CS3249 User Interface Design

**Final Report**

**The Microscope**

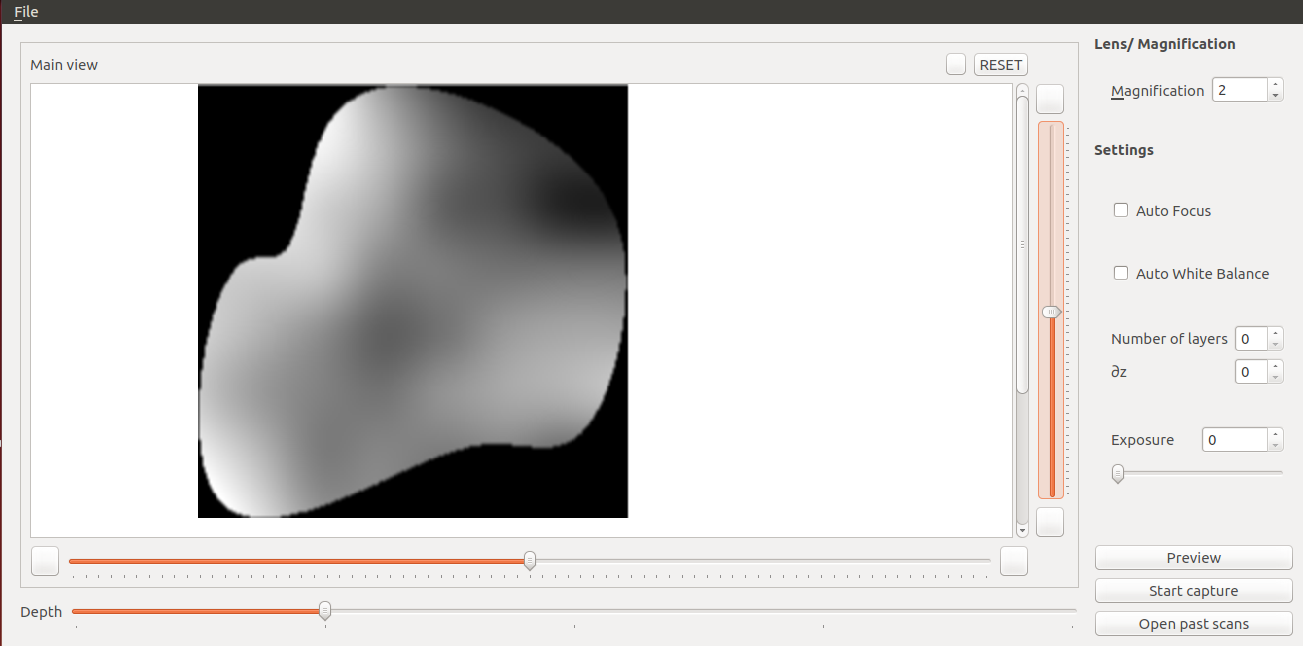
Team QX

Kit Wei Min

Wong Wei Jie Bob

Oh Qi Xuan

# GUI Design



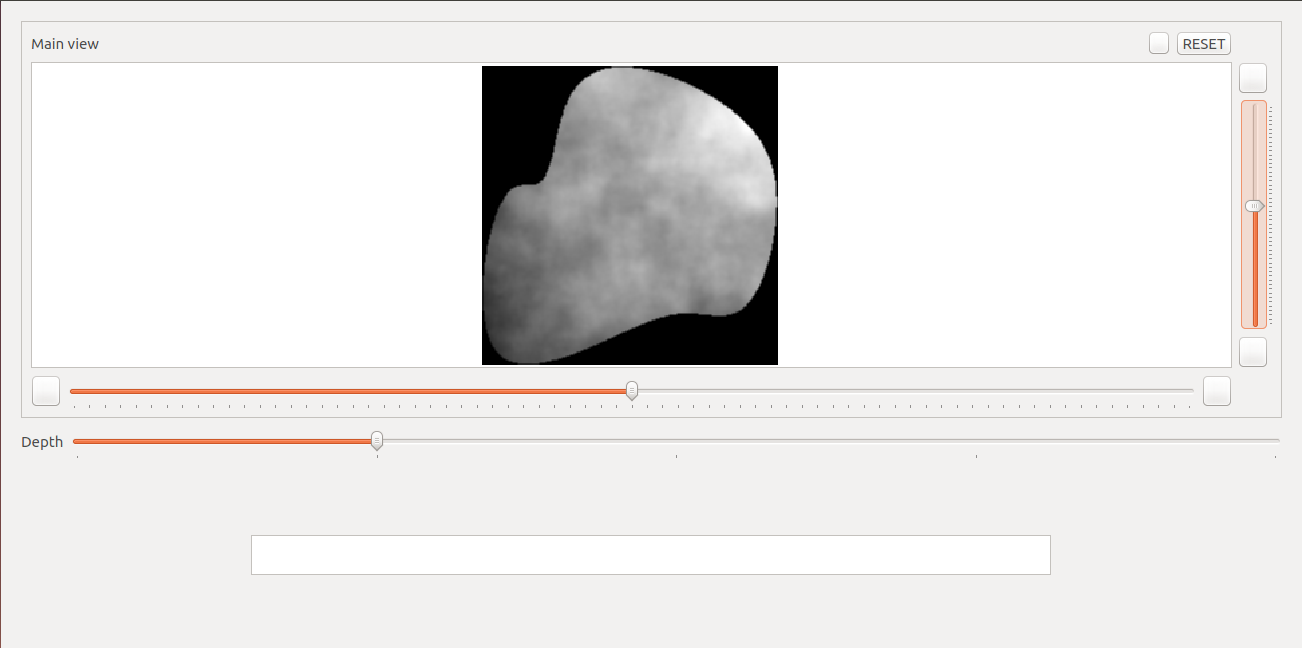
Based on the initial microscope design from tutorial 1, we are proposing a revamped GUI with several elements that emphasize aspects of the program important to the user, and de-emphasize lesser-used but still important elements.

