

RADICALX



10/20/2023

LLM Interactive Webpages

- Jasmine Nguyen
- Kenedy Ducheine
- Jenny Nguyen



Outline

1. Intro
2. Objectives
3. Inspiration
4. Development
5. Implementation/Prototyping
6. Potential Modifications and Improvement
7. Sources and Works Cited



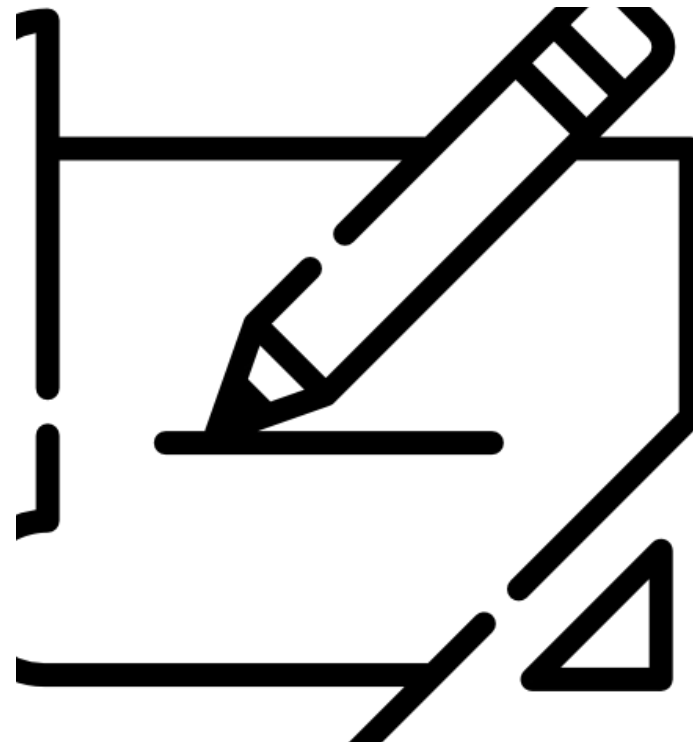
Introduction

Since a LLM can be prompted to output CSS or Javascript functions, we will test if they can receive coordinates and then output functions that can change the visual appearance of the canvas. For this project, we will be exploring this in a manually controlled environment to see how well can an LLM output Javascript code.



Purpose

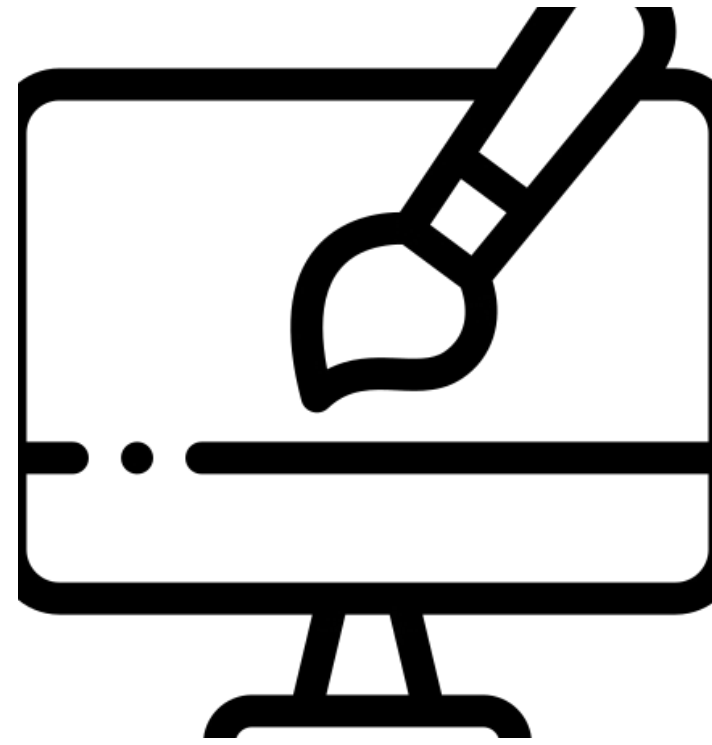
Design



Planning and Research

Design and develop an interactive canvas where a chat UI, powered by an LLM, interacts with a code editor.

