Rails Code Challenge

Welcome to the DoubleGDP Coding Challenge!

Overview

To complete this challenge, you will need to write a simple Rails-based web app, and provide us the source files to be built.

The purpose of this challenge is to assess your skills and approach to composing a simple web app with CRUD capabilities given a set of screens. We will also assess the generated HTML, CSS, and JS output.

This challenge is expected to take about 2-3 hours.

The Challenge

It's pretty simple. You'll need to build a functional Rails app from scratch using the provided screens and explanation as a reference.

Although this is a basic exercise, we'll be looking for simple, well-designed, performant, and **tested** code in the submission. This is a backend Rails challenge, so concentrate on the MVC aspect of Rails.

Please include a `README` with setup instructions, and any tests or other documentation you created as part of your solution.

Also, be prepared to answer the following questions during the follow-up interview or/and add your responses to your `README`:

- How did you decide on the technical and architectural choices used as part of your solution?
- Are there any improvements you could make to your submission?
- What would you do differently if you were allocated more time?

- How long did it truly take you to complete the challenge? (this is for us for future reference to understand if we need to make adjustments to the challenge so that it fits in the allotted time frame.

>>> You must use RAILS 6 and SQLite as a database. <<<

Details

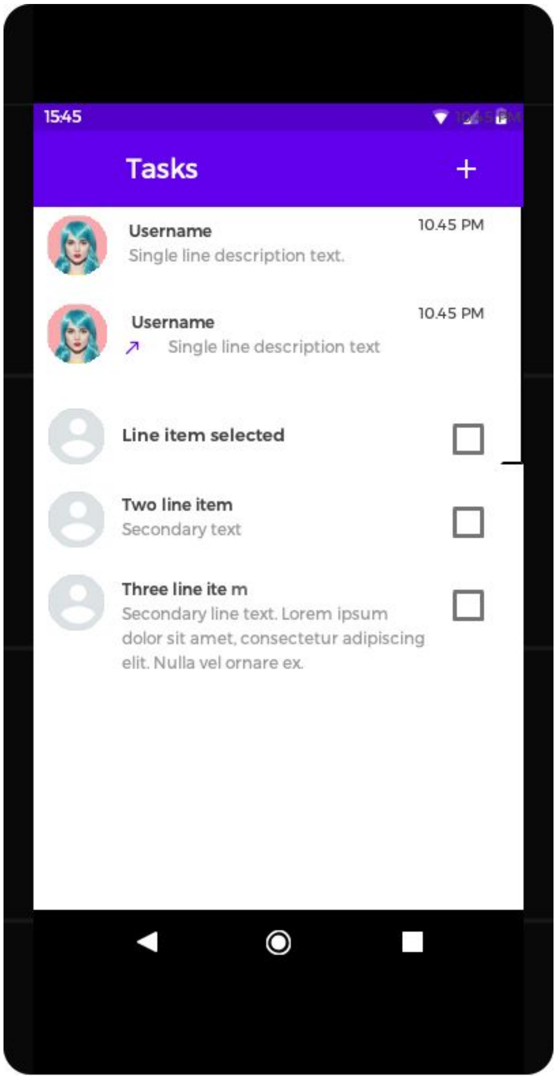
You will need to build the following 2 pages:

- A "tasklist" page
- A "taskadd" page

The pages should also be usable on mobile and tablet devices.

You can assume that you do not have to support legacy browsers without features such as `fetch` or `flexbox`.

"tasklist" Page


15:45
Tasks
+
Username
Single line description text.
10.45 PM
Username
Single line description text
10.45 PM
Line item selected
□
Two line item
Secondary text
□
Three line ite m
Secondary line text. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla vel ornare ex.
□

Refer to the screen on the left.

This will be your `starting page` screen.

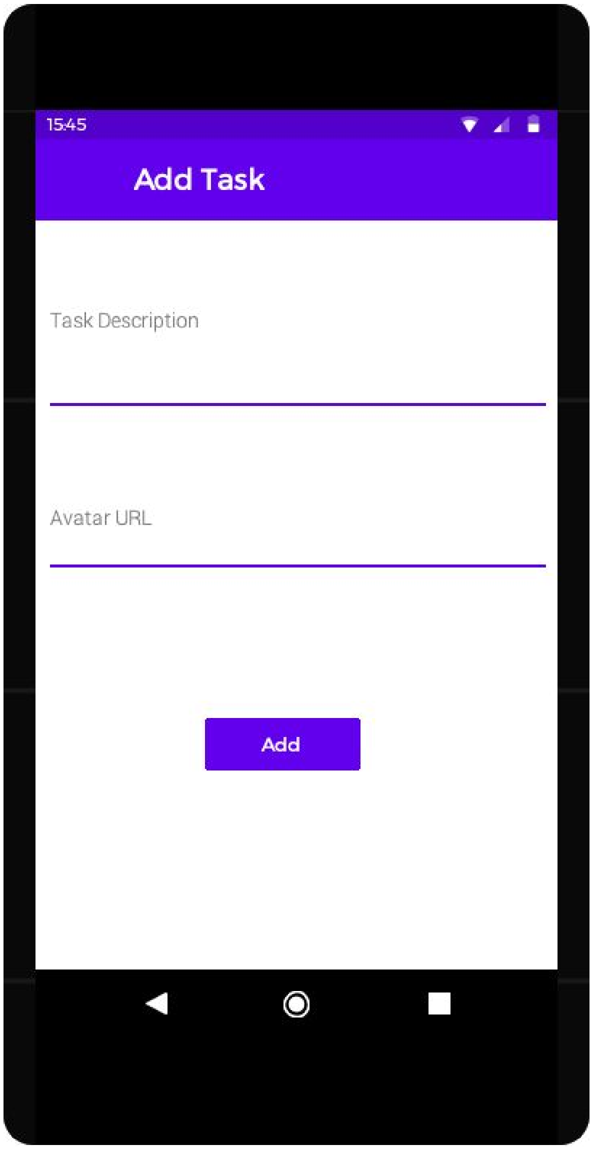
When clicking on + the “taskadd” screen should open

When new items are added, they will show with an avatar, a task description, and a checkbox.

When the checkbox is checked, the checkbox disappears and is replaced with the time of click. The database should be updated with the correct value.

If there are already Tasks in the database, the screen should open and list the tasks.

"taskadd" Pages



Refer to the screen on the left.

- You will need to be able to enter a task description (ie: I have to read the hobbit)
- You will need to be able to enter an avatar URL (ie: a small image file, could even be local if easier)

When Clicking “Add”, the task is added to the database and the “tasklist” page and the “taskadd” screen disappears.

Submission

You should submit your work on a GitLab or GitHub private repository and allow the hiring manager to access it.   If a private Gitlab / Github repository is not a viable option, contact your hiring manager to set up an alternative option.

FAQ

What language, framework, build tool... should I use?

You must use RAILS 6 and SQLite as a database.

You may use whatever you like as long as the solution is built using Rails. You are free to use any CSS, and javascript libraries you prefer.

We also prefer the use of minimal dependencies. This is a simple project designed to showcase your skills. Please, use Gems and Libraries with moderation.

Good luck