

# Front-End Assignment

---

This is a simple front-end code challenge created to serve two purposes:

1. Show off your skills,
2. Give us a better understanding of your skills.

## The Task

---

The programming challenge has 3 parts:

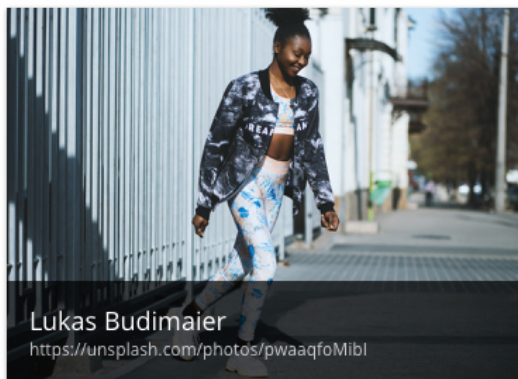
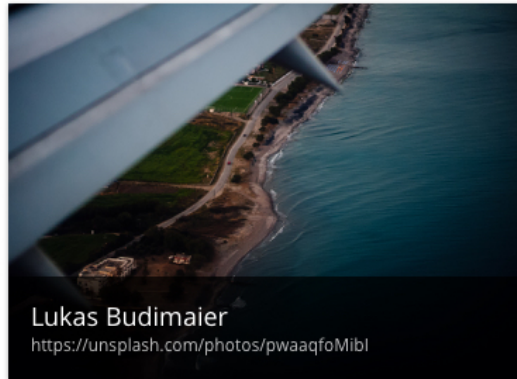
- Part A: HTML + CSS
- Part B: Javascript
- Part C: Infinite Loading (optional)

## PART A · HTML + CSS

### Photo Fetcher

☐ Make photos grayscale

Fetch New Photos



Layout max width: 750px

Gap between tiles: 16px

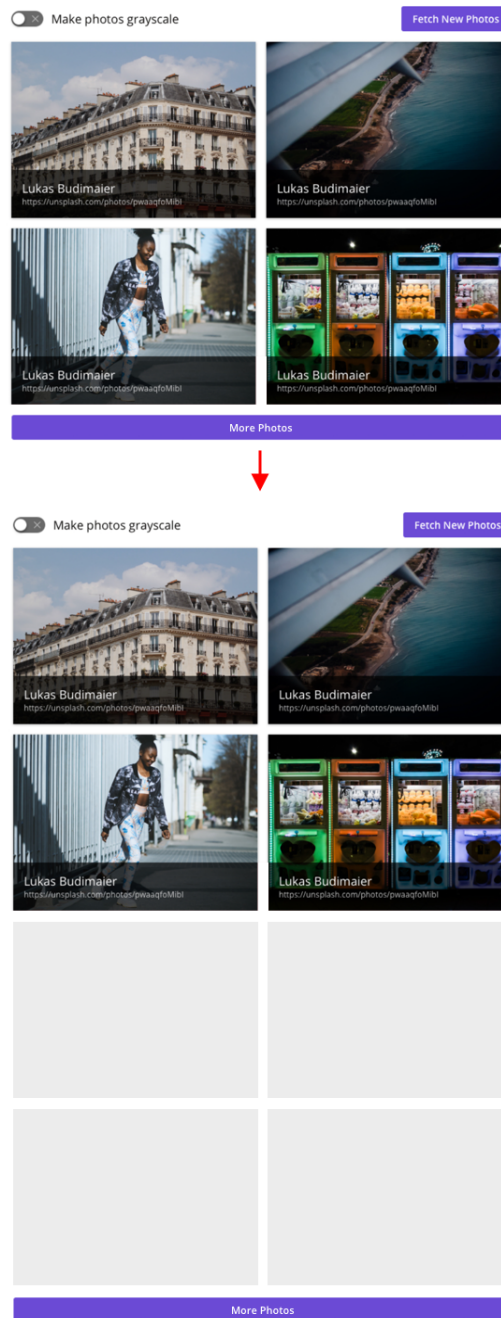
- Create a layout based on the given screen above.
- Style the layout consistently and beautifully, adhering as closely to the design as possible.
- NOTE: You are free to use any preprocessor you wish, or write raw CSS. *Bonus points for an explanation of your choice.*

## PART B: Javascript

- Consume this public API @ <https://picsum.photos/> to retrieve a list of photos.
- Display random 4 photos in the HTML + CSS layout created in step A, sticking to the layout, placing data where it fits.
- Toggle switch - while enabled, turns all the visible photos to be grayscale.
- Button “Fetch New Photos” will display new set of random 4 photos.
- NOTE: You need to do it in React.JS.

## Part C: Infinite Loading

- Add button to load 4 more photos
- Display skeleton loading indicator while waiting on images
- Button Fetch New Photos will reduce the number of photos back to 4 photos
- Example of loading behavior:



## How do we evaluate

---

- Clean and readable code.
- Pixel perfection.
- Componential thinking.