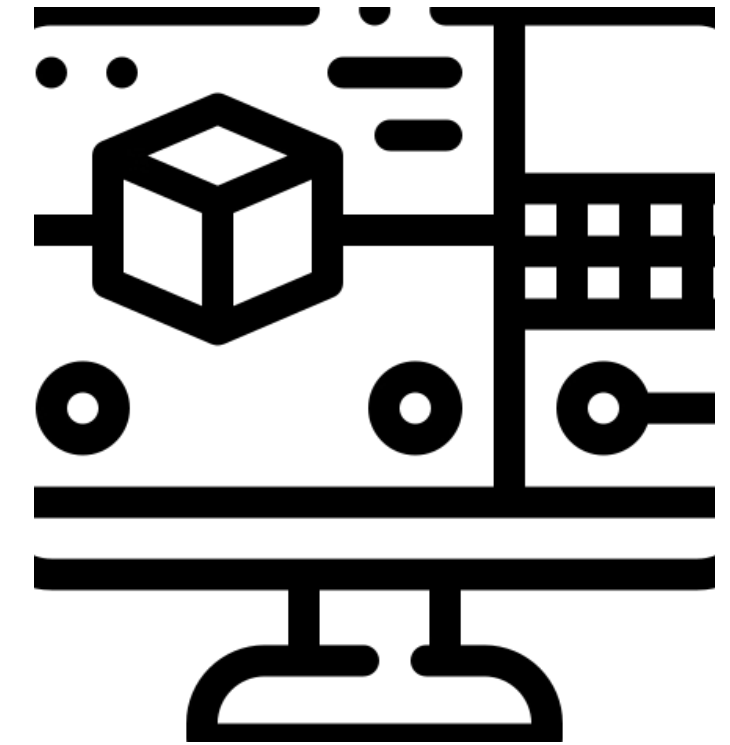
Interaction



Creativity and Learning

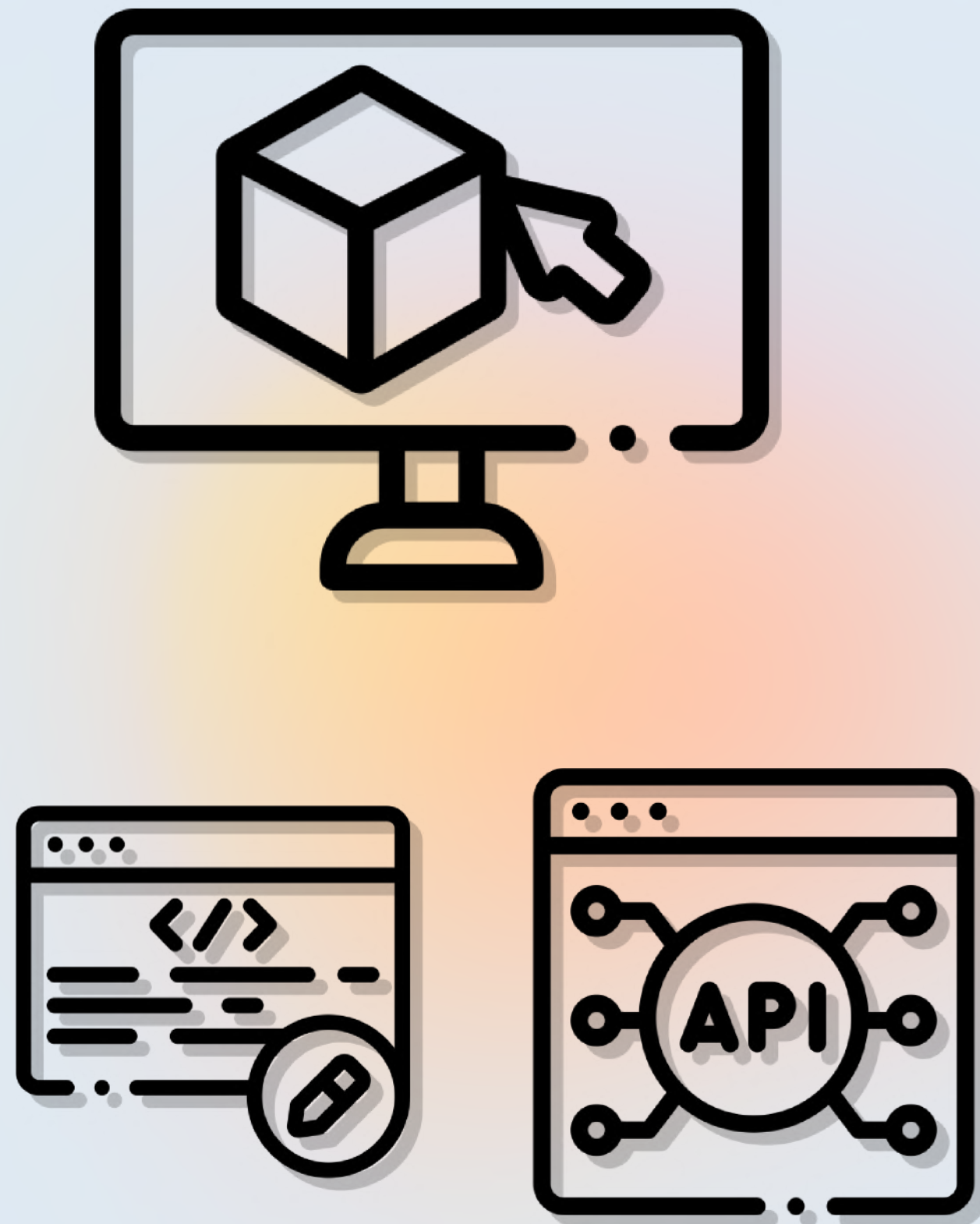
Dynamically represent the LLM's commands in a visually intuitive manner for the user.

Execution



Functionality and Testing

Incorporate Vertex AI with web frameworks

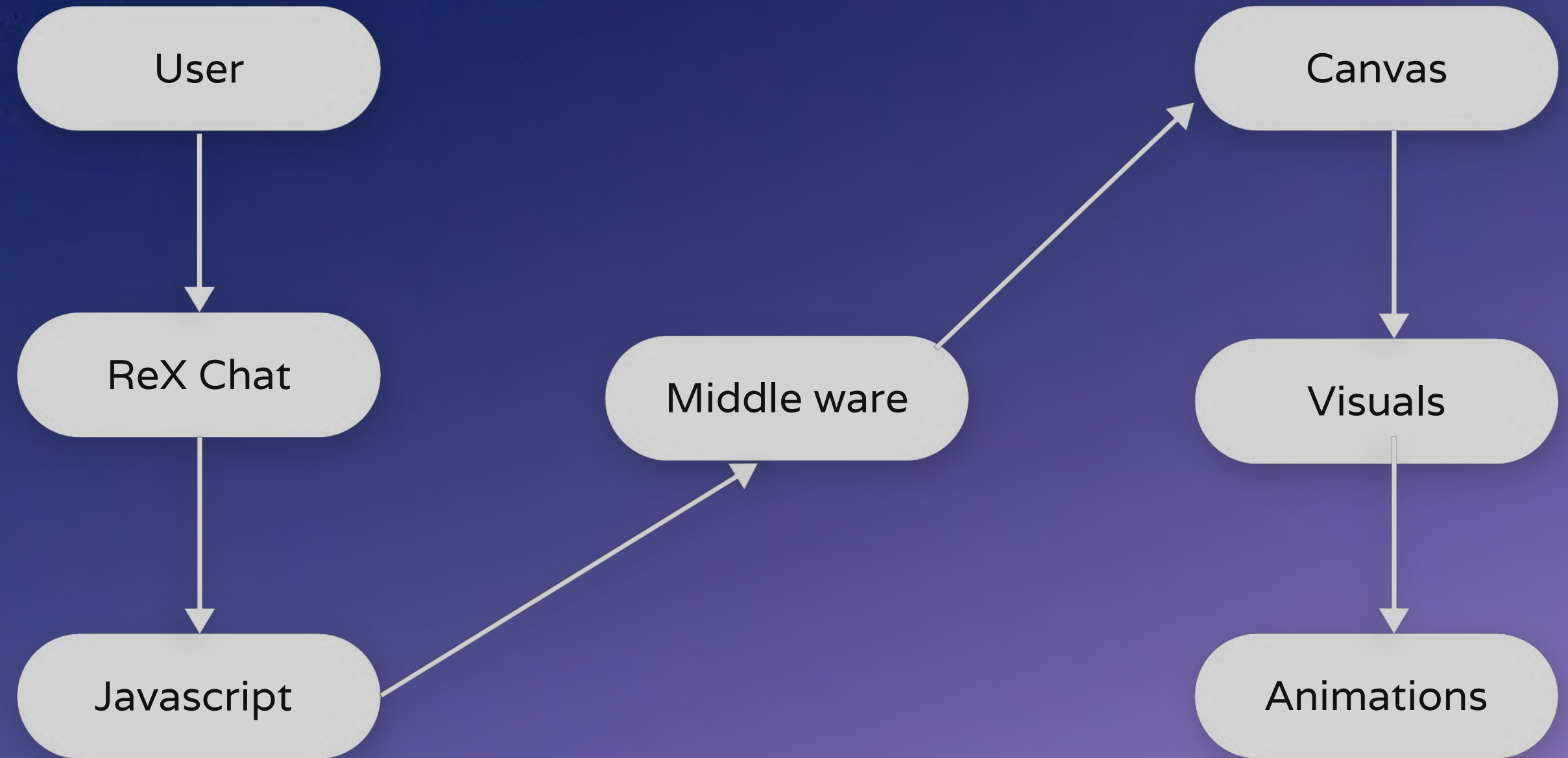


Objectives

- Our goal is to successfully hook up an LLM such as Vertex AI to a playground where users can get real-time JavaScript or HTML outputs onto a canvas in a web app.
- We want the user to ask questions and the chatbot will generate related animations, vectors, or illustrations on a separate canvas on the same webpage to help them learn.
- For example, if a user wants to see the circumference of a circle or perimeter of a shape, the bot will generate a visual illustrating on how to answer that question.
- Allow users to interact and use the canvas to play learning games and learn how to code them.

