Assignment task for Ruby developers

Description

- Given a list of fixed geographical areas in `GeoJSON` format, type `Polygon` (see attached example)
- The goal of the application is to determine whether an incoming geographical Location is **inside** at least one of the given areas

Task (Option 1)

- Create an API-only `Ruby on Rails` application
- The application implements the following use-cases via API endpoints:
  - Fetch the list of the given areas in `GeoJSON` format
  - Check if a given Location is **inside** any of the given areas:
    - IN: `GeoJSON` of type `Point`
    - OUT: **inside?** `true/false`
- The application should gracefully handle errors
- Automation tests with `RSpec`
- Put source code in `GitHub`
- Host an application in the cloud (e.g. Heroku) and provide a link

Task (Option 2)

- The same as Option 1 but without using any geo-utilities

Task (Option 3)

- Create an API-only `Ruby on Rails` application
- Incoming Locations are stored in DB (`postgres` or any other)
- The application implements the following use-cases via API endpoints:
  - Fetch the list of the given areas in `GeoJSON` format
  - Create a Location
    - IN: Location name (Text)
    - OUT: ID of the created Location
    - In the background determine the coordinates of the given Location and store in DB
  - Fetch Location by ID
    - IN: Location ID
    - OUT: Location name, coordinates, **inside?**: `true/false`
    - Gracefully handle errors (e.g. invalid input parameters, geocoding failed etc.)
- Use `Sidekiq` for background processing
- Automation tests with `RSpec`
- Use any of the external geocoding services (e.g. Google Geocoding API)
- Do not use any geo-utilities for identifying if a Location is **inside**
- Put source code in `GitHub`
- Host an application in the cloud (e.g. Heroku) and provide a link