Perelman School of Medicine University of Pennsylvania

”Phenotyping Carotid Artery Calcific Atherosclerotic Plaque”

Developer Team

- Cameo Dorais, `cameo.dorais001@umb.edu`
- Marc Dugas, `marc.dugas001@umb.edu`
- Kleopatra Gjini, `kleopatra.gjini001@umb.edu`
- Anthony Tran, `anthony.tran002@umb.edu`
- Steven Tran, `steven.tran003@umb.edu`

The Song Lab Team

- Jae Song, MD `Jae.Song@Pennmedicine.upenn.edu`
- Brett Cucchiara, MD
- Grace Wang, MD

Project Proposal v1.2 05/19/2022

1 Introduction

1.1 Purpose

The purpose of this project is to develop a web-browser based tool which will identify atherosclerotic calcifications in the carotid artery and calculate valuable data based on its findings. This tool will take in user input and calculate morphological features of calcified plaque such as the size, density, and distance from the carotid artery wall based on the passed-in DICOM image collections. This tool will be used by experts in the medical field, and will be especially valuable for neurologists, vascular neurologists, and vascular surgeons who are still researching the impact calcified plaque has in cerebrovascular ischemic events.

1.2 Scope

Carotid Calcium Scanner (CCS) is a web-based browser tool that will receive and process DICOM files and note potential calcified plaque developments in the blood vessel. The tool will also allow users to highlight areas on the image that they deem a calcified plaque location to be analyzed for relevant data computations and download a marked up version of the image and log file containing calculated data for their reference in cerebrovascular research and diagnostics.

CCS will not work with three-dimensional renders. It is currently not implemented with a Machine Learning algorithm. CCS will also not offer any medical advice regarding plaque calcification or the risks involved with excessive plaque built-up. CCS is a tool designed for medical professionals or researchers to identify potential plaque markers in images in order to diagnose or research the features of calcified plaque to inform future judgements on the stabilizing or destabilizing nature of calcified plaque.

CCS is working toward saving researchers time by automatically noting locations that have potential calcified plaque, but grants researchers the freedom to identify suspect markings on the passed in images. This will enable researchers to engage in other aspects of the research around the effects of plaque buildup as well as the effects that excessive calcified plaque can have on blood pressure, arterial flexibility, and other related studies. Currently, CCS will only allow users to manually highlight areas with calcified plaque and calculate the required morphologic features of the passed-in annotations.

1.3 Overview

The developers are comprised of a student group that is focused on designing software to identify markers of calcified, atherosclerotic plaque in the carotid artery. Atherosclerotic plaque is the build of plaque which is a mixture of fat, cholesterol, calcium and other substances within the arteries. Our focus in particular is calcified atherosclerotic plaque, which is the subject of many studies as it functions differently in the artery than fatty plaques. The goal of this project will be to design a tool that will allow users to note aspects of the DICOM image that they have determined as an identifier of calcified plaque and perform calculations that return the density, size, and volume of the plaque deposit.

The application can be reached: here.

1.3.1 Existing Work

Although there is a vacancy for software that are specific to identifying carotid atherosclerotic plaque, there are applications designed to view and modify DICOM images currently available to interested users.

- OHIF (Open Health Imaging Foundation *A web-based application used to view medical images.*
- DWV (DICOM Web Viewer) *Software that allows users to load and analyze DICOM images.*

These items are DICOM viewers that must be downloaded in order to function:

- MicroDicom *Free software that processes DICOM files for viewing.*
- RadiAnt DICOM Viewer*DICOM file viewer*

1.3.2 Limitations

- CCS will work for series of 2D images but not 3D renders.
- CCS will not be based on any machine learning algorithm.
- CCS will only recognize calcific plaque deposits; it is not designed to return details of morphologic features of non-calcified (fatty) plaque buildup in the lumen.
- CCS cannot be used to determine inflammation or lesions in the arterial wall, but information pertaining to inflammation, legions, or other arterial damage can be ascertained from information gathered regarding plaque size, density, and clustering.
- Users should only upload stacks of images a single time without reloading or it will affect brush segmentation values and painting abilities.

1.4 Definitions

This project is based heavily in the field of vascular health, and so a lot of medical terminology is used to ensure that explanations of processes are precise and accurate. As such, necessary medical terminology is also included in our definitions alongside any acronyms used when describing the CCS tool.

