Shafi Rehman

Intro to Programming 2

Drawing App Extension

Part 1: I will extend on the drawing application. I will expand the current system functionality, create extensions that will augment the existing functionality, and add new features to the app. I chose this project in order to create a tool for product managers that need to create clear and simple user flows, block diagrams and other visualizations that can communicate ideas to their teams. I have created a plan to

System Expansion: These features will apply across all the drawing app tools

* Increasing color palette
* Rotation tool
* Selector for transparency of drawn lines/shapes
* Line Weight Selector

Current Tool Expansion: Existing drawing app functionality will be augmented with new options

* Dotted and/or dashed line (line tool)
* Paint splatter effect, which will create randomly irregular and randomly transparent splatter (within spray can tool)

New Functionality: Completely new features to add

* Text Input with text box and text sizing
* Stamp Tool
* Eraser tool
* Editable shapes
* Swimlane Template for flow diagrams
  + Motion: Highlighting specific shapes in a specific order to mimic “movement” through flow diagram

This is a large number of functions to increase, but I am confident that I can complete it on time because much of it is relatively less complex and builds on what we have learned already. The most complex will be the swim lane template as well as the motion functionality. I plan to use P5.js and P5.DOM, primarily, and any techniques I learn from external sources to build more complex pieces of the app (p5.shape, for example). Some of the challenges I expect to face are with the implementation of the templates and the way to portray motion in an ordered way.

**Part** **2**: My current design work has focused on investigating useful applications of the drawing app. I have talked to designers, and asked them to show me the features available in [Photoshop](https://www.photoshop.com/en), Adobe [Illustrator](https://www.adobe.com/products/illustrator.html), and [Invision](https://www.invisionapp.com/). I also spoke with product managers who create UML (Unified Modeling Language) diagrams, and also examined the functionality of [Gliffy](https://www.gliffy.com/examples/uml-diagrams).

One of my key findings was that flow diagram tools are static, and don’t do a good job of showing where in the flow someone is, and showing the movement to the next steps. This is particularly important because many people use these diagrams on video calls to explain their thoughts, and they want to clearly show how their flow diagram works while also keeping audience’s attention. This was the direct driver of the feature to show specific “motion” across drawn objects. I am not completely sure how I will implement this in code, but I have considered a few options, including changing the color of each line in a preselected order by allowing the tool to specify this order.

I have also created a code diagram, please see the attached image “SR Code Diagram” in the folder.

Finally, for current code progress, I have not currently begun the code implementation. I think I have enough time to learn how to execute each of the additional components, and I spent a good portion of the pre-midterm time working on design and brainstorming, rather than coding.

Part 3

The rest of my project plain is in the Gantt chart that is included in my attached submission, as I was unable to get a screenshot to insert here. My high-level plan was to complete research/brainstorming by the midterm, transition to development of features, and then test/document/debug. I allotted time in development based on my perceived complexity, so the most time was allocated to the New Functionality sections, rather than the System Expansion. I also have made sure to leave extra time as a buffer if there are unforeseen events that come up, so that I may submit on time with an app that functions as expected.

Part 4: Sources and Inspiration

I have spent much of this time working on deciding what to design, and so much of the sources for inspiration I have used are focused on this. I have listed all sources below, but I expect that these sources will continue to grow as my work continues, and this expanded list will be present in my final submission. I also read some blog posts and websites about coding a drawing app, so I could better understand other programmer points of view and incorporate them into my thinking.

* Student video from Week 4 (Evan Griffiths)
* InVision – transparency and text boxes
* Photoshop – splatter effect
* Motion diagram/UML block diagrams – Gliffy
* P5.shape
* <https://www.tutorialspoint.com/learn_javascript_and_html5_canvas_build_a_paint_drawing_app/index.asp>
* <http://www.williammalone.com/articles/create-html5-canvas-javascript-drawing-app/>