

# Code Assessment: Scalable Tracking Number Generator API

## Objective:

Design and implement a RESTful API that generates unique tracking numbers for parcels. This API should be scalable, efficient, and capable of handling high concurrency.

## Requirements:

### 1. Setup:

- Scaffold a new project using the latest version of:
  - **Spring Boot** (for Java candidates)
  - **Django REST Framework** (for Python candidates)
- The project should follow best practices in structure and configuration.

### 2. API specification:

- Create an API endpoint accessible via **GET /next-tracking-number**.
- The API should accept the following query parameters:
  - **origin\_country\_id**: The order's origin country code in ISO 3166-1 alpha-2 format (e.g., "MY").
  - **destination\_country\_id**: The order's destination country code in ISO 3166-1 alpha-2 format (e.g., "ID").
  - **weight**: The order's weight in kilograms, up to three decimal places (e.g., "1.234").
  - **created\_at**: The order's creation timestamp in RFC 3339 format (e.g., "2018-11-20T19:29:32+08:00").
  - **customer\_id**: The customer's UUID (e.g., "de619854-b59b-425e-9db4-943979e1bd49").
  - **customer\_name**: The customer's name (e.g., "RedBox Logistics").
  - **customer\_slug**: The customer's name in slug-case/kebab-case (e.g., "redbox-logistics").

### 3. Constraints:

- The generated tracking number must satisfy the following:
  - It must match the regex pattern: **^[A-Z0-9]{1,16}\$**.
  - It must be unique; no duplicate tracking numbers should be generated.
  - The generation process should be efficient.
  - The system should be scalable, capable of handling concurrent requests, and should be able to scale horizontally.

### 4. Response structure:

- The API should return a JSON object containing at least the following fields:
  - **tracking\_number**: The generated tracking number.
  - **created\_at**: The timestamp when the tracking number was generated (in RFC 3339 format).

- You may include additional fields in the response if you see fit.

5. **Implementation considerations:**

- **Efficiency & concurrency:** The solution should be optimized for high performance and be capable of handling multiple concurrent requests without degradation.
- **Scalability:** The solution should be designed to scale horizontally across multiple instances.
- **Creativity:** You may use any or none of the parameters provided to generate the tracking number. Additional creativity in using the parameters or extending the functionality will be considered.

6. **Deployment:**

- Deploy your application to a publicly-accessible platform (e.g., Google App Engine, AWS, Heroku, or any other suitable free service).

7. **Deliverables:**

- **Source code:** Submit the full source code, including a `readme.md` file with instructions on how to set up, run, and test the application.
- **Deployed application:** Provide the URL of the deployed API.