- CCS: Carotid Calcium Scanner
- JS: JavaScript
- HTML: Hypertext Markup Language
- CSS: Cascading Style Sheets
- UI: User Interface
- CC: Computational Code
- DM: Data Manager
- MLA: Machine Learning Algorithm
- OHIF: Open Health Imaging Foundation
- DICOM: Digital Imaging and Communications in Medicine. Type of input file used by CCS.
- CT: A Computed Tomography scan combines a series of X-ray images taken from different angles around your body and uses computer processing to create cross-sectional images (slices) of the bones, blood vessels and soft tissues inside your body
- CTA: Computed Tomography Angiography
- PET: Positron Emission Tomography scan
- Plaque: The buildup of fats, cholesterol and other potentially harmful substances in and/or on the artery walls.
- Atherosclerosis: Thickening or hardening of the artery caused by a build up of plaque on the inner artery lining.
- Calcifications: Calcium that has built up in a body tissue, which causes it to harden. Calcification of the arteries causes the arterial wall to stiffen and become inflamed, which can result in chunks of calcified plaque to break off and obstruct blood flow to the brain.
- Cerebrovascular Ischemic Event: A term for when the flow of blood to the brain is reduced to a point where it cannot meet its metabolic demand. This leads to a reduction in oxygen flow to the brain, death of brain tissue, stroke, or death.
- Stroke: the sudden death of brain cells due to a lack of oxygen. The lack of oxygen is generally due to a blockage of blood flow or rupture of an artery connected to the brain.
- Lumen: The cavity or channel within a tube or tubular organ such as a blood vessel or the intestine.
- Stenosis: An abnormal narrowing of a channel in the body; in this case, referring to narrowing of the artery due to atherosclerosis.

2 Requirements

This program will work within an internet browser that will take images passed in by the user. The user will highlight areas of potential calcified plaque that the program will use to calculate various morphologic features of and return quantitative data for the user. The output of this program is a json file with tool state configurations that in the future will allow users to return to their previous annotations and continue where they left off. The user will be able to interact with the image using a limited tool set and manipulate the image to determine morphologic features of the plaque such as volume, size, total number of identified bodies of plaque, and density of the plaque.

The user will need to be connected to the internet in order to access this application. Its performance will be dependent on the user's personal machine, but the UI is designed to be easy to understand and navigate for any level of computer skill. The tools needed to manipulate the DICOM are shown with recognizable symbols and text, and there is always a help icon available in a clear location for users to help them should they become confused. All buttons are large and bright to easily convey what they do when interacted with. Complicated calculations and programmatic approaches are totally invisible to users to put a veil between users and the daunting "guts" of the program.

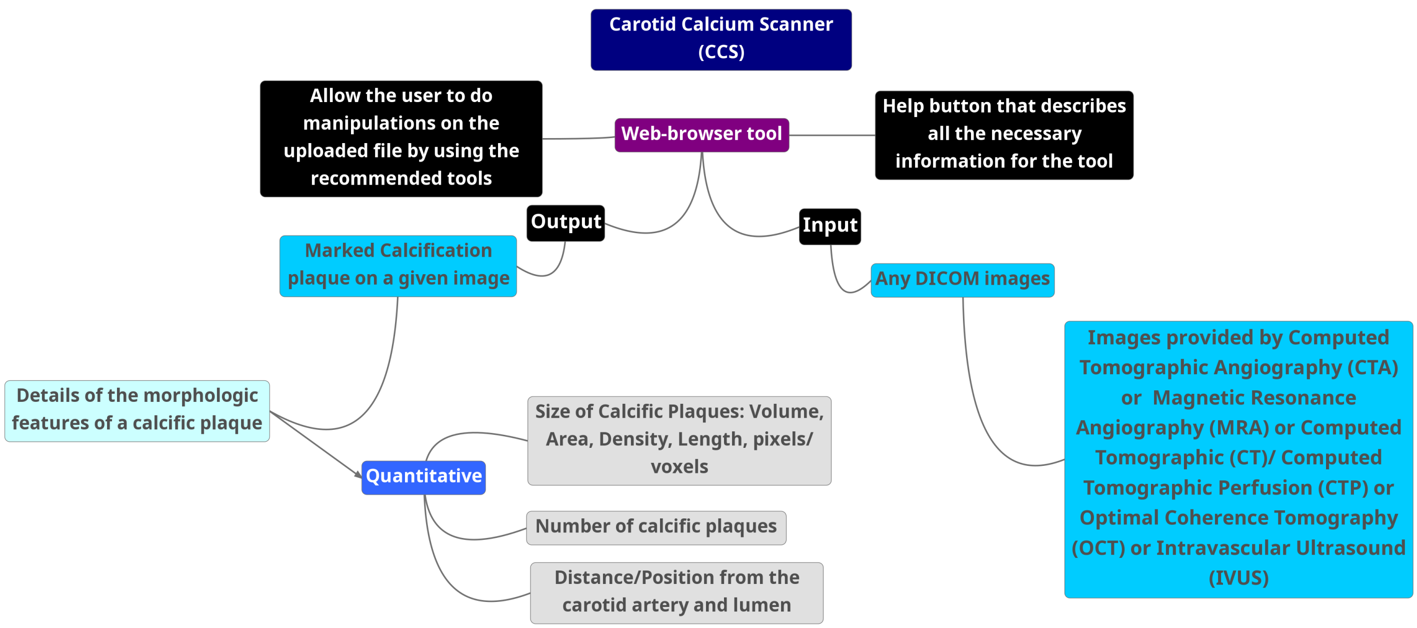

Figure 1: MindMap: The program Carotid Calcium Scanner (CCS) is a web-browser tool that will include a info button with all the necessary information, and will allow the user to manipulate any DICOM images. The results from this tool will be a JSON file containing the configurations of the tools.

