies

Spring Boot 2+

Gradle 4.10+

Springfox-swagger2 2.9.2

springfox-swagger2-ui 2.9.2

## How to run

1. Clone project

~/java-project$ git clone <https://github.com/singh-kulwant/toll-parking-engine.git>

2. Build project

~/java-project/toll-parking-engine$ sudo gradle clean build

3. Start application

~/java-project/toll-parking-engine$ sudo gradle bootRun

4. Open uri in any Rest Client

* **To park vehicle**
  + Url: <http://localhost:9090/parking/vehicle/>
  + RequestMethod: POST
  + Sample RequestBody for standard gasoline vehicle:

{

"vehicleRegistration":"PO4534",

"vehicleType" : "GASOLINE"

}

* + Sample RequestBody for 50KW electric vehicle:

{

"vehicleRegistration":"CY345OD0",

"vehicleType" : "HEAVY\_ELECTRIC"

}

* + Sample RequestBody for 20KW electric vehicle:

{

"vehicleRegistration":"YT3828UR",

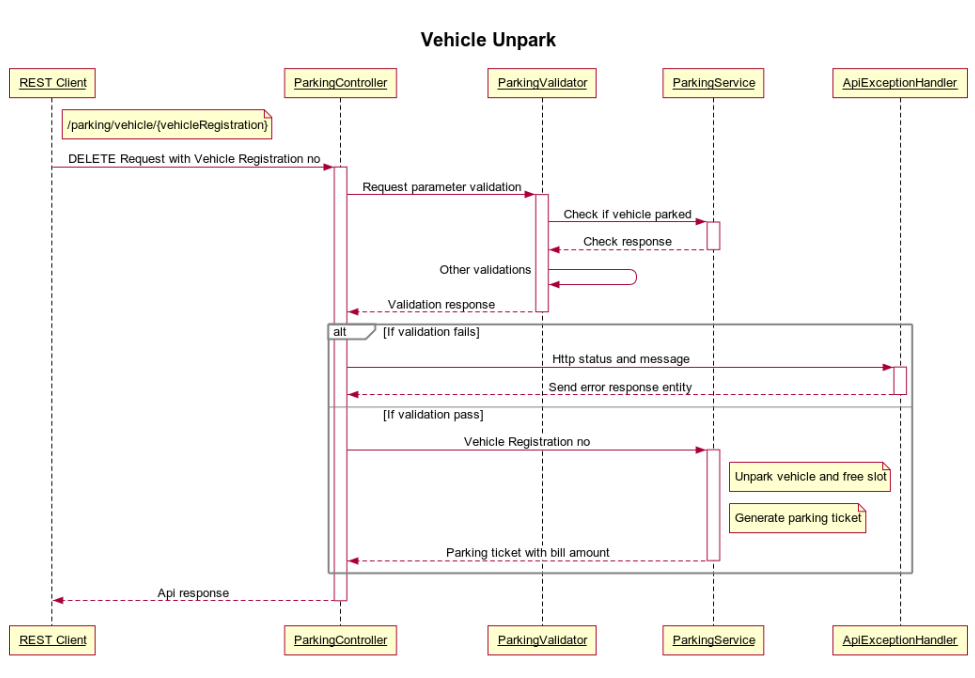
"vehicleType" : "LIGHT\_ELECTRIC"

}

**Note:** vehicleType is enum – possible values are [GASOLINE, HEAVY\_ELECTRIC, LIGHT\_ELECTRIC]

* **To unpark vehicle:**
  + Url: <http://localhost:9090/parking/vehicle/>{vehicleRegistration}
  + RequestMethod: DELETE
  + Sample request uri
    - http://localhost:9090/parking/vehicle/PO4534
* **To get live status of parking slots:**
  + Url:<http://localhost:9090/parking/status>
  + RequestMethod: GET

## Sequence Diagrams



## Packages

|  |  |
| --- | --- |
| **Package** | **Description** |
| com.parking.app | Main class package |
| com.parking.config | Parking engine configurations and handlers |
| com.parking.rest | Parking engine controller |
| com.parking.service | Parking service and validator |
| com.pricing.models | Pricing policy interface and implementations |
| com.slot.models | Parking slot models |
| com.vehicle.models | Vehicle classes and enums |

## Hierarchy

# Class Hierarchy

* java.lang.Object
  + com.parking.config.ApiMessage
  + com.pricing.models.FixedHourlyBilling (implements com.pricing.models.PricingPolicy)
  + com.pricing.models.HourlyBilling (implements com.pricing.models.PricingPolicy)
  + com.parking.config.ParkingConfigurationLoader
  + com.parking.rest.ParkingController
  + com.parking.service.ParkingService
  + com.slot.models.ParkingSlot
    - com.slot.models.HeavyElectricSlot
    - com.slot.models.LightElectricSlot
    - com.slot.models.StandardSlot
  + com.pricing.models.ParkingTicket
  + com.parking.service.ParkingValidator
  + org.springframework.web.servlet.mvc.method.annotation.ResponseEntityExceptionHandler
    - com.parking.config.ApiExceptionHandler
  + com.parking.config.SwaggerConfiguration
  + com.parking.app.TollParkingApplication
  + com.vehicle.models.Vehicle

# Interface Hierarchy

* com.pricing.models.PricingPolicy

# Enum Hierarchy

* java.lang.Object
  + java.lang.Enum<E> (implements java.lang.Comparable<T>, java.io.Serializable)
    - com.vehicle.models.VehicleType