Eshaan Chaudhari

# Software Development Internship at MEDO.ai: Technical Test Report

Eshaan Chaudhari

# Image Annotation Tool

## Problem
To create an image annotation tool that allows to annotate images based on interesting and not interesting properties.

## Requirements
1. Typescript and HTML
2. Follow the given format for JSON output

## Solution

### Technology
For this application, I used the Angular framework. Using this framework, I created components using Typescript, HTML and CSS. I have commented all of the files so the code can be easily understood.

### Instructions for running the application locally
1. Clone the repository at https://github.com/eshaanc20/Annotation-Tool
2. Using the terminal, navigate to the root of the project
3. Ensure the Angular CLI is installed. If it is not installed, run:

   ```
   npm install -g @angular/cli
   ```
4. Run the following command:

   ```
   ng serve
   ```
5. Open Google Chrome and the project can be viewed at http://localhost:4200/

### Instructions for using the application
1. Upload an image using the upload file button that can be found on the bottom left of the canvas
2. Once an image is chosen, the image will show up on the canvas
3. In order to draw rectangles, right click to create the rectangle, keep pressing down and drag the rectangle until the rectangle covers the part of the image and then stop dragging by unclicking
4. If the rectangle is incorrect, another rectangle can be drawn by following step 3 again, and the previous rectangle will be erased.

5. If the rectangle is correct, select “Interesting” or “Uninteresting” from the tool bar and the rectangle will change color. This indicates that the annotation has been confirmed, and the rectangle will remain permanently on canvas.
6. A green rectangle on the canvas indicates annotation for “Interesting” and an orange rectangle indicates annotation for “Uninteresting”
7. There are no limits to how many annotations or images can be uploaded
8. Click JSON button at the bottom right of the canvas to view the annotations in JSON format

**Design Decisions**

For annotations, only rectangles are allowed to be drawn on the canvas. The rectangles cannot be drawn if an image is not uploaded on the canvas. I have made the code flexible so rectangles can be redrawn until it is correct to help account for errors. In order to confirm the rectangle that is drawn, “Interesting” or “Uninteresting” button needs to be clicked to confirm the annotation. After confirming, the rectangle will permanently remain on canvas, and the annotation is added. Before confirming, the rectangle on the canvas will have a black outline. Once the rectangle is confirmed by pressing either of the two buttons stated above, the rectangle will change to green if the annotation is “Interesting” or orange if the annotation is “Uninteresting.” All of the confirmed rectangles for a specific image will remain on canvas permanently. I have added a button called “Clear” in the toolbar to remove all annotations for the current image that is on the canvas. For the JSON output, it will be displayed once the button is clicked. It will automatically update once new annotations are added. CSS styling is done to improve the user interface.

**Implementation**

All of the components can be viewed in src/app. There are two components. The Navigation component contains the information about the application. The App component has most of the code and contains the functionality for this application. Classes.ts file contains all of the classes used to handle annotations within the application.