3 Specifications

CCS will be a web-based software that will be implemented using JS, CSS, and HTML, with the potential for a machine learning component written in python in the future. It will make use of preexisting medical imaging libraries to interact with and display a passed in DICOM files (generated from a CTA, PET/CT, or IVUS). The user will then note brightened areas in the carotid artery that are known to represent calcified plaque deposits. The program will then compute necessary statistics and return the calculations based on the pixel signal intensity, total number of notable pixel clusters, and size of the pixel clusters. The volume of a plaque deposit will be determined by taking the area of each highlighted section of the image in a specific series and the height of the plaque deposit, found by calculating the difference in location of the first and last image of the plaque and performing disc integration. The modified DICOM files will be displayed to the user where they will have access to tools necessary to highlight additional areas of calcified plaque as well as image manipulation tools such as zoom, pan, rotate, and filter. Calculations will be done based on the areas denoted by the user to return the relevant morphologic features. Once the user has finished annotating the modified DICOM files, the user will be able to download a json file containing the tool configurations.

4 Design

4.1 Use Cases

The audience of this tool will initially be researchers at the University of Pennsylvania's Department of Radiology and will eventually be expanded to any other consumers interested in this open source software, primarily focused on research on calcification of plaque in the carotid artery. Preconditions of using this tool include access to the internet with a modern, up-to-date web browser such as Google Chrome or Microsoft Edge, and a series of DICOM images to be passed into the program for analysis and modification.

The primary Use Case for this tool is image manipulation of DICOM files to better identify calcific plaque in the carotid artery to reduce necessary time allocations for manual identification of plaque deposits and allow researchers to better allocate time for other research tasks.

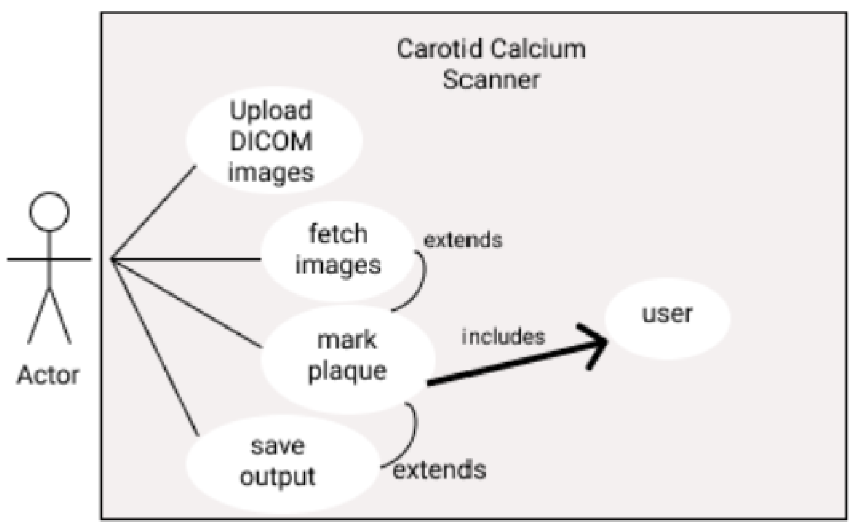

Figure 2: Use Case UML Diagram: The Consumer will upload a DICOM image into the web-browser. CSS will then fetch the image sent, mark the found plaque by user input and perform the needed calculations on what is found. Finally the annotation data will be saved into a JSON file.

4.2 Architecture

The code behind CCS is designed to create a user friendly tool that will enable the client to analyze and annotate their DICOM images for further research into calcified plaque in the carotid artery. The User Interface (UI) class will be tasked with handling the user upload and input, including image manipulation and annotation. The User Interface offers a drag and drop area for uploads and a menu that include tools such as a brush, zoom and pan tool to allow for image manipulation with relative ease. The User Interface will call upon web-based classes that will handle image and data uploading and the analyzing and manipulating of the properties of passed-in image files.

Viewer is a class that handles the storage of compressed images and quantitative data and eventually the machine learning processes on the current model. The Viewer allows the client to analyze and manipulate the images and gather pertinent data through various displays, highlights, and other edits that will be available. Once the user has finished editing their images, users will be able to save their state by downloading a json file containing all necessary tool configurations and annotation information regarding their uploaded DICOM images.

