

Interactive Generative Art Gallery

The Broad Vision:

My vision for my final project is a gallery website of several interactive generative systems. These systems will be accompanied with a control panel from which the user can manipulate their properties in real time. The goal of this project is to explore and experiment with generative systems as a means of producing artwork. I hope to provide systems which are flexible enough to explore the major focuses of graphic artwork. I want users to be able to easily customize a system to their liking and be able to save stills directly from the program.

The Gallery:

Part of my project will be focused on the curation and collection of generative artwork using the systems in my project. I will use the systems to generate a sizable collection of work and I will pick the best and most creative outcomes to display. This portion of the site will serve as a photo gallery and will showcase the possibilities of my generative systems.

Alongside each photo or collection of photos, I will offer a brief description of the systems and the settings I used to obtain them. I want to do this to give a visitor a way to get started in their exploration. I am currently considering the commentary to be toggle-able so as not to spoil any surprises and to let the art speak for itself.

The Systems:

At the moment, I have 5 generative systems I hope to deliver:

1. Flowfield particle tracer (Prototype)
2. Cellular Subdivision
3. Circle Packing
4. L-system plant generator
5. Diffusion Limited Aggregation

For each of these systems, I will dedicate a page to it. Each page will be composed primarily of two elements: the canvas and the control panel. The canvas serves the simple task of displaying the output of the generative system. The control panel houses a series of sliders, input fields and buttons which allow the user to directly interface with the variables that control various aspects of the system.

Challenges and Methods

To achieve my vision, I will be using:

- HTML: Index page, a page for each system, DOM elements which allow the user to navigate the site and interact with the systems.
- CSS: Stylesheets and transitions for the elements on the page.
- Javascript: p5.js, jQuery, jQuery UI.

In designing my prototype, I wanted to try to focus on the interactive aspect. Up until now, I have been providing mathematical values for calculation as *const* values. A major hurdle I had to overcome was how to add a DOM input element which allowed the user to interact with the canvas. My current implementation uses JQuery's `.append()` function to essentially append a custom `<div>` to a container. The code would then read the data from the container and use it to calculate and generate the visuals. (Im beginning to wonder if this is the best implementation.) Another possible solution would be to write the various interactive elements directly into the HTML document instead of adding them at runtime.

Going forward, I feel like most of my struggle is going to stem from interacting with DOM elements. I'm going to do a lot of work on polishing my GUI system for the control panel so that I can easily recycle the code across the pages. I want the site itself to give off some sense of professionalism so I'm going to be sure to give it some polish. I want to use subtle transitions for interactive elements like buttons and sliders. The overall look of the page will be sleek and simple so as to place emphasis on the generative systems over anything else.

One final challenge I see ahead is pertaining to saving the canvas as a file. To my knowledge, p5.js includes a `saveCanvas()` function which allows the user to download the canvas as a jpg or png file. I hope to use this without any issues to fulfill the task.