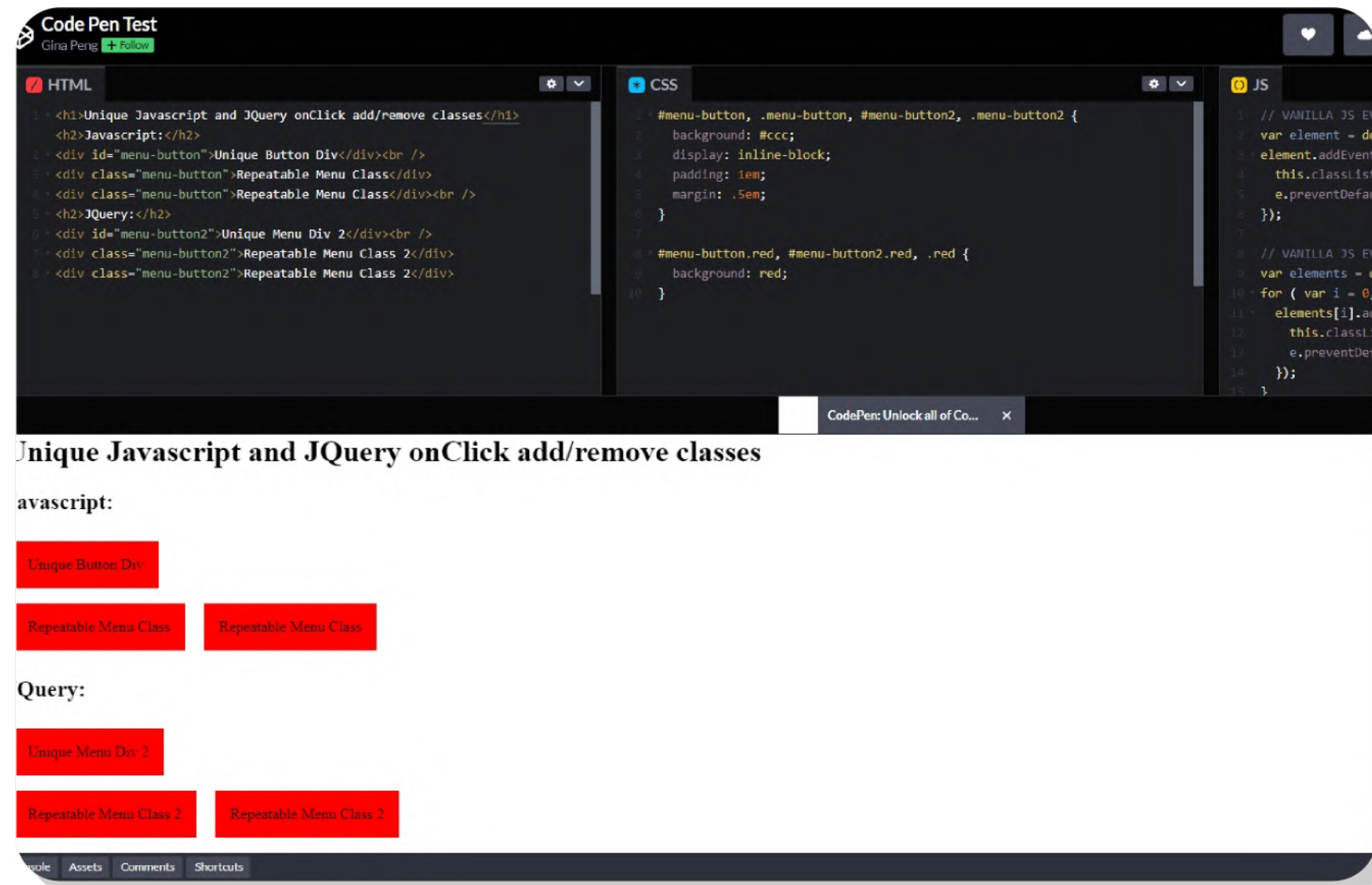
APPLICATION STRUCTURE

We will be exploring the middleware that can bridge the front end with Gen AI.



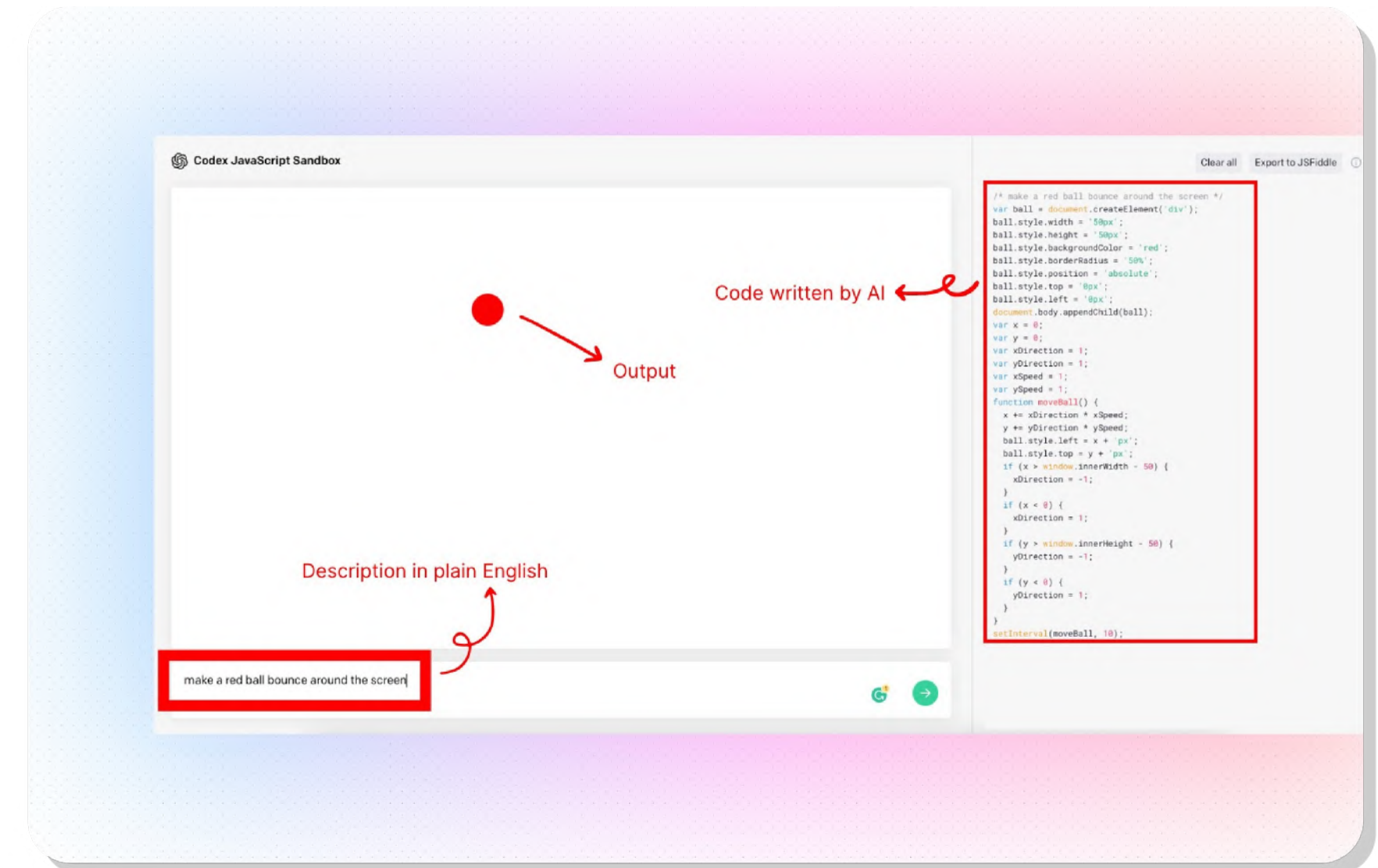
INSPIRATION, REFERENCES, AND OTHER SIMILIAR PRODUCTS

Codepen.io



- Social development environment for front-end developers
- Learning tool that allows you to catch errors early in an interactive environment
- Shows off the code of each file as well as the end result on the same screen