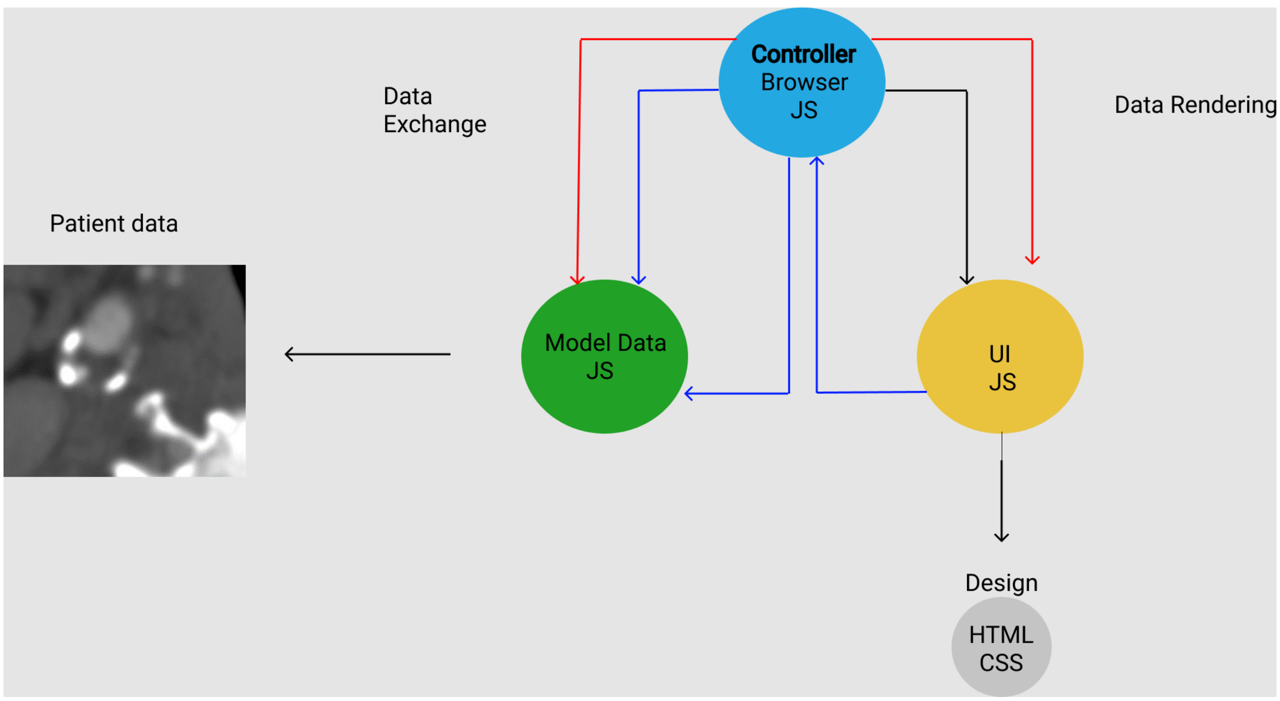

Figure 3: Our MVC Diagram illustrates the connection between the Data Model, User Interface (UI) and the Controller that handles the data processing.

4.3 Workflows

Users of the CCS tool seek to analyze series of DICOM images for markers of calcified plaque and empirical calculations based on recognized deposits of calcified plaque. This is achieved by first drag and dropping an appropriate DICOM file to the CCS tool. Once the file is in the viewer, the user is then able to view the various attributes of said image. They may add their own highlights and annotations that the computational code can use to complete necessary calculations. The user then has the option to download a json file containing the necessary tool configurations and image pixel information.

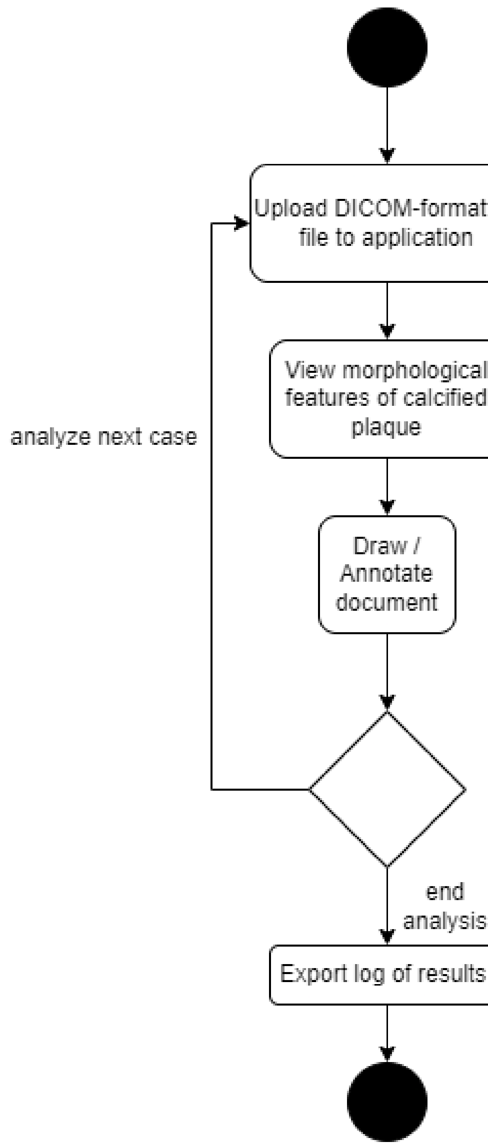

Figure 4: Activity UML Diagram shows the path starting from when the user uploads their DICOM series and ending with the exportation of the modified images once users have completed their annotations and received the desired morphologic data.