Mockup Sketch and Diagram

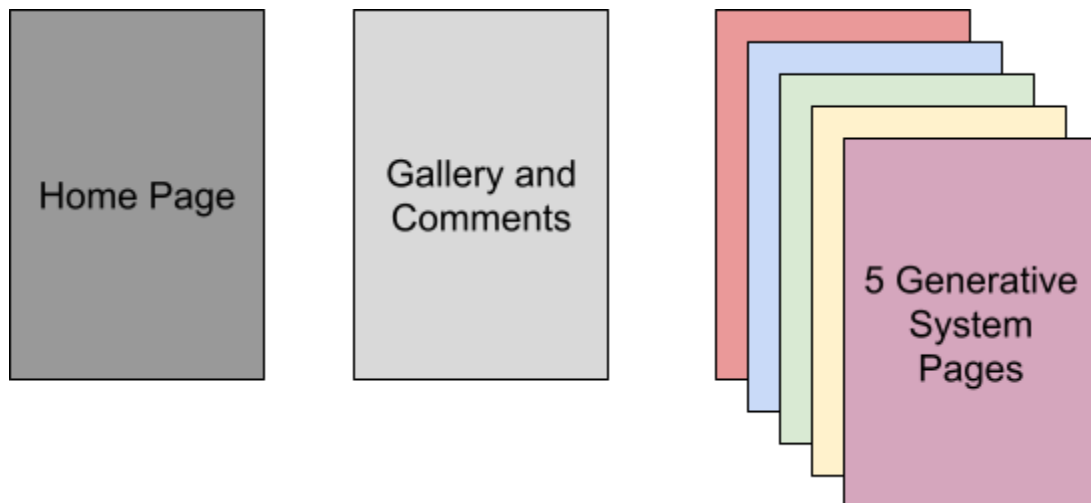
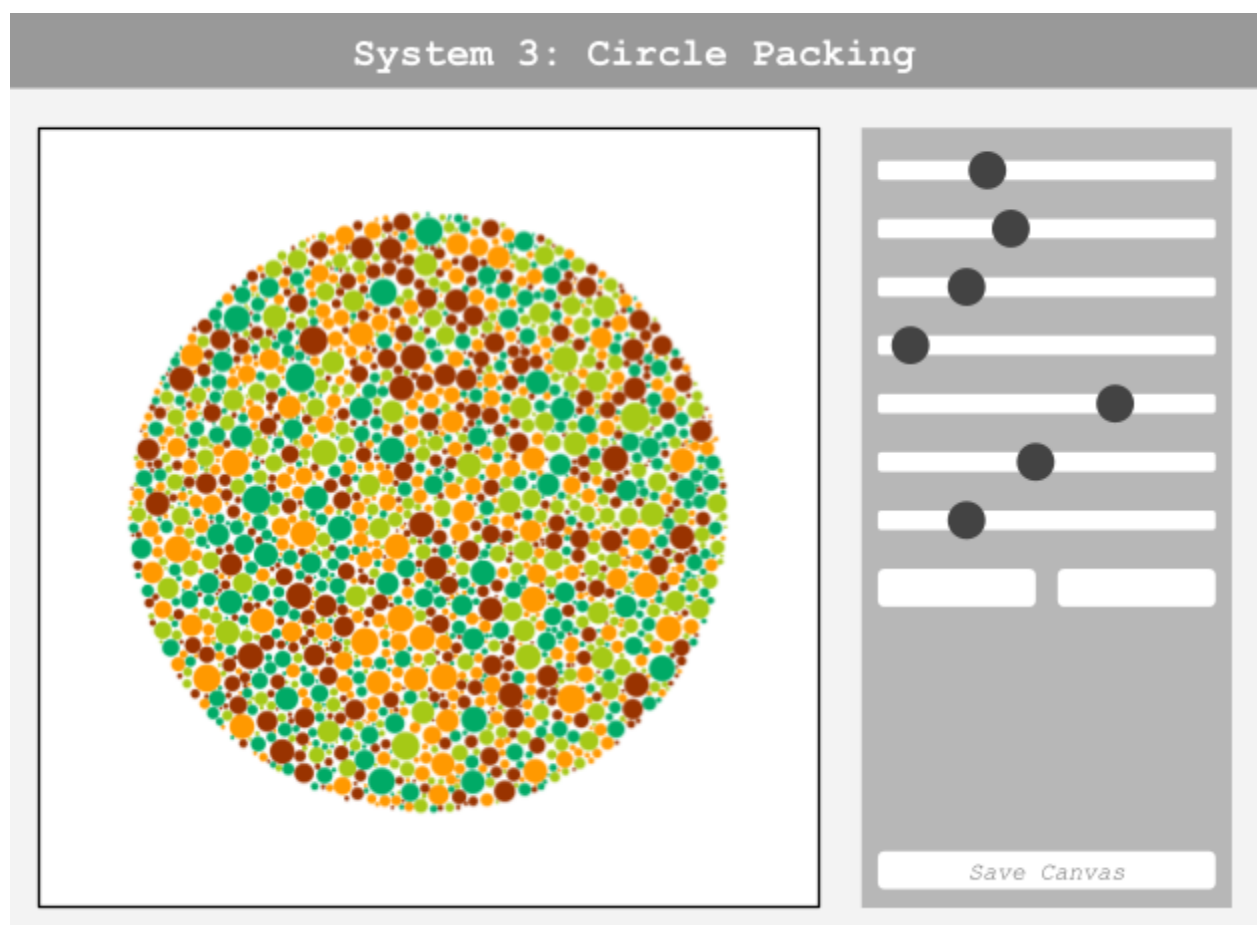


Diagram of all pages for the website.



Example generative system page layout.