

Frontend developer test v3

Overview

This task involves creating an Angular webpage using the provided design wireframe.

The task needs to be completed following the specifications as closely as possible. Best practices for every language should be adhered to and the development process should resemble how you would develop in an actual work environment (using TDD, committing to VCS, etc).

You can always ask for guidance or details to be specified.

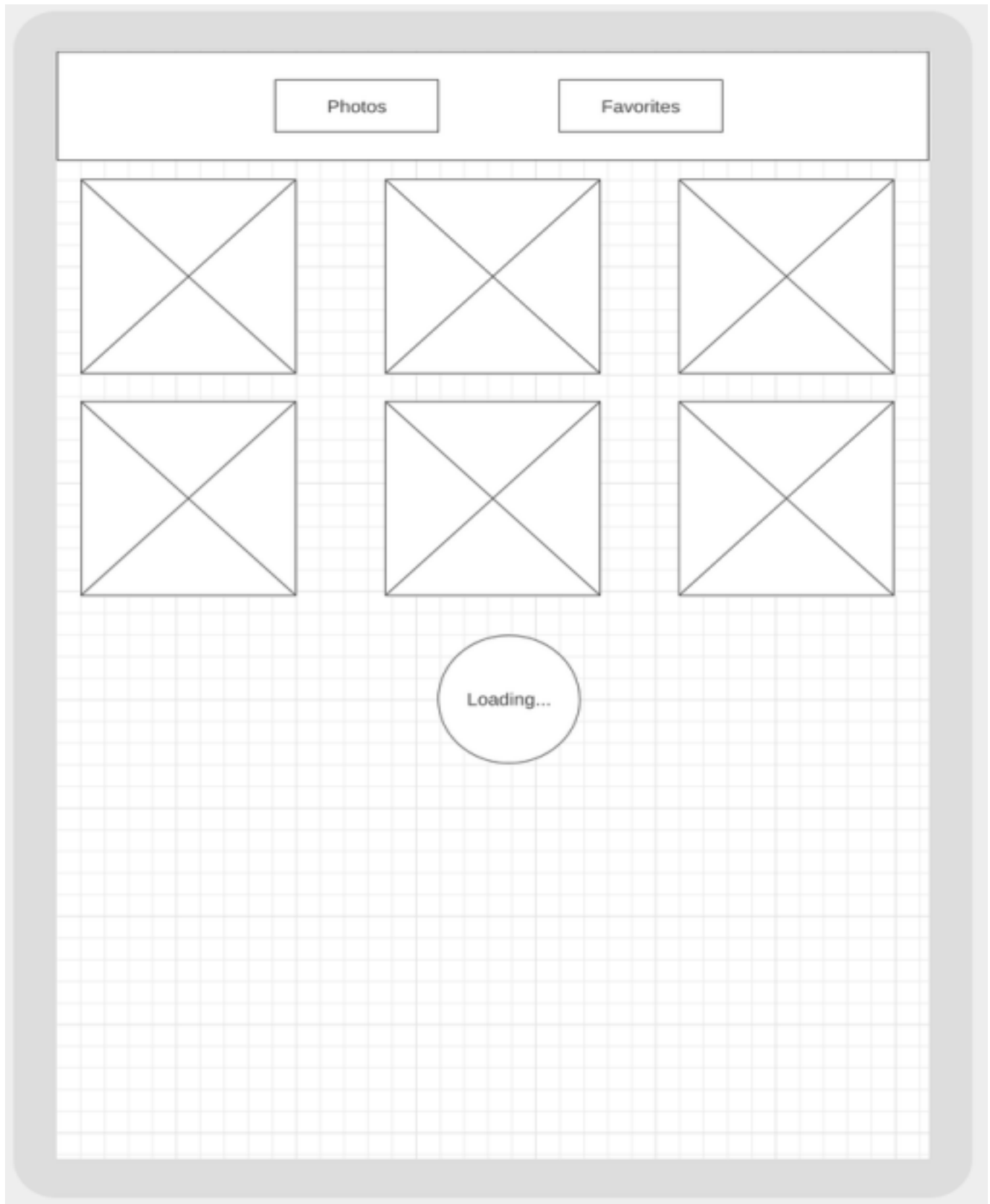
Task description

Create a photo library that includes an infinite random photostream, with the ability to save photos to your “Favorites” library.