4.4 Technologies and Implementation Details

4.4.1 General Information

Our modelling and prototypes are all designed on figma.com. Our code is hosted on GitHub in a public repository for anyone to review and access. Any open source code used for this project is readily linked in the supplemental documents section of this document. The program is client-side and does not rely on a web server, so the hardware necessary for this program is reliant on the user's own hardware.

4.4.2 User Interface

The web page will be written entirely in a mix of HTML and CSS for formatting and web page design and JavaScript for actual information processing and communication with the main body of the program. Our user interface is meant to be clean and simple, but easy enough to understand that non-experts and ”non-computer people” can easily navigate between the home page and the image manipulation page.

4.4.3 Computational Code

Our functional code is centered around JavaScript- based image manipulation APIs as well as mathematical libraries housed within the Cornerstone libraries. Here is where the program really shines. The program will analyze marked locations and compute data on pixel intensity, size of the pixel cluster, and total number of clusters and calculate the necessary outputs to be passed to the Data Manager to be returned to the user. Any newly identified plaque in the images as indicated by the user will be analyzed accordingly and the information given to the controller to be appended to the log file.

4.4.4 Automated Identification

Once the initial version of CCS is functioning properly with user input, we would like to extend the program to automatically note areas that show signs of calcific plaque build up and perform the necessary calculations without the need for user annotations previously. It will still allow for user annotation once the modified DICOM is displayed to the users but will do a preliminary mark up of all images in the series to minimize the amount of work necessary by the user.

4.4.5 Machine Learning

Our ultimate goal is to implement a machine learning algorithm that can learn from user input to better determine all calcified plaque deposits and reduce the amount of corrections needed by the user. This would also be written in either JavaScript or Python and would rely heavily on either Pytorch or TensorFlow.js, which are both preexisting libraries that focuses on Deep Learning. Both are widely used with large bodies of work to reference and clear proof of successful implementation, but at our current stage of development, more research will be needed to determine which library will be a better fit for the task at hand.

