

# Rails Assessment

## A Note on Routing Services

We use the [Google Directions API](#) for routing and traffic information. You can use that for this exercise if you want. An alternative is the [open route service](#) for which you can create a free account connected to your Github.

## Definitions:

- The **driving distance** between two addresses is the distance in miles that it would take to drive a reasonably efficient route between them. It is not the straight line distance. It can be calculated by using a routing service
- The **driving duration** between two addresses is the amount of time in hours it would take to drive the driving distance under realistic driving conditions. It can be calculated by using a routing service
- The **commute distance** for a ride is the driving distance from the driver's home address to the start of the ride, in miles
- The **commute duration** for a ride is the amount of time it is expected to take to drive the commute distance, in hours.
- The **ride distance** for a ride is the driving distance from the start address of the ride to the destination address, in miles
- The **ride duration** for a ride is the amount of time it is expected to take to drive the ride distance, in hours
- The **ride earnings** is how much the driver earns by driving the ride. It takes into account both the amount of time the ride is expected to take and the distance. For the purposes of this exercise, it is calculated as:  $\$12 + \$1.50 \text{ per mile beyond 5 miles} + (\text{ride duration}) * \$0.70 \text{ per minute beyond 15 minutes}$

**Prompt:** The primary goal of this exercise is for you to demonstrate how you think about testing, readability, and structuring a Rails application. We are also evaluating your ability to understand and implement requirements.

## Specification:

- ☐ Create a Rails 7 application, using Ruby 3+
- ☐ Include the following entities:
  - ☐ Ride
    - ☐ Has an id, a start address and a destination address. You may end up adding additional information
  - ☐ Driver
    - ☐ Has an id and a home address
- ☐ Build a RESTful API endpoint that returns a paginated JSON list of rides in descending score order for a given driver

- ☐ Please write up API documentation for this endpoint in Markdown or alternative
- ☐ Calculate the score of a ride in \$ per hour as: **(ride earnings) / (commute duration + ride duration)**. Higher is better
- ☐ Google Maps is expensive. Consider how you can reduce duplicate API calls
- ☐ Include RSpec tests