OpenAI Codex JS Sandbox

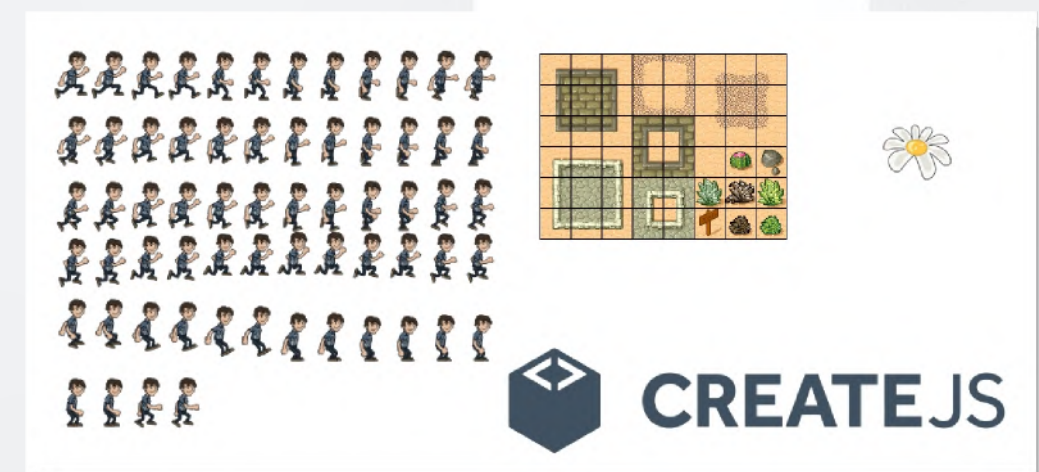
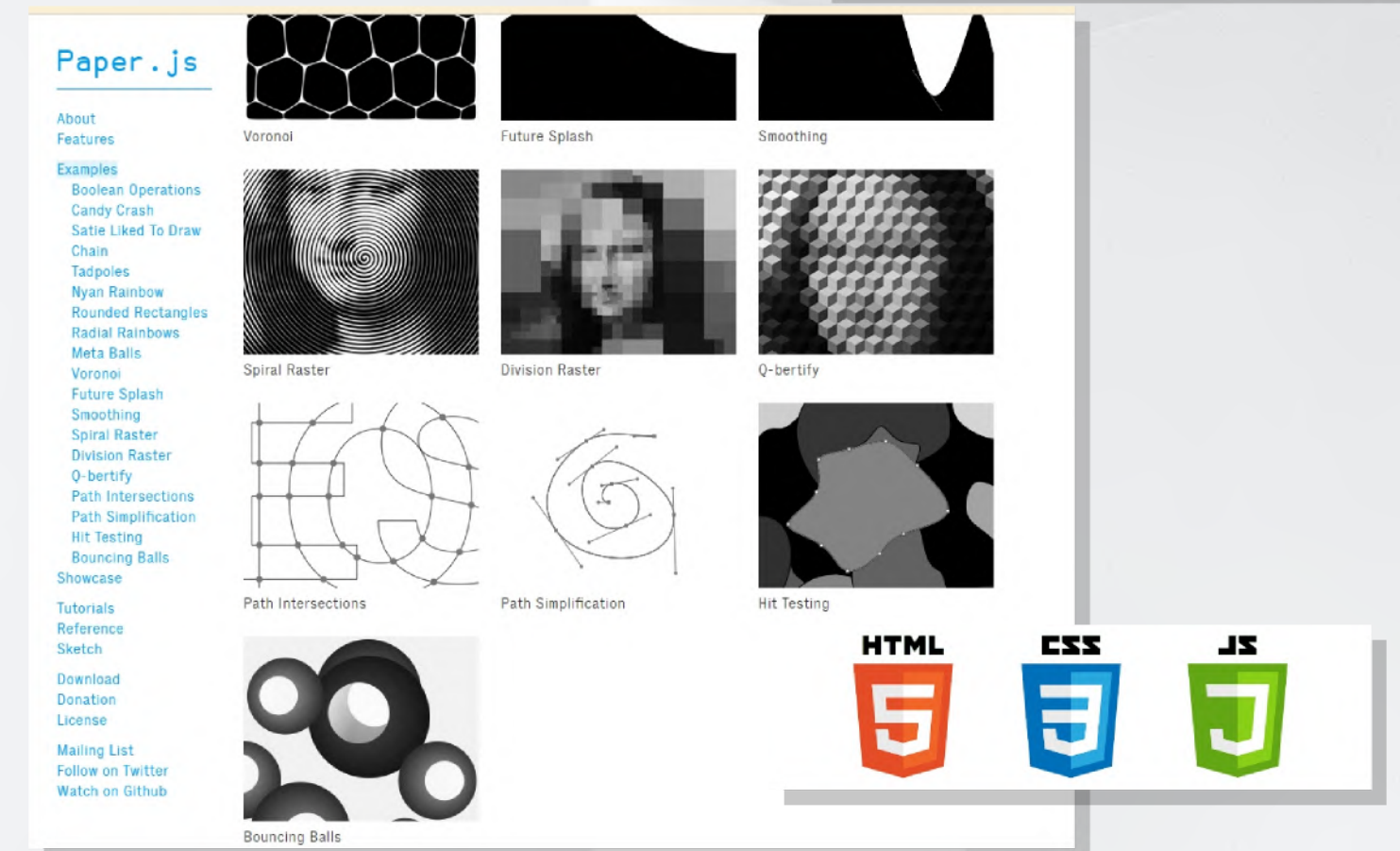
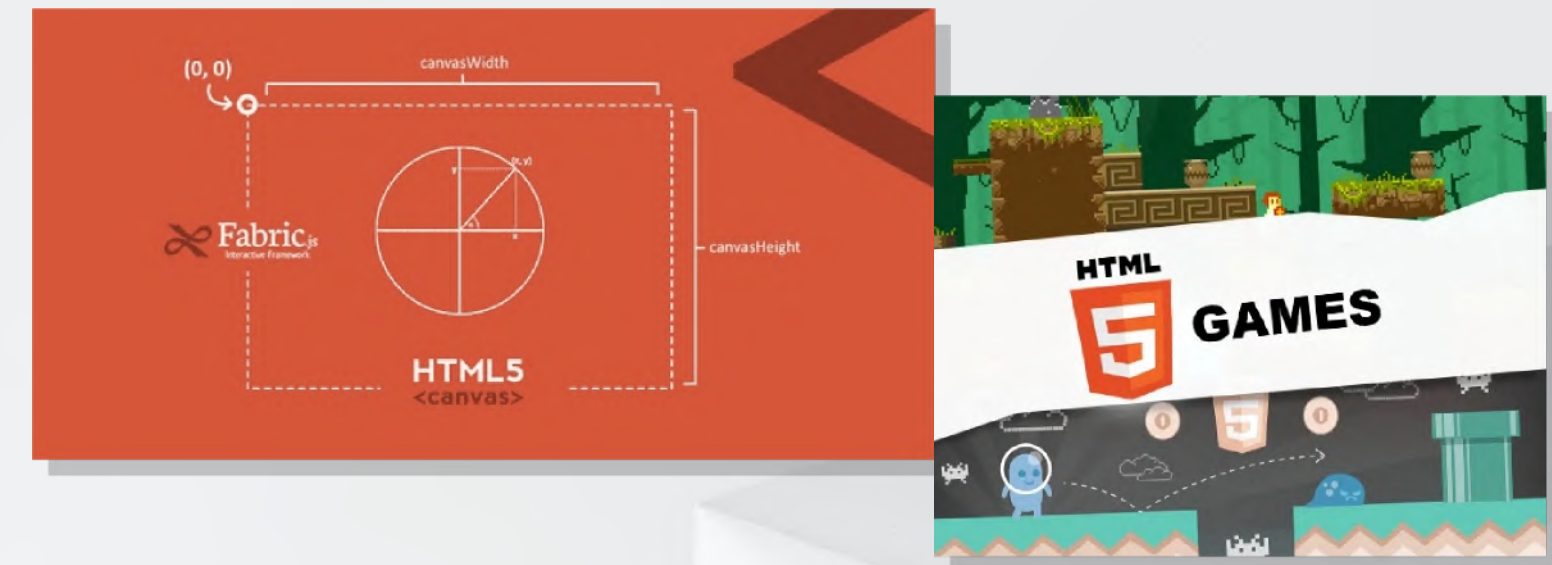


- Was initially introduced as a free limited beta in 2021.
- Can interpret simple commands in natural language and execute them on the user's behalf
- Build a natural language interface to existing applications.
- Discontinued in March 2023 to push for GPT but their playground has a usage limit