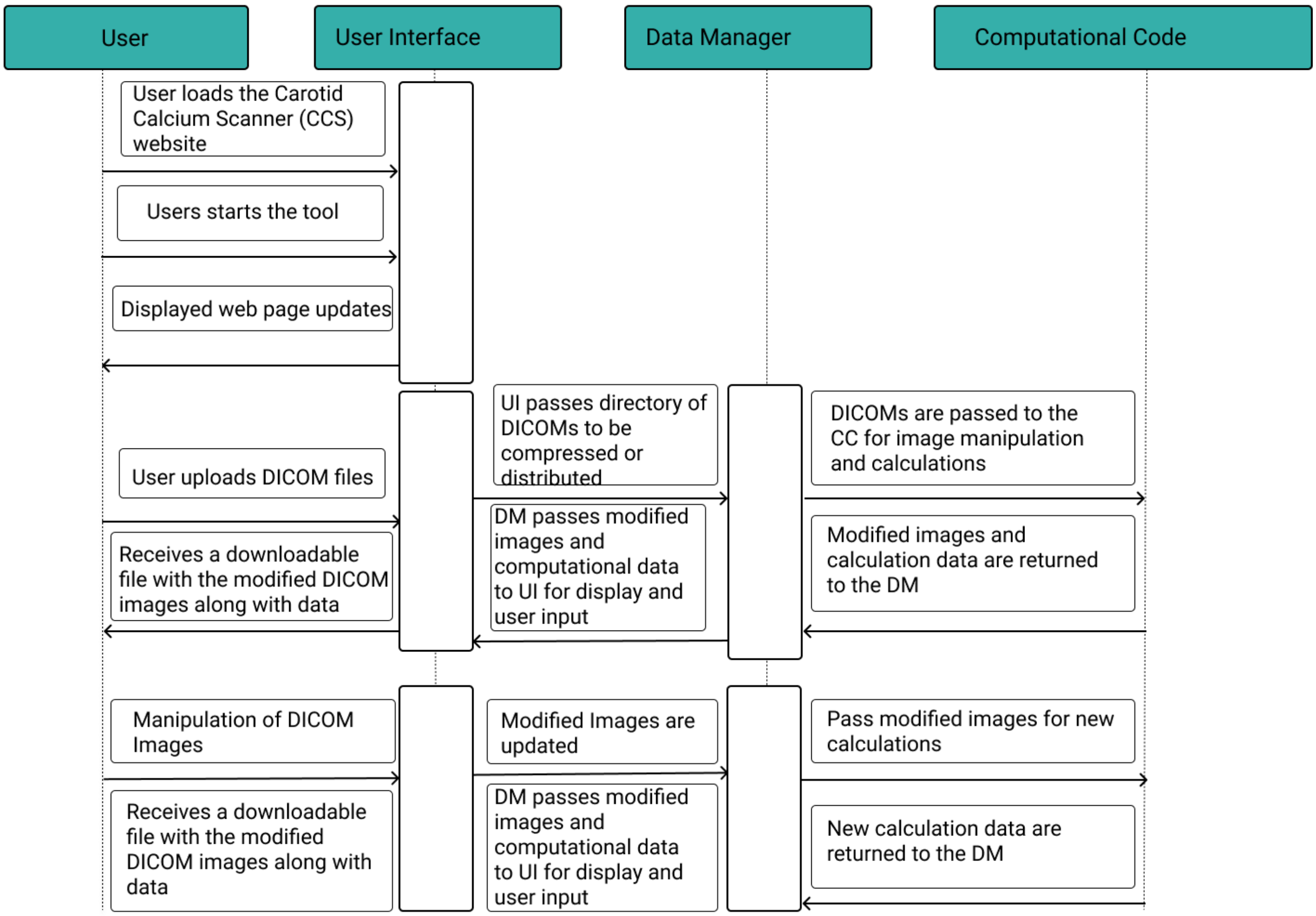

Figure 5: The Sequential Diagram shows the series of actions completed by various actors or aspects pf the program, denoting which aspect or actor starts each action and the steps required to complete the action.

4.5 User Interface

Our User Interface (UI) is going to be as user friendly as possible. As dark mode is becoming more and more popular among users and is more energy efficient than its counterpart light mode, our background will be a dark background with accents of our feature color #35AFAA and images of CTA scans. In the center of the screen, our home page will offer a brief summary of what the tool will do and a ”Get Started” button that will take the user to another page that will allow users to upload their collection of DICOM images for analysis. At the top of the home page, there is the tool’s logo, and in the bottom right corner, there is a link to an info page connected to the text ”About Us”. ”About Us” offers information about the student team developing this tool, the application itself, how to use the annotation and manipulation tools, the clinical relevance of studying calficied plaque in the carotid artery, and some definitions for terminology that might be encountered during research.