First of all, instead of a large preview button, most of the space is taken up by a frame intended to contain the preview image. This frame updates only when the Preview button is pressed, to allow the user a fixed frame of reference when adjusting settings. We believe this is the main focus of the application and should be sized accordingly, as opposed to being assigned to another button as in the prior design. Allowing the frame to be in the same window as most of the application’s options enables fast review of changes. The frame is positioned on the upper left, as this allows it to be viewable easily at eye level as opposed to being positioned lower down, which may necessitate looking down at a slight incline and aggravate neck strain and user irritation.

Magnification is located on the top right of the GUI, as an important but infrequently-used component. Though a slider may intuitively be an option, we believe that this would only introduce unnecessary clutter to the interface, as most uses of microscopes have different magnification options purely for versatility in perusing different samples, and do not require fast iteration through different magnification options. A spin box is sufficient to both allow modification of the magnification and display it.

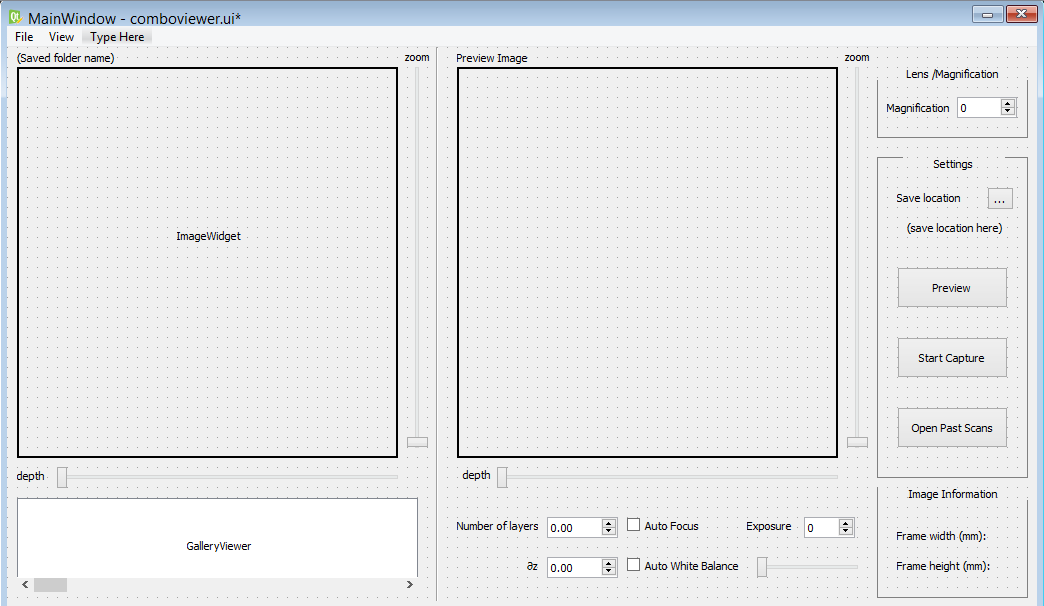
Settings are located below magnification, and contains options to select save location. Buttons for Preview, Start Capture and Open Past Scans are located at the bottom right hand corner, as they are the last thing that the users will look for when they use the program.

Selecting Start Capture will open up a dialog that informs the user that the scan is ongoing. Selecting Open Past Scans will open the file dialog for the user to select the folder storing the scans that the user is interested in. The GUI will then be updated to a new view as shown below.



The main viewer now occupies the bulk of the window box. Sliders are used to zoom in on the image and to scroll through the different layers, though for advanced users, controls using the mouse and keyboard will be implemented as well. Users can also get a preview of all the different layers with the photo gallery located below the main viewer, and can select a particular layer by clicking on the image as well. The scrolling of the GalleryViewer will be synced with the slider that controls depth, so that the user can have a view of the surrounding layers while scrutinising a particular area in the main viewer. The GalleryViewer is not fully implemented yet.

We realise that users might want to be able to do side-by-side comparisions between past scans and what they are observing real-time, and thus we are offering a third view, with both views combined and placed alongside each other for easier comparision. This is shown below.



(old concept image of comboview)

Unfortunately, we are unable to get the ComboView up in time, but it can be a possible future implementation once the rest of the program is more stable.

# Software Architecture

# softwareArchitecture.png

The ImageWidget is a common widget used by the PreviewViewer and the ImageViewer. It communicates with its parent through QT's slots and signals, allowing for loose coupling between the components.

FileHandler that handles all files used by our program. This allows for changes to be made to the type of files that the Microscope can read without affecting the rest of the program, allowing the program to be extended without affecting the rest of the program.

# Implementation Strategy

PreviewViewer is the main window of the Microscope and will be shown when the program is first opened. QMainWindow is thus used for it.

ImageViewer is a QWidget that can be shown and hidden from the PreviewViewer. ImageWidget is also a QWidget that is a child of both ImageViewer and PreviewViewer.