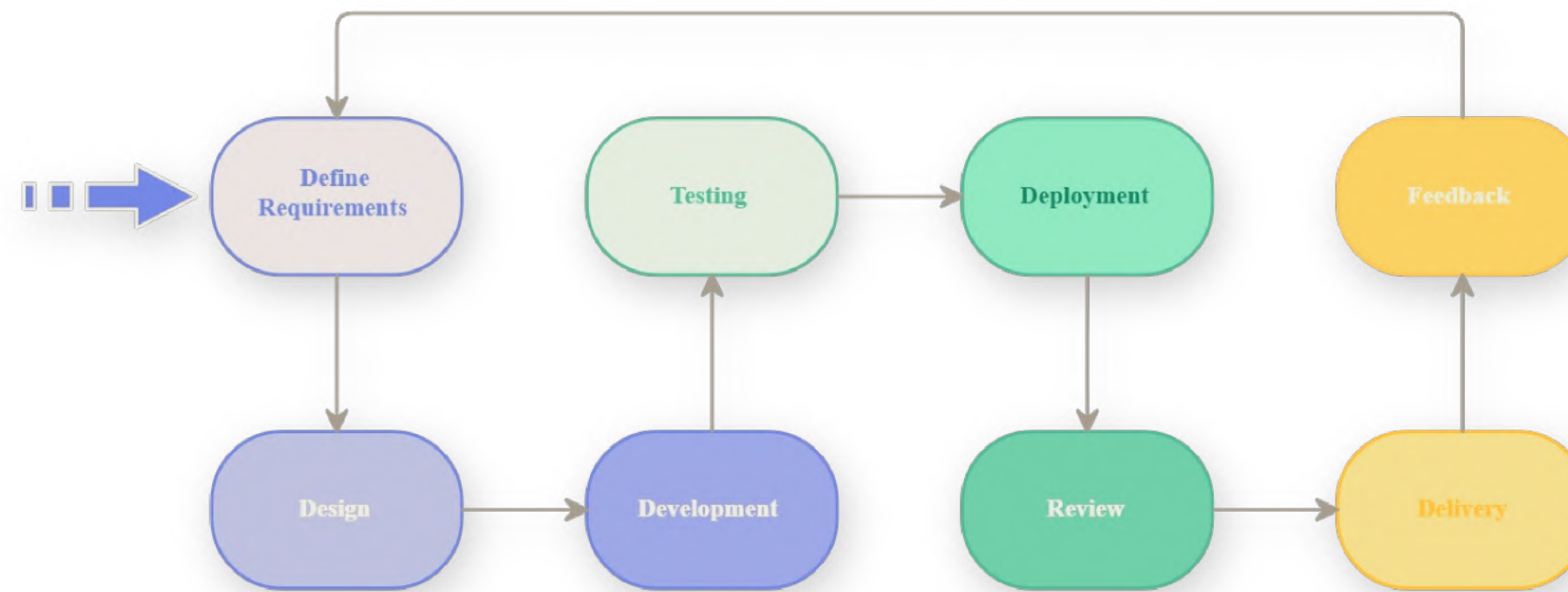
Possible Techniques

Need to create a bridge between the LLM's high-level outputs and the low-level canvas drawing commands without hardcoding every possible shape or interaction.

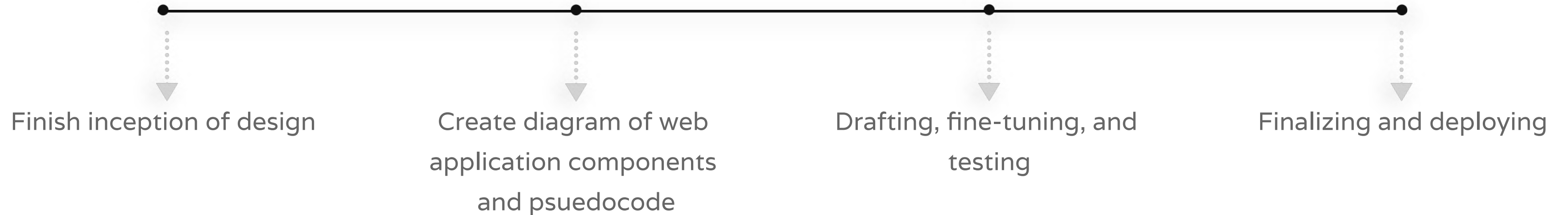
- Creating a middleware layer.
 - Retrieve model response from user input.
 - Have a middleware layer map the commands to specific functions that handle drawing with HTML5 canvas libraries such as fabric.js or paper.js.
 - Build a dynamic rendering engine for high-level commands