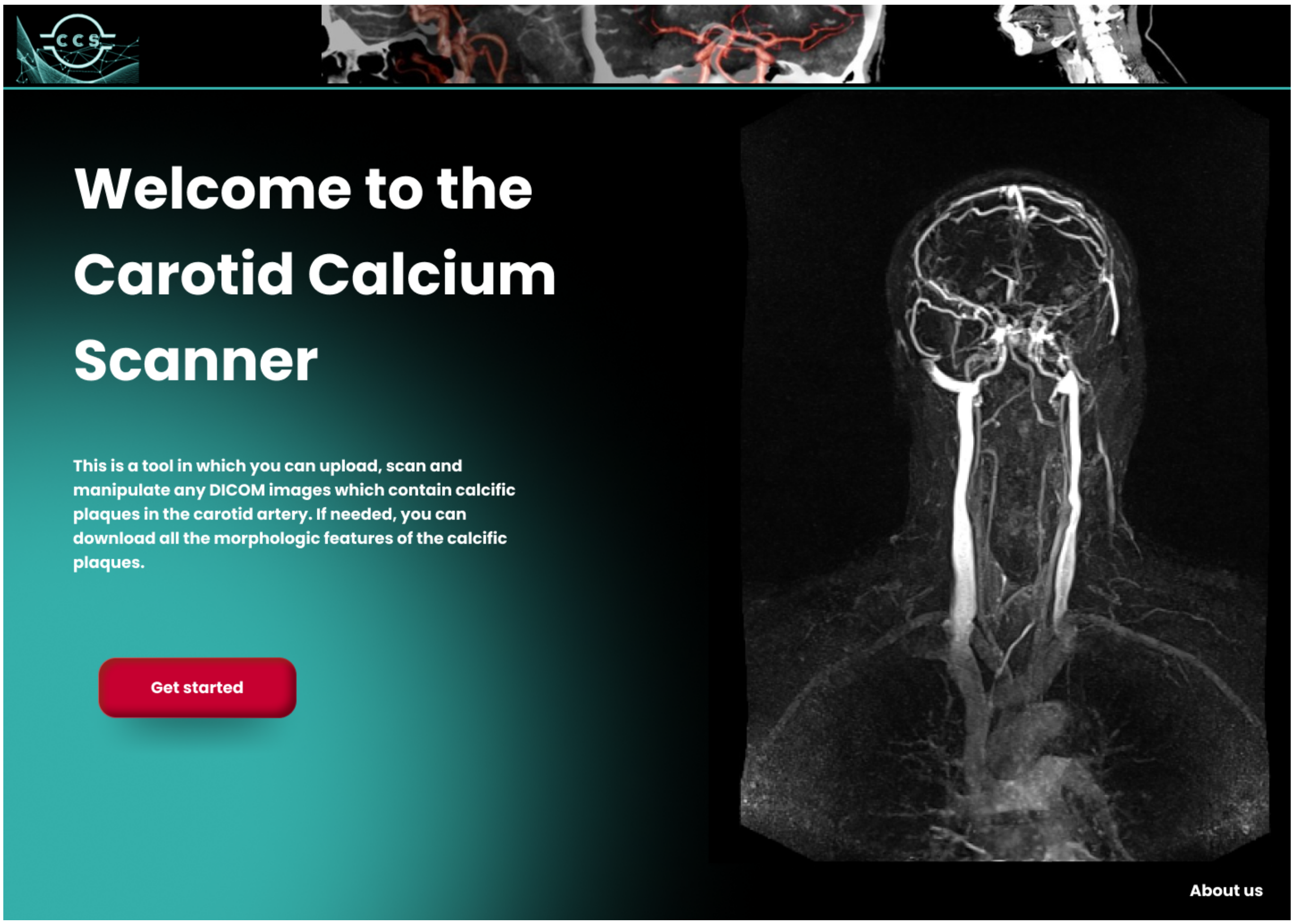

Figure 6: User Interface 1: Example of a Home screen

Once users choose to ”Get Started”, they will be taken to a second page where they can upload their collection of DICOM images. In the middle of the screen, there is a grey outline of a box with the words ”Drag and Drop” in the center to invite users to drop any number of DICOM files or a json file. The info button is at the top right of the screen, indicated by an ”i”.

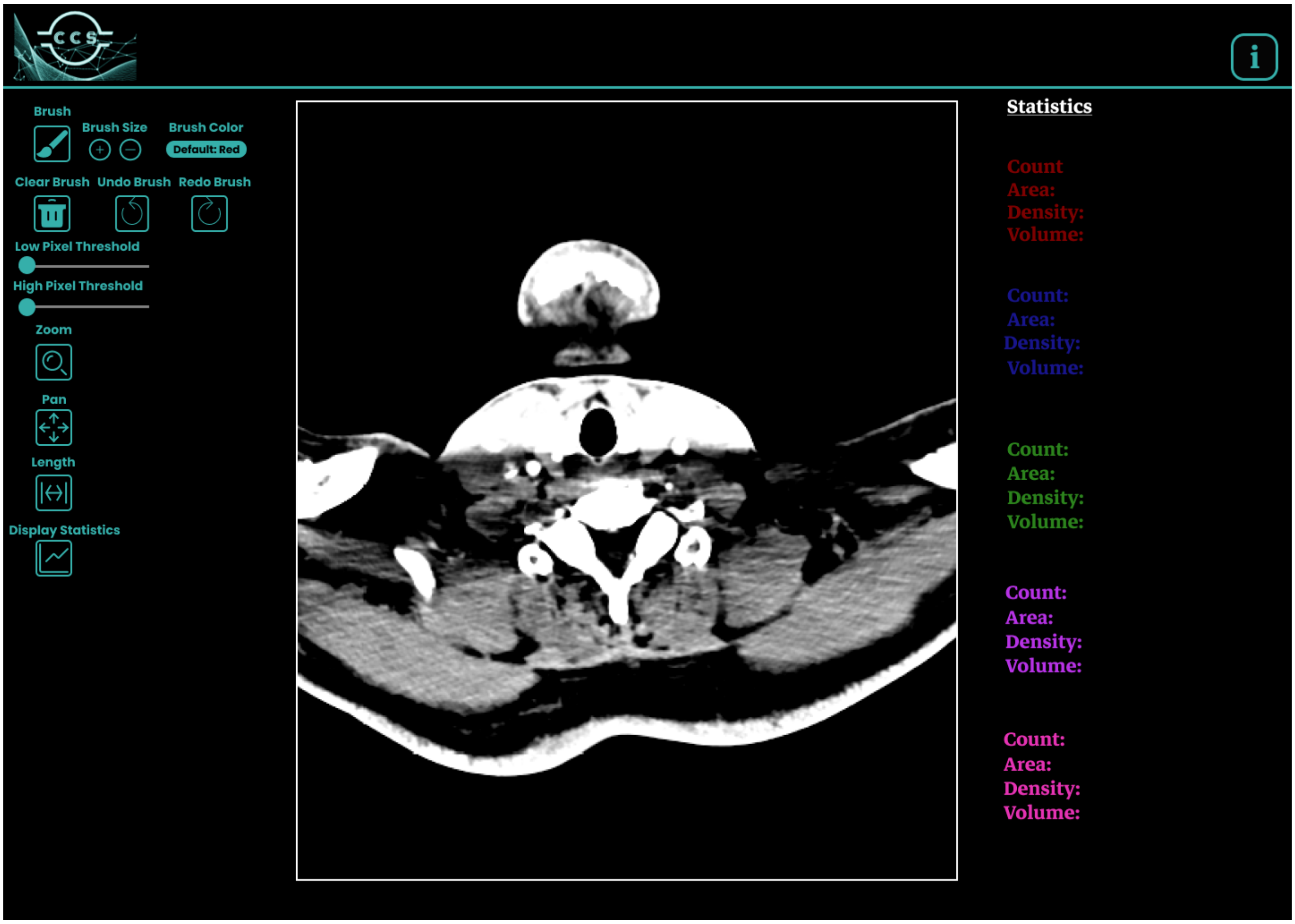

Figure 7: User Interface 2: Example of a File upload screen

Once the images have been passed into the program, the images will be displayed in the gray box outline, and users can view the DICOM images and add edits using editing tools offered at the top of the page. Each icon for the toolbar has a description of the tool, and users can learn more about using these tools on the ”info” page.

