R Software Developer Assessment

Thank you for your interest in the R Software Developer role. This assignment will allow us to evaluate your coding proficiency, problem-solving skills, and ability to work with real-world data. Dedicate no more than 3-4 hours to complete this task.

Task Overview

Build a simple JSON data API using {ambiorix} and process a dataset using {data.table}.

The API should expose endpoints that allow users to interact with the processed data.

- **Estimated Time:** 3 - 4 hours.
- **Submission Format:** GitHub repo.

Part 1: Data Processing

Dataset

Use the flights dataset from {nycflights13}.

Assume this is a large dataset stored in a database but for this task you can load it in-memory.

Tasks

1. Load the dataset and cast it into a data.table.
2. Perform the following transformations:
   - Compute the **average departure delay** for each airline.
   - Find the **top 5 destinations** with the most flights.
   - Create two new columns:
     - one which is a **unique id** for each row/flight.
     - and another indicating whether a flight was **delayed for more than 15 minutes.**
3. Save the processed data as a CSV or an SQLite database file.

Part 2: JSON Data API

Using Ambiorix, implement these endpoints:

1. POST /flight
   - Creates a new flight entry.
   - Accepts a JSON payload with flight details.

2. GET /flight/:id
   - Returns details of the flight specified by that id.
3. GET /check-delay/:id
   - Returns whether the flight is classified as "delayed".
4. GET /avg-dep-delay?id=given-airline-name
   - Returns the average departure delay of an airline.
   - Use query strings to access the given airline.
   - If no airline is provided, return all the airlines.
5. GET /top-destinations/:n
   - Returns the top n destinations with the most flights.
6. PUT /flights/:id
   - Update details of the flight specified by that id.
   - Accepts a JSON payload with the **new details**.
7. DELETE /:id
   - Deletes the flight with the specified id.

Implementation Notes

- The API should return JSON responses.
- If possible, store the processed data in SQLite and query it dynamically.

Evaluation Criteria

- Correctness & efficiency of {data.table} operations.
- Good architecture of the API.
- Code readability, efficiency & function documentation.
- Error handling & validation.

Submission Format

- Github repo link,
- Link to a screen recording of working API, explaining implementation rationale (you can use loom, zoom or obs).

Submission Deadline

- Provide all deliverables by **25th February 2025, 5 PM EAT**.
- Final projects should be submitted to recruitment@actserv-africa.com

For any questions or clarifications, feel free to reach out.