PROJECT PLAN

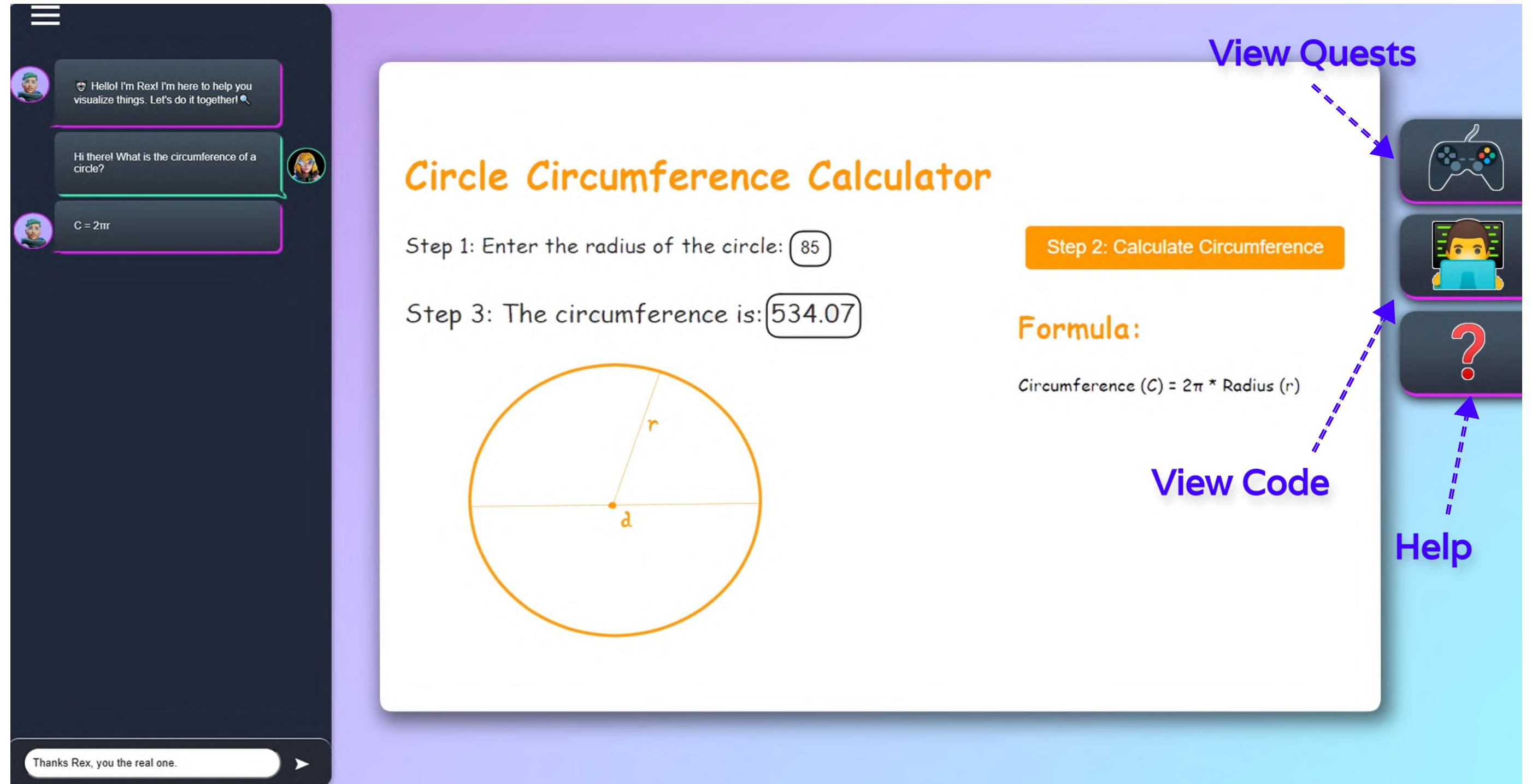


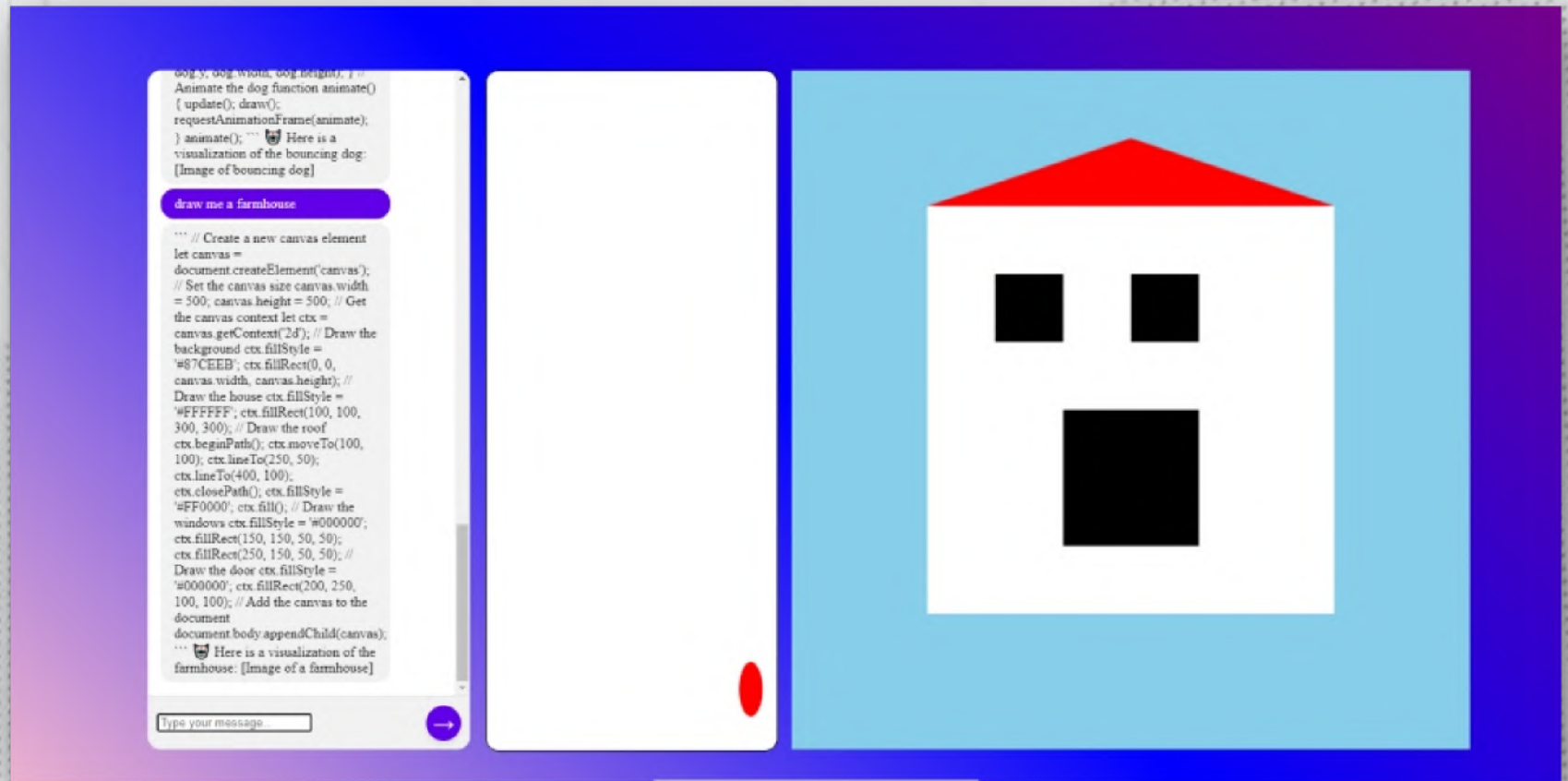
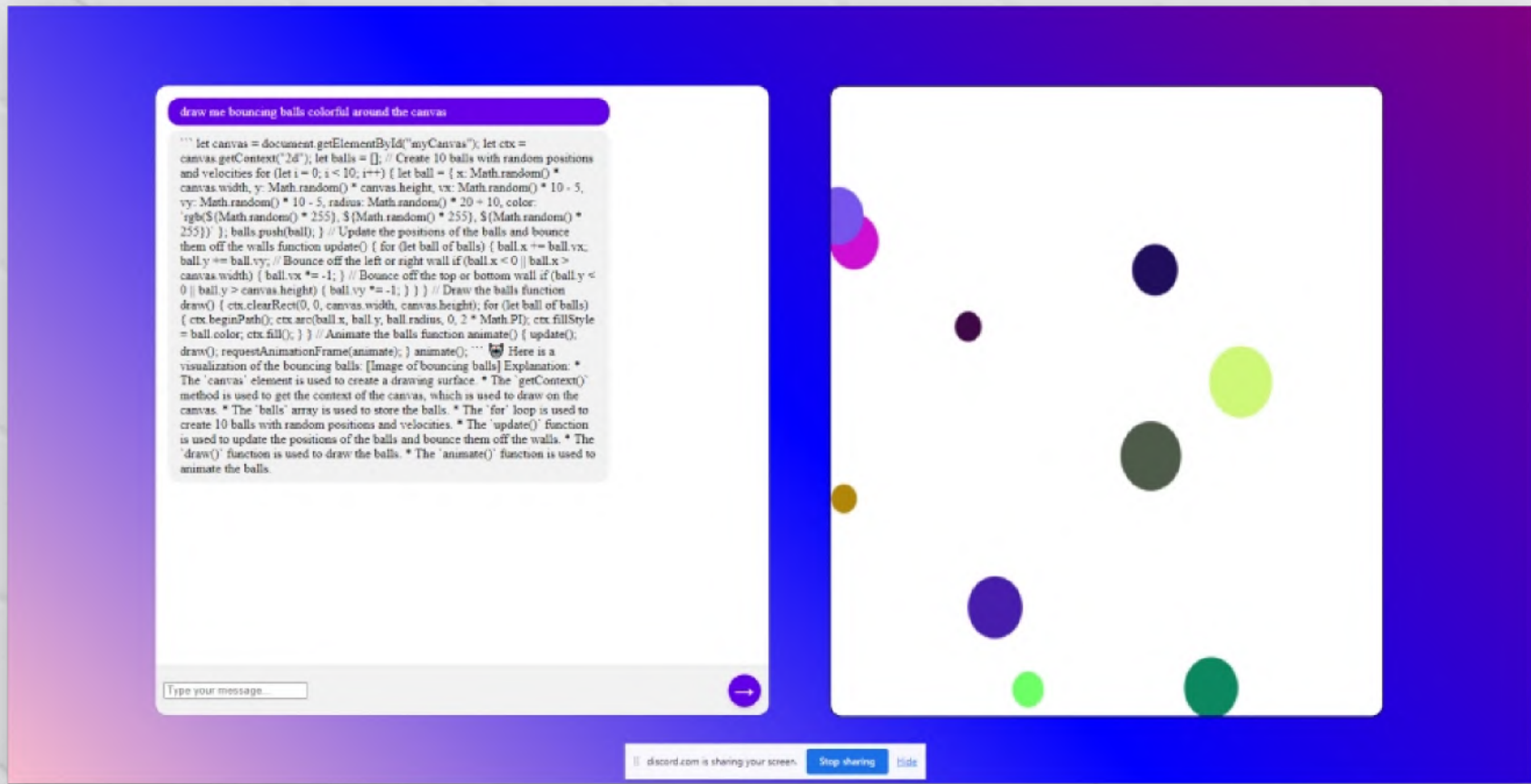
- Agile Software Development Process -



Rough Mock-Up

General goal of how we want to design and improve it. Can be difficult to implement.





Prototyping 1.1

Basic demo to test the model. Currently can output abstract images with simple shapes. Has limited capability with HTML canvas.

