

Data Foundation Systems

Design Document

Data Annotation Tool - Moderation + Tool Box

Team DepthFirstSearch

Team Members:

Losetti Mourya (2021101121)

Gowlapalli Rohit (2021101113)

Gnana Prakash Punnavajhala (2021111027)

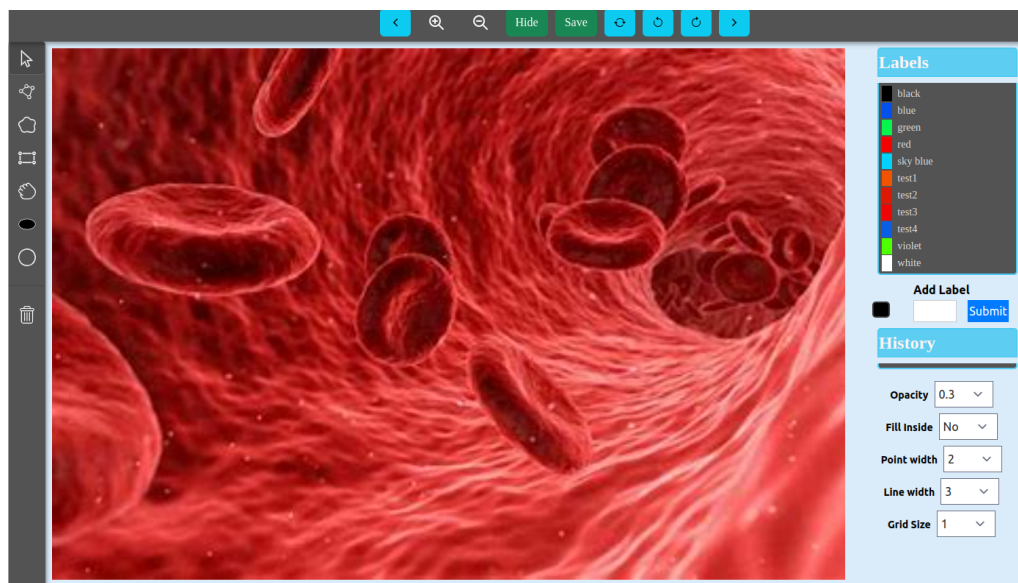
TA - Smruti Biswal

Design Document

This versatile image annotation tool encompasses two distinct operational modes: Annotation Mode and Moderator Mode. Through the incorporation of these dual modes, the platform fosters seamless cooperation between annotators and moderators. Annotators are empowered to focus on the creation of accurate annotations, while moderators can diligently review and refine the annotations to ensure their correctness and comprehensiveness. This collaborative methodology streamlines the annotation process, ultimately enhancing the overall quality of annotated images. The React framework serves as a robust groundwork for constructing this adaptable image annotation tool, presenting attributes such as flexibility, reusability, and development simplicity. By merging Annotation Mode and Moderator Mode, the project's objective is to simplify and optimize the image annotation process, rendering it an indispensable resource for tasks demanding precise and trustworthy image annotations.

