|  |  |  |
| --- | --- | --- |
| Functional Requirement Number | Functional Requirement Description | Functional Requirement Function(s)/Class(es) |
| FR1 | The system must be able to load DICOM files. | capapp.StartPage   * Class containing the tkinter GUI where the user will be able to load the DICOM files and make predictions. * The page will contain two buttons (load and predict) to allow the user to load the file and use the loaded data for prediction * Tkinter allows one to “load” a file by simply recording the absolute file path as opposed as literally loading the file as an object into the program. |
| FR2 | The system must be able to make predictions based on loaded DICOM files. | capapp.Startpage.predict   * Function within the custom GUI class used to begin the prediction procedure.   pipeline.obtain\_data   * Function which uses the pydicom library to read, process and extract useful data from the DICOM files. * This will load the file and process the data for model input. * Activated upon the user interaction with the predict button on the main page GUI.   models.TumorClassifier   * Custom class based on the TensorFlow model class that is used to classify the image found within the DICOM file. * Uses a combination of convolutional neural networks, pooling layers, and linear layers for final classification. * Outputs the prediction for use. * Activated once data is fully extracted from user selected files. |
| FR3 | The system must be able to display data extracted. | capapp.MainDashboard   * Displays the data within a dashboard format. * Contains a list of image files and some basic data related to the image file. * The list will also have a button to the right to allow the user to another frame that displays the targeted image. * Allows download of a report through the Download Report button. This button activates a function within the MainDashboard class that downloads the observed report locally.   capapp.DetailedDash   * Displays the image with some data on the patient to the right of the image. * Allows user to pass through images by clicking the Next button or Back button |