Basic Video Demo

```

orange, ctx.lineWidth, ctx.strokeStyle = "black", ctx.stroke(), // Find the
circumference of the circle let radius = 100; let circumference = 2 *
Math.PI * radius; // Draw a line to show the circumference
ctx.beginPath(); ctx.moveTo(250, 250); ctx.lineTo(250 + radius *
Math.cos(0), 250 + radius * Math.sin(0)); ctx.stroke(); // Draw a label
for the circumference ctx.font = "16px Arial"; ctx.fillStyle = "black";
ctx.fillText("Circumference: " + circumference, 250, 250 + radius + 20);
''' 🐼 I have made the circle bigger and drawn a line to show the
circumference. I have also added a label to show the circumference.

```

thanks!!!! can you find the perimeter of a square if each side is 7? can you illustrate that on my canvas?

```

    ... let ctx = document.querySelector('canvas').getContext('2d'); // Draw
    a square ctx.beginPath(); ctx.rect(100, 100, 7, 7); ctx.stroke(); // Find
    the perimeter of the square let sideLength = 7; let perimeter = 4 *
    sideLength; // Draw a label for the perimeter ctx.font = "16px Arial";
    ctx.fillStyle = "black"; ctx.fillText("Perimeter: " + perimeter, 100, 100 + 7
    + 20); ...

```

The perimeter of a square is the distance around the square. The perimeter of a square can be calculated using the formula: $P = 4 * s$ where P is the perimeter and s is the length of one side of the square. In this example, the length of one side of the square is 7, so the perimeter is: $P = 4 * 7 = 28$ I have drawn a square on the canvas and labeled the perimeter.

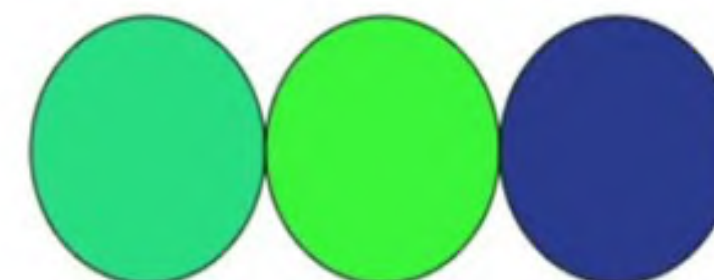
can you draw me 3 randomly colored circles with a black border?

```

... let ctx = document.querySelector('canvas').getContext("2d"); //
Generate 3 random colors let colors = []; for (let i = 0; i < 3; i++) {
  colors.push('rgb(' + (Math.floor(Math.random() * 256)),
    (Math.floor(Math.random() * 256)), (Math.floor(Math.random() *
    256))) + ');'); } // Draw 3 circles for (let i = 0; i < 3; i++) { ctx.beginPath();
  ctx.arc(250 + i * 100, 250, 50, 0, 2 * Math.PI); ctx.fillStyle = colors[i];
  ctx.fill(); ctx.strokeStyle = "black"; ctx.stroke(); } ... 🐱 Here are 3
randomly colored circles with a black border. [Image of 3 randomly
colored circles with a black border] 🚫 🐱

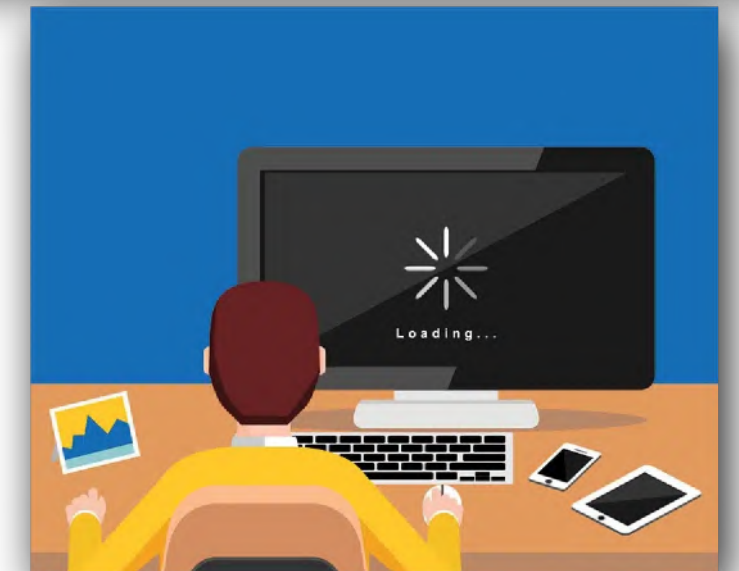
```

can you rd

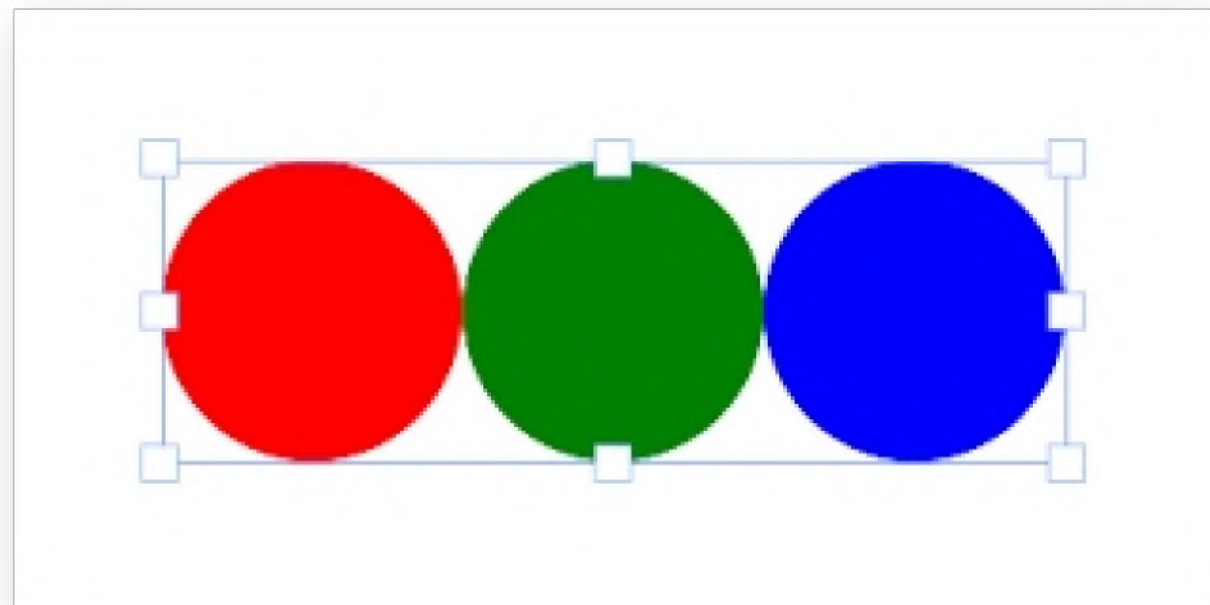


Issues and Roadblocks

1. Slow model and image response
2. Images do not display on the canvas at times
3. Limited vectors
4. Cannot handle complex prompts
5. Poor readability with the code and bot conversation



