**My Assignment**

## Question 1:

- I'm extending the drawing app template because I found it interesting

**- Extensions Implemented**

1. **Arrow Tool**
   - **Complexity:** Moderate
   - **Implementation:** Uses trigonometry to calculate the arrowhead angles and position.
2. **Diamond Tool**
   - **Complexity:** Low to Moderate
   - **Implementation:** Calculates diamond vertices based on the center point and dimensions.
3. **Ellipse Tool**
   - **Complexity:** Low
   - **Implementation:** Uses the ellipse method in the canvas context.
4. **Eraser Tool**
   - **Complexity:** Low
   - **Implementation:** Functions similarly to the freehand tool but removes content.
5. **Spraycan Tool**
   - **Complexity:** Moderate
   - **Implementation:** Uses randomization to simulate spray patterns.
6. **Square Tool**
   - **Complexity:** Low
   - **Implementation:** Similar to the rectangle tool with equal sides.
7. **Text Tool**
   - **Complexity:** Low to Moderate
   - **Implementation:** Allows text input at mouse click location.
8. **Undo and Redo Functionalities**
   - **Complexity:** High
   - **Implementation:** Uses stacks to keep track of canvas states.

## Question 2:

- I looked up other drawing apps and implemented features that seemed useful

- For coding, I've written about 60% of the core functionalities and extensions, I've also started implementing basic undo/redo functionalities using stack data structures for managing canvas state transitions.

Next, I plan to refine existing tools to enhance functionality and user interaction, addressing any identified bugs or performance issues. Additionally, I'll integrate advanced features like more robust undo/redo capabilities with comprehensive history tracking as well as a node based ui. Testing across various browsers and devices will ensure compatibility and reliability, while thorough debugging will resolve any behavioral or responsiveness issues. Finally, I'll

document the app comprehensively, providing clear usage instructions and examples to help users maximize its capabilities.

## Question 3:

To organize my time for the rest of the project, I've created a Gantt chart that outlines the major tasks and their respective timelines over the next several weeks.

**- Weeks 13-14:**

**Node Based UI**: I will start with the Node Based UI because it's a foundational element. Other tools and features depend on the structure and functionality provided by the Node Based UI. Completing this first ensures a stable base for subsequent tasks.

**- Weeks 15-16:**

- **Image Tool**: After setting up the Node Based UI, I will move on to the Image Tool. This tool's functionality is enhanced by the Node Based UI, making it logical to follow up with it.
- **Laser Pointer**: Concurrently, I will begin working on the Laser Pointer tool. The overlap in weeks ensures efficient use of time and resources, as some aspects of these tools can be developed in parallel.

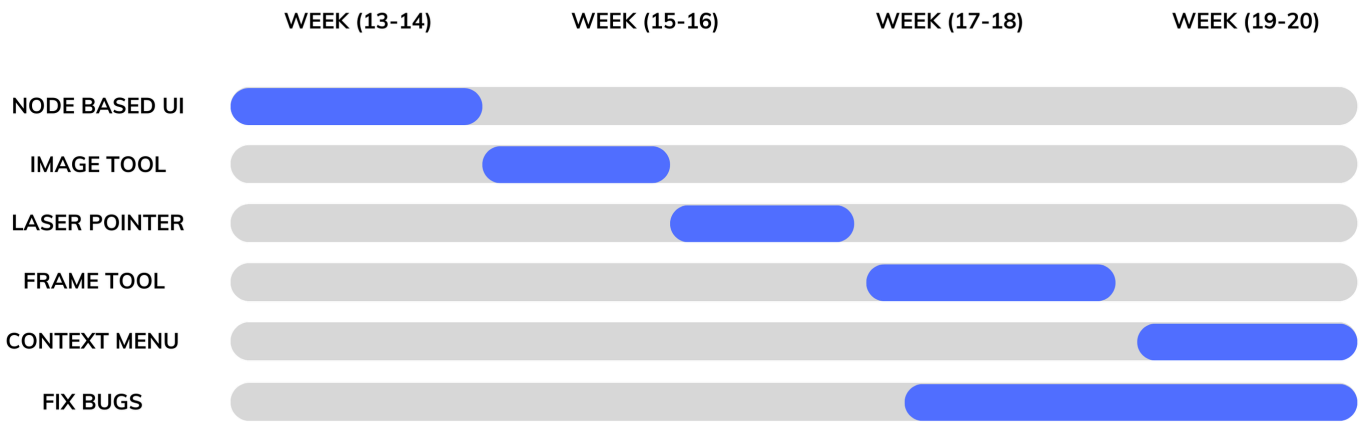
**- Weeks 17-18:**

- **Frame Tool**: I will start the development of the Frame Tool, using insights gained from previous tasks to ensure smooth integration.

**- Weeks 19-20:**

- **Context Menu**: With the foundational tools in place, I will focus on developing the Context Menu, which ties together various functionalities and enhances user interaction.
- **Fix Bugs**: Finally, I have allocated a significant portion of time to fixing bugs. This is crucial to ensure the stability and usability of the entire project before submission.

## GANTT CHART




## Question 4:

- p5js official documentation

- jquery

- lucide icons (for the icons)

## Question 7:

My project is a drawing app that stands out due to its unique features and innovative design. I've extended a standard drawing app template with new tools and functionalities, applying complex coding techniques to enhance user experience. Inspired by existing solutions, I adapted and evolved these ideas to create a more versatile and user-friendly application.