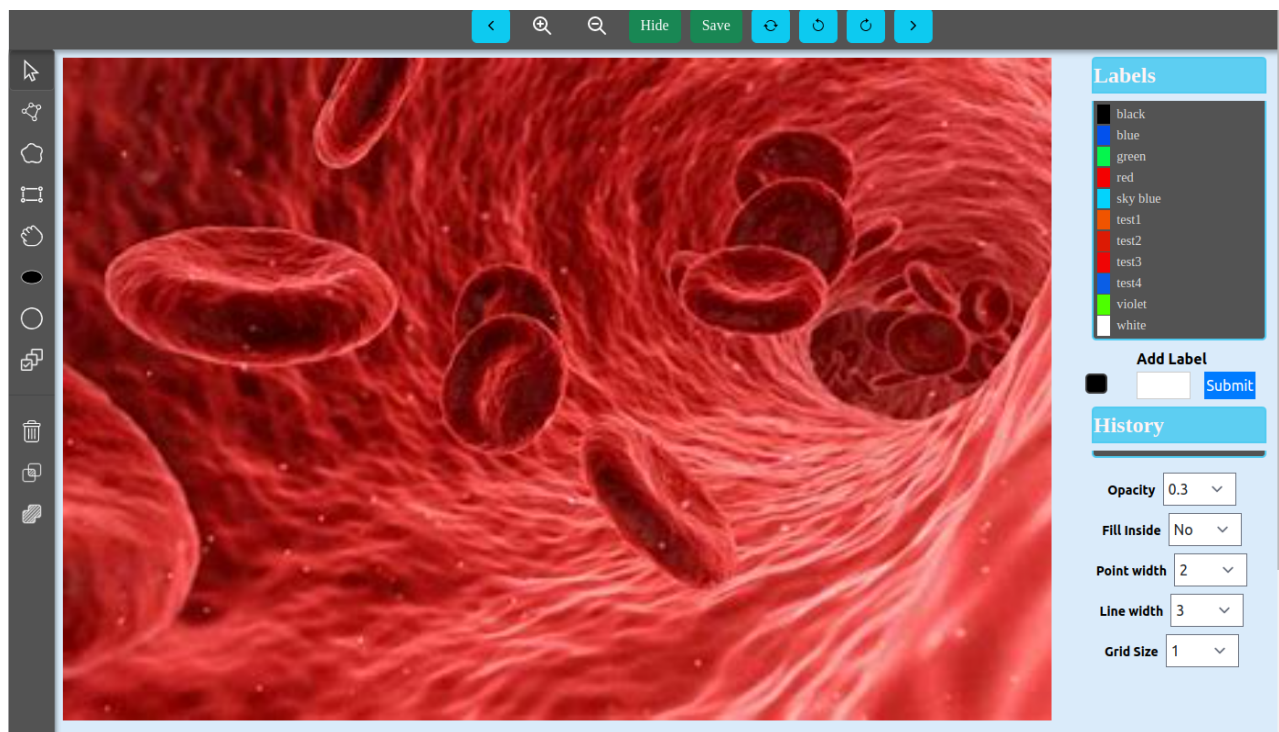
Annotation Mode:

Within Annotation Mode, annotators are equipped with the capability to draw and annotate image regions employing a variety of tools. This mode empowers annotators to pinpoint particular areas of significance, affix labels to them, and include relevant information.



Moderation Mode:

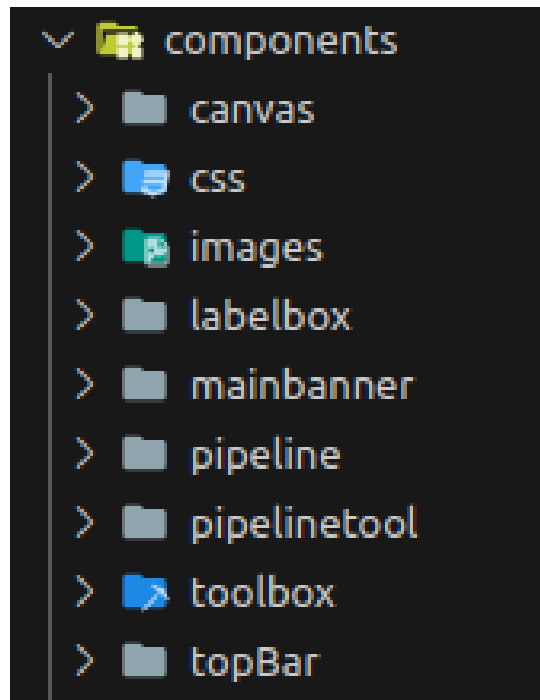
Conversely, Moderator Mode is tailor-made for moderators entrusted with the task of overseeing and supervising annotations created by multiple annotators. Moderators hold a pivotal role in upholding the precision and uniformity of annotations. They possess the authority to either endorse or rectify annotations, thereby affecting essential refinements to sustain quality standards.