Fabric .JS

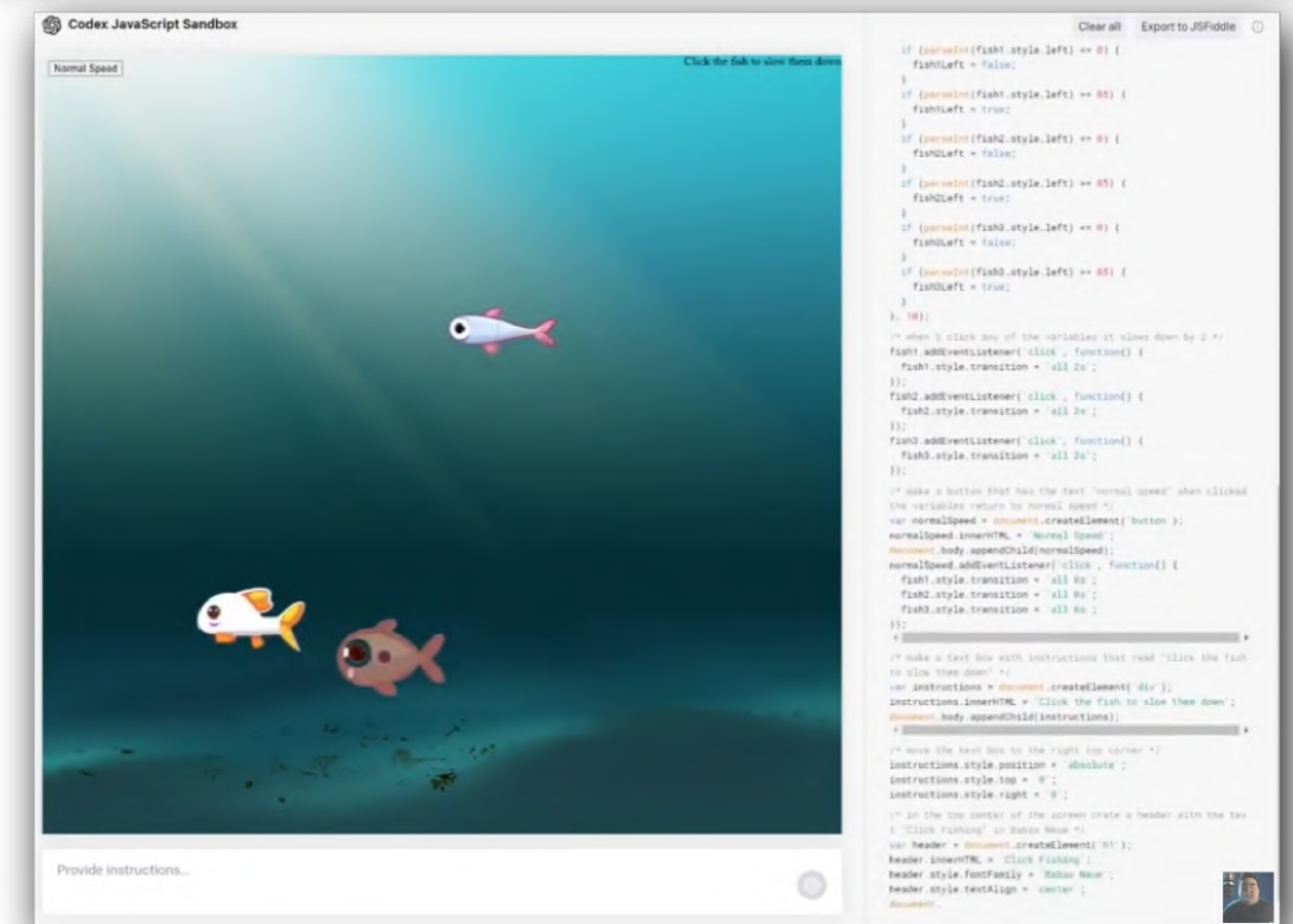
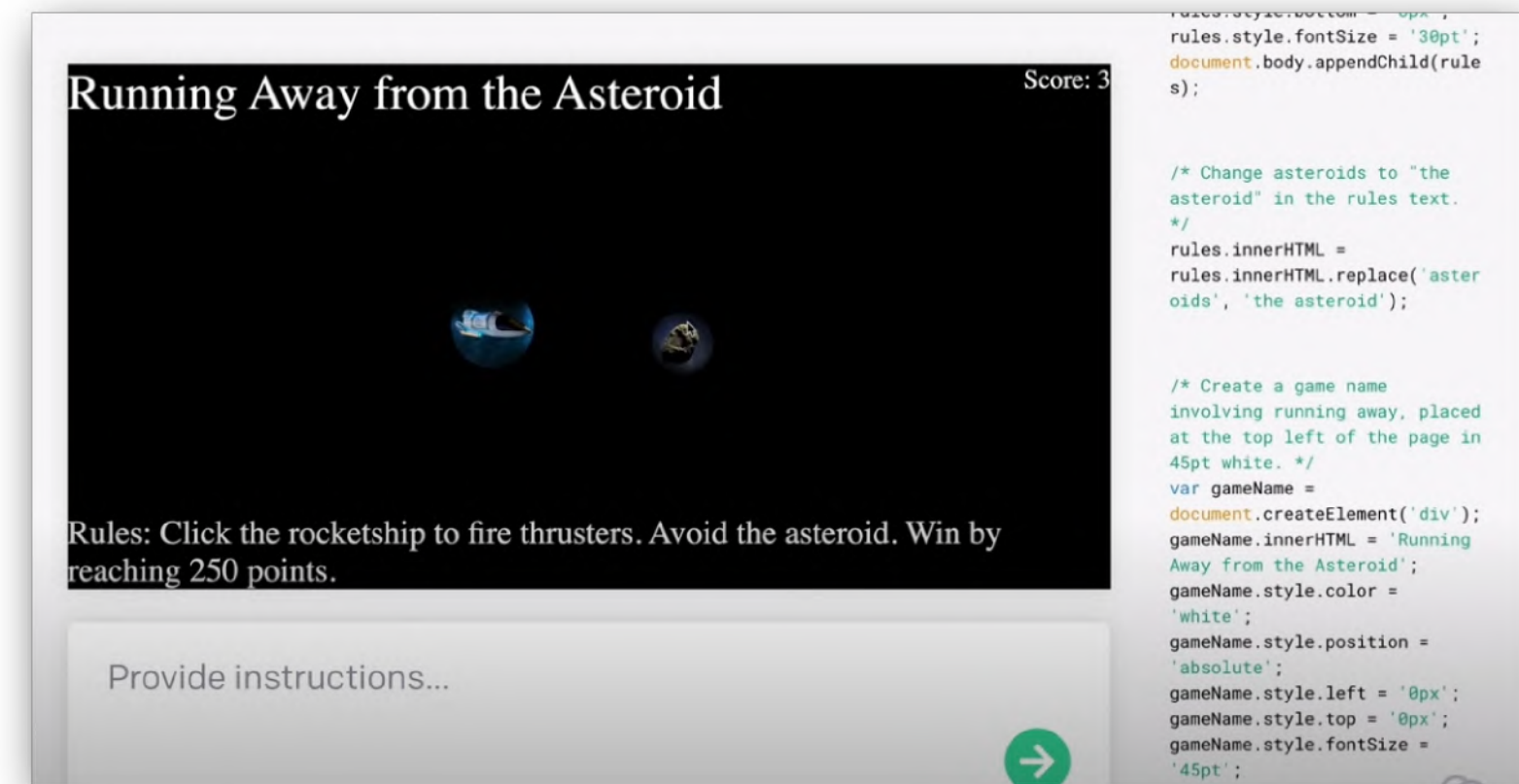


Next Steps

- Switch to Node.js to use fabric.js, a powerful and simple Javascript HTML 5 canvas library
- Benefits of fabric.js:
 - Can populate complex shapes and images to the canvas
 - Users maybe be able to add text and manipulate the shapes themselves after the LLM outputs them.
 - Built-in animation support, ability for drawing and erasing

Potential Modifications and Plans

1. Creating more detailed illustrations with the LLM
2. Creating non-random animations
3. Allow the model to fetch and display royalty free images from a URL
4. Experiment with generating 3D models
5. Further fine-tuning the model
6. Adjusting the UI and the chatbot's text format for readability
7. Allow users to interact with the canvas directly by creating and playing HTML5 Games



Sources

<https://sourceforge.net/software/product/OpenAI-Codex/alternatives>

<https://codepen.io/>

<https://openai.com/blog/openai-codex>

<https://asana.com/resources/agile-methodology>

<http://fabricjs.com/>

<http://paperjs.org/>



Pitch

Want to make a presentation like this one?

Start with a fully customizable template, create a beautiful deck in minutes, then easily share it with anyone.

Create a presentation (It's free)