In the toolbar, users will have access to:

- A panning tool to move around the selected image
- A zoom tool to magnify or reduce the image.
- A brush tool to annotate the image.
- A drop-down menu to allow users to select between 5 paintbrush colors
- Size adjusting buttons to increase or decrease the brush size.
- A clearing tool to remove user annotations.
- An undo and redo function for paint brush annotations.
- A Length tool to take various length measurements such as distance from the lumen.
- A Display Statistics button to display the calculations completed after the user completes their annotations.

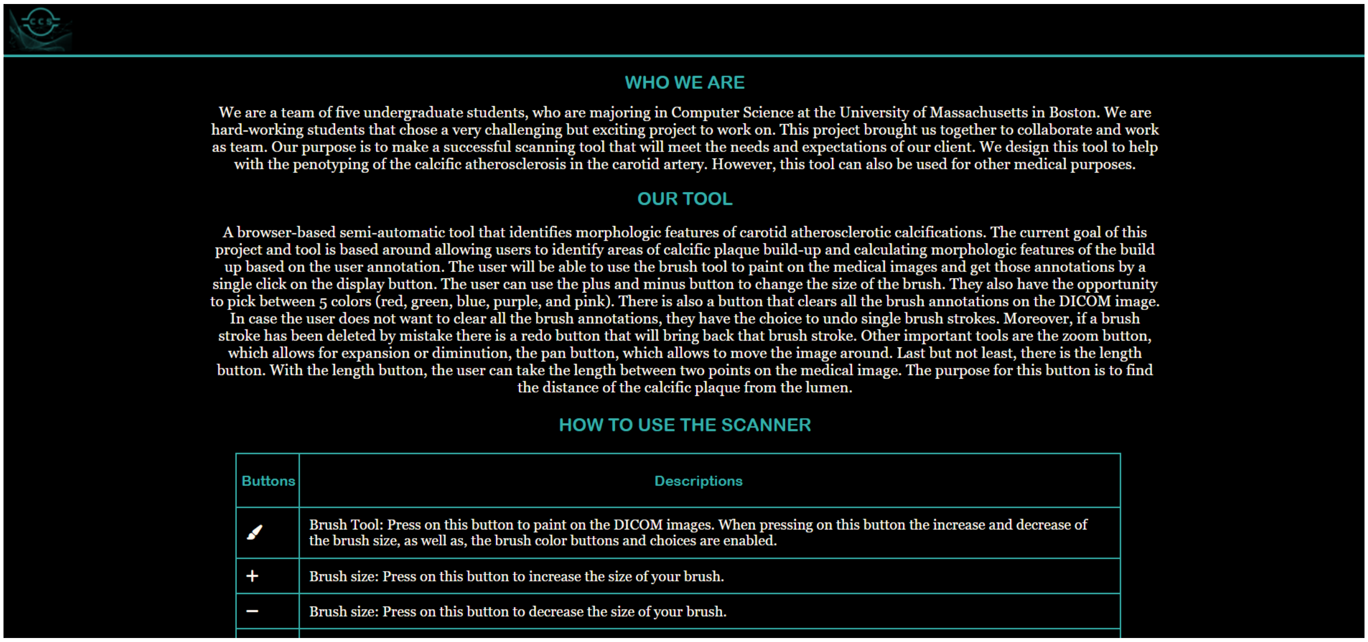

Figure 8: User Interface 3: Example of a plaque highlight screen complete with a multi-functional toolbar

With these tools, users will be able to manipulate the passed in images and receive information of the necessary quantitative features of the plaque buildups in each image. Once the user is satisfied with the collected data they have requested, they can download a compressed file of the modified images by pressing the ”q” button on their keyboard. This will download a JSON file that contains the tool configurations that house all annotations and markups completed by the user.

5 Timeframe and Milestones

Subject to Change

- 2/25: Initial Project Proposal
- 3/25: Revised Project Proposal
- 4/1: Set up preliminary DICOM viewer
- 4/8: Set up stack scroll tool
- 4/15: Implement paint brush tool with 2 colors
- 4/22: Implement a pixel counter to work with paintbrush tool to determine statistical features
- 5/13: Project Presentation
- 5/11-5/20: Implementing slider tool for threshold adjustment and developing tool state update feature from json file upload
- 5/20: Final Product Documentation

6 Supplemental Documents and Notes

Reference Material on Atherosclerosis and Plaque Calcification

- Calcification in Atherosclerotic Plaque Vulnerability: Friend or Foe? by Xuan Shi et al.
- Definition of Lumen
- The Song Lab
- A Primer of Calcium Scoring
- Github Repository
- The different types of DICOM images
- Homepage GIF
- Patterns
- Open Health Imaging Foundation (OHIF)
- Tensorflow.js
- Sample Icons