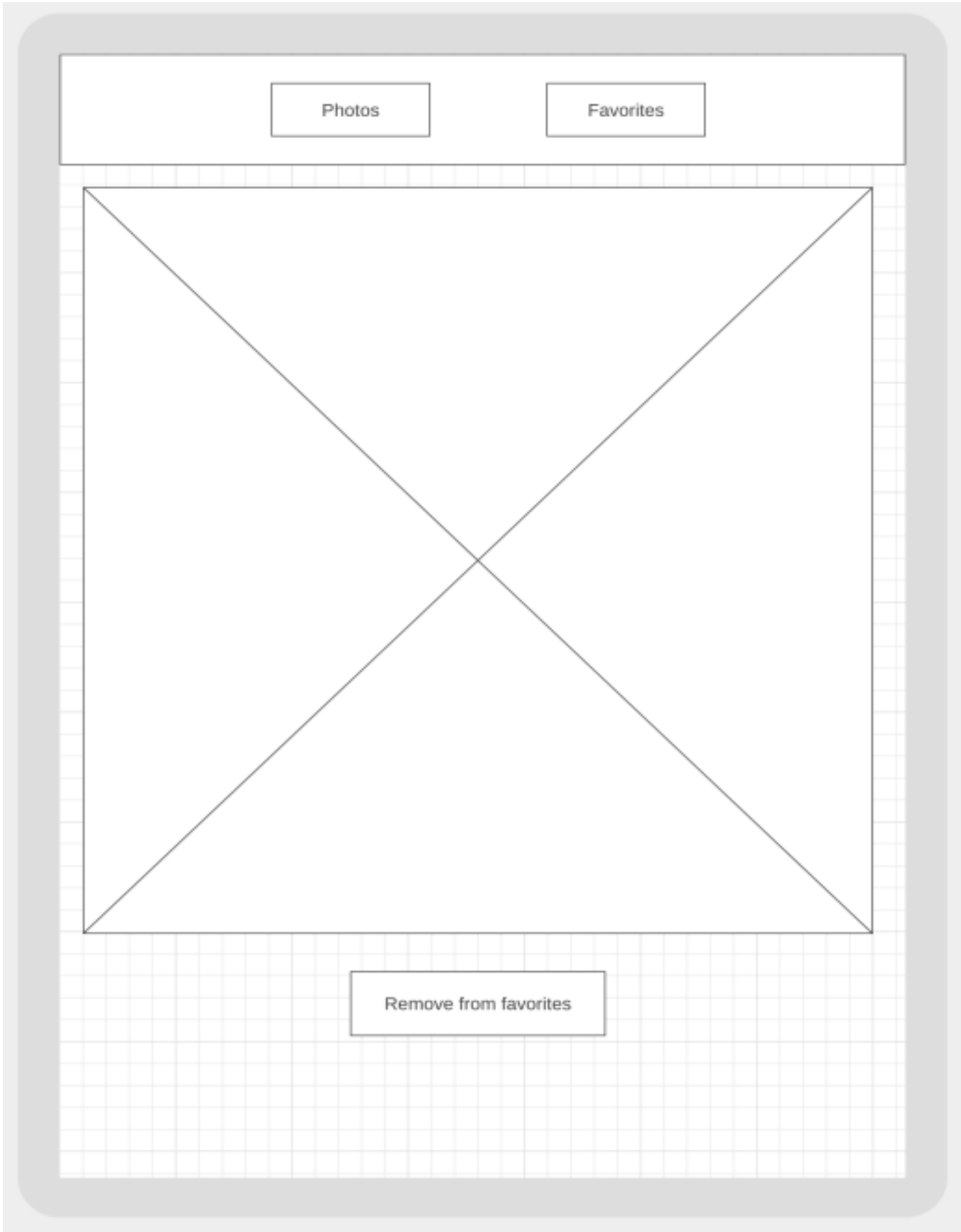
- Design
 - Below are wireframes, which give a general view of what the pages should look like
 - The theme (e.g colors, fonts) you can choose by yourself
- Header
 - Consists of 2 buttons and allows you to switch between your “Favorites” library and a random photostream.
 - An active view must be highlighted.
- “Photos” screen has an infinite scrollable list of photos
 - Located at / path.
 - Clicking a photo adds it to Favorites.
 - When scrolling, new photos should be loaded. Loader icon should be displayed.
 - Use <https://picsum.photos/200/300> to get random images (or any other resource).
 - Emulate real-world API, when getting photos. Loading new photos should have a random delay of 200-300ms.
- “Favorites” screen
 - Located at /favorites path.
 - Contains a list of favorite photos (no need for infinite scrolling here, just list of all photos).
 - Clicking on a photo opens a single photo page.
 - Favorites list should persist after a page refresh.
- Single photo page
 - Located at /photos/:id path.
 - Shows just a single full-screen photo, instead of a grid.
 - Should contain the “Remove from favorites” button.
 - The header remains the same on this page.

Here are mockups of what the pages should look like:

List view



Detail view (single photo page)



General requirements

1. Use Angular Router module
2. Use the latest Angular, and SCSS instead of CSS
3. Use Angular Material components
4. Implement the infinitive scroll on your own. Do not use libraries.
5. Don't use any backend server for retaining state
6. Add unit tests
7. Think carefully about how to structure your code. Make separate reusable components, modules, etc Test your code

For generation random images see:

<https://alternativeto.net/software/unsplash-it/about/>