Components:

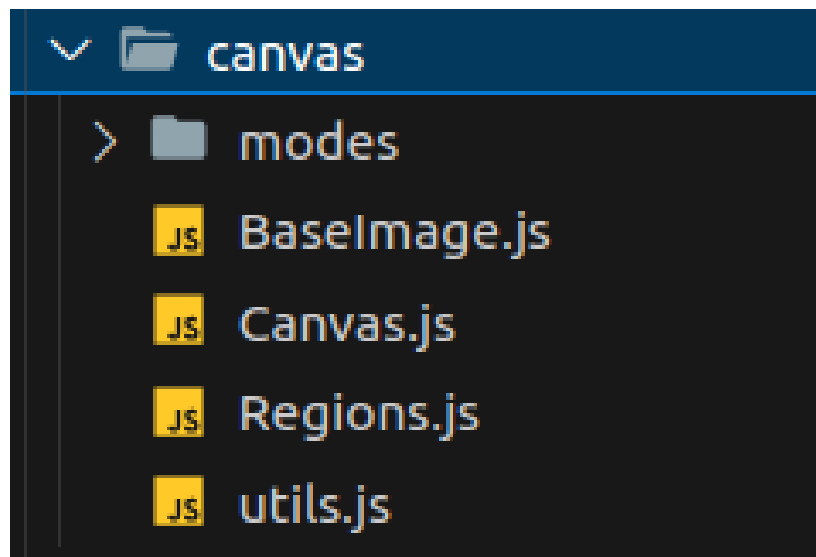
The project comprises numerous components that cooperate to provide a smooth and user-friendly annotation process. These elements are meticulously crafted to function in synergy, guaranteeing streamlined and efficient annotation workflows.

Directory Structure:



Canvas:

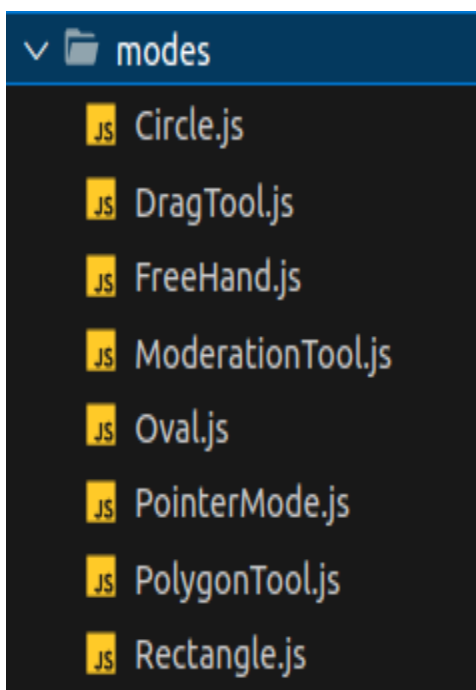
The primary element is the Canvas, which serves as the display for the image undergoing annotation. The Canvas consists of various sub-components, with the Base Image Component being responsible for presenting the unaltered original image.



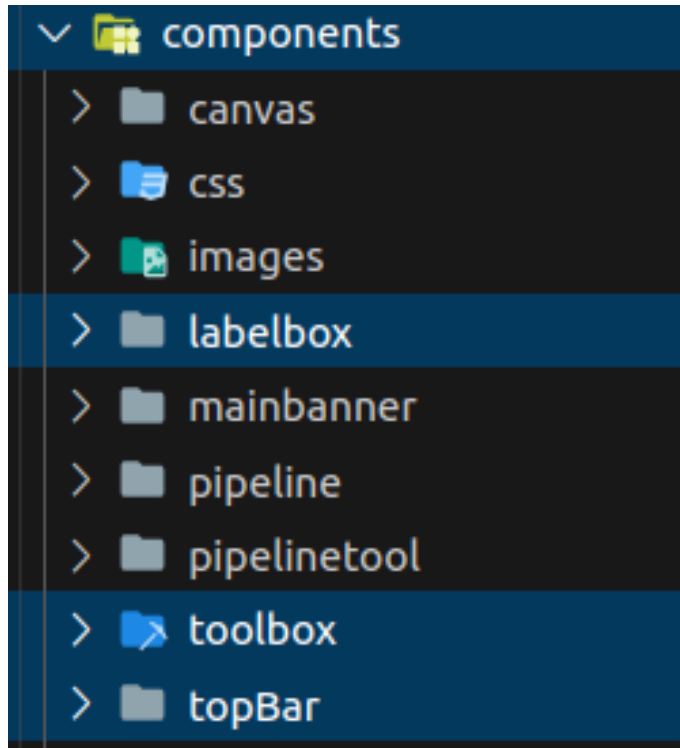
Modes of Canvas:

Every annotation tool is embedded as a distinct mode that governs the Canvas, facilitating distinct mouse event responses tailored to specific annotation functions. Transitioning between annotation modes initiates a shift in the mouse event listeners connected to the Canvas, ensuring that the selected mode behaves appropriately.

```
src > containers > imagepage > components > canvas > JS Canvas.js > ...
const setCanvasMode = (id, imageURL) => {
  switch (id) {
    case 0:
      return <PointerMode imageURL={imageURL} />;
    case 1:
      return <PolygonTool imageURL={imageURL} />;
    case 2:
      return <FreeHand imageURL={imageURL} />;
    case 3:
      return <Rectangle imageURL={imageURL} />;
    case 4:
      return <DragTool imageURL={imageURL} />;
    case 5:
      return <ModerationTool imageURL={imageURL} />;
    case 10:
      return <Oval imageURL={imageURL} />;
    case 9:
      return <Circle imageURL={imageURL} />;
    default:
      return <PointerMode imageURL={imageURL} />;
  }
};
```



Toolbars:



The interface incorporates supplementary components to simplify the annotation process:

- The Left Toolbar, also known as the toolbox component, provides various tools for drawing and annotating image regions.
- The Right Toolbar, referred to as the labelbox component, presents choices for selecting labels, colors.
- The Top Toolbar, referred to as the TopBar component, offers the ability to zoom in and out, provides collapsible panels on the left and right sides, allows for undo, redo, and reset of annotations, as well as the options to hide and save annotations.

In summary, this React project creates a versatile and user-friendly environment for image annotation. It introduces separate modes for annotators and moderators to enhance collaboration and precision in the annotation process.

Dependencies:

The Node.js project relies on multiple dependencies to bolster diverse functionalities and features. When these dependencies are integrated, they supply the essential tools, libraries, and utilities required for implementing the functionalities and features outlined in the project. This culminates in the development of a robust and effective image annotation tool within the Node.js environment. Below is an overview of key dependencies, linked to the previously mentioned components and modes of operation:

```
"dependencies": {
  "@headlessui/react": "^1.6.4",
  "@heroicons/react": "^1.0.6",
  "@mui/material": "^5.8.2",
  "@mui/styled-engine-sc": "^5.8.0",
  "@tailwindcss/forms": "^0.5.2",
  "@testing-library/jest-dom": "^5.16.4",
  "@testing-library/react": "^13.3.0",
  "@testing-library/user-event": "^13.5.0",
  "@uiw/react-md-editor": "^3.19.1",
  "axios": "^0.27.2",
  "bootstrap": "^5.3.0",
  "flowbite": "^1.5.3",
  "flowbite-react": "^0.3.4",
  "konva": "^9.2.0",
  "openseadragon": "^4.1.0",
  "polybooljs": "^1.2.0",
  "react": "^18.1.0",
  "react-bootstrap": "^2.4.0",
  "react-dom": "^18.1.0",
  "react-dropdown": "^1.11.0",
  "react-icons": "^4.4.0",
  "react-konva": "^18.2.10",
  "react-router-dom": "^6.3.0",
  "react-spinners": "^0.13.8",
  "reactjs-popup": "^2.0.5",
  "tailwindcss": "^3.1.3",
  "use-image": "^1.1.1",
  "zustand": "^1.0.7"
},
```

❖ axios:

Axios is a JavaScript library used for making HTTP requests from web browsers and Node.js applications. It simplifies the process of sending asynchronous HTTP requests and handling responses, making it a valuable tool for interacting with web services and APIs.

