

# goTenna Full Stack Developer Assessment

## Introduction

Welcome! As a full stack developer at goTenna you will be tasked to develop, maintain and test the various applications that support our products. You must be able to write clean, maintainable code that can scale with the changing requirements and technologies of the given platform. And most importantly, you must share a passion for what you do and express it through your work.

Our products are used in many different arenas, including casual, everyday situations like hiking, to applications like emergency response and fire fighting. As such, we are expected to be able to create supporting software that is secure, robust, and easy-to-use. On to the assignment!

## Photo Viewer Application

Given [this](https://pastebin.com/BmA8B0tY) (<https://pastebin.com/BmA8B0tY>) csv file containing a list of image URLs, create a local photo gallery web application. This should be comprised of a front end gallery and a back end API. Utilize whichever software and frameworks you feel would best demonstrate an efficient approach to the problem, and include clear instructions on how to build your project for us to test.

Each URL is of the following format:

`https://picsum.photos/id/{id}/{width}/{height}`  
(e.g. <https://picsum.photos/id/1055/200/300>)

### Front End

- Images should be viewable in a grid-like responsive layout
- Gallery should be paginated in the method you best see fit (pages, infinite scrolling, etc.)
- Gallery should include an option for filtering images by dimension
- Gallery should include a “toggle grayscale” option (appending ?grayscale to any URL will return a grayscale version of that image)

### Back End

- Build a lightweight backend for parsing the .csv file and returning the URLs to the front end
- A RESTful API with an endpoint, or set of endpoints, that the frontend hits to retrieve images. Endpoint must support:
  - retrieving images/filtering images (paginated)

- toggling grayscale
- Bonus points for additional features such as request caching, image manipulation, error handling, and user authorization.

*Feel free to add any visual or back end flair that will show off your skills as a developer, as long as it doesn't slow down or over-complicate the application.*

## Submitting

Submit your completed project to your GitHub account and include a README.md file explaining how to build your project, and any quirks or other considerations you want us to be aware of. Send us a link to the project (make sure it's public!) and we'll take a look!