❖ bootstrap:

The Bootstrap framework provides a set of CSS and JavaScript elements designed for creating responsive web designs. These can be employed to elevate the project's styling and layout, contributing to an improved overall presentation.

❖ Flowbite and Flowbite-react:

Flowbite is an open-source UI component library that includes utility classes inspired by Tailwind CSS. Flowbite-React is the React-oriented version of Flowbite, offering a collection of ready-made UI components and utility classes for quickly developing user interfaces with a focus on style and responsiveness.

❖ konva:

Konva is a 2D drawing library that facilitates the efficient creation and manipulation of intricate scenes through the HTML5 Canvas. It can be harnessed to incorporate the drawing and annotation features into the Canvas component of the project.

❖ openseadragon:

OpenSeadragon is a versatile, open-source image viewer that offers high-performance capabilities for displaying and interacting with large and detailed images. It is particularly useful for zooming and panning within massive images and can be integrated into projects to enable advanced image viewing and exploration features.

❖ polybooljs:

This dependency offers functionality for conducting boolean operations on polygons, enabling the execution of geometric computations and tasks that might be essential for manipulating and processing annotated regions in images.

❖ react and react-dom:

React serves as the primary JavaScript library for constructing user interfaces, while react-dom is the package responsible for rendering React components within web browsers. Together, they establish the cornerstone of the project, facilitating the development of reusable and modular UI components.

❖ react-bootstrap:

React Bootstrap is a library that furnishes React components designed with Bootstrap styling. It enables the seamless incorporation of Bootstrap components into the project's user interface, ensuring a uniform and aesthetically pleasing design.

❖ react-dropdown:

React-Dropdown is a JavaScript library that provides a user interface component for creating customizable dropdown menus in React applications. It simplifies the implementation of dropdown functionality and allows for the creation of interactive and user-friendly selection elements within web applications.

❖ react-icons:

React-Icons is a JavaScript library that offers a wide range of popular and customizable icon sets as React components. It simplifies the integration of icons into React applications, allowing developers to easily include icons for various purposes, such as enhancing the visual appeal and functionality of their user interfaces.

❖ react-konva:

This library offers a React adaptation of the Konva drawing library, allowing for the seamless inclusion of Konva capabilities within React components. This, in turn, eases the development of interactive and customizable drawing elements.

❖ react-router-dom:

React Router is a library that empowers navigation and routing within React applications. It is handy for realizing transitions between diverse views, like navigating between images to annotate.

❖ react-spinners:

React-Spinners is a JavaScript library designed to provide a collection of animated loading spinners and indicators as React components. It simplifies the task of integrating loading animations into React applications, enhancing the user experience by giving visual feedback during content loading or processing.

❖ reactjs-popup:

Reactjs-Popup is a JavaScript library that offers a versatile and customizable modal or popup component for React applications. It simplifies the creation of popup windows, dialog boxes, and modal interfaces within React projects, providing a convenient way to present information or actions to users in a visually engaging manner.

❖ tailwindcss:

Tailwind CSS is a utility-first CSS framework for building responsive and highly customizable web interfaces. It simplifies the process of designing and styling web applications by providing a wide range of pre-defined utility classes, which can be easily applied to HTML elements. This framework streamlines the development of visually appealing and responsive websites while allowing for extensive customization and flexibility.

❖ use-image:

The use-image library supplies React hooks designed to simplify the loading and management of images. This aids in streamlining the handling of images within the project, making the process more straightforward and efficient.

❖ zustand:

Zustand is a state management library tailored for React applications, featuring a user-friendly and lightweight API for state management. It is valuable for controlling the state associated with annotation modes, toolbar choices, and various other components in the project.