

# CMPT 371 – Team 3 Testing Document

Virtual Reality Medical Imaging Software with Luxsonic Technologies  
Inc.

## **1 Introduction3**

## **2 Delta Changes4**

- 2.1 ID 2 Changes4
- 2.2 ID 3 Changes5
- 2.3 ID 4 Changes6
- 2.4 ID 5 Changes //TODO6

## **3 Testing Shift6**

## **4 Smoke Testing7**

- 4.1 Verify all dependencies (libraries, imports, etc)7
- 4.2 User gets into the default scene8

## **5 Manual System Testing8**

- 5.1 Open File Browser //TODO8
- 5.2 Close File Browser //TODO9
- 5.3 Navigate through directories9
- 5.4 Load DICOM Image from File Browser //TODO10
- 5.5 Create a Copy from the Tray //TODO11
- 5.6 Move Copy //TODO11
- 5.7 Resize Copy //TODO12
- 5.8 change copy brightness //TODO13
- 5.9 Change Copy Contrast //TODO14
- 5.10 Invert Copy Colours //TODO15
- 5.11 change copy saturation //TODO16
- 5.12 select/deselect image //TODO17
- 5.13 select/deselect all images //TODO18
- 5.14 Close a Copy //TODO19
- 5.15 Restore Image to Default //TODO19
- 5.16 Quit the Workspace //TODO20
- 5.17 Minimize Display/Dashboard //TODO20
- 5.18 Maximize Display/Dashboard //TODO21
- 5.19 Move Left and Right Through Images in Display //TODO21
- 5.20 Button Presses/Depresses //TODO21

## **6 Manual Regression testing22**

## **7 User testing22**

## **8 Acceptance Testing23**

- 8.1 Quit Workspace23

8.2 Move Copy	23
8.3 Show/Hide Dashboard	23
8.4 Create Copy	24
8.5 Close Copy	24
8.6 Show File Browser	24
8.7 Close File Browser	24
8.8 Load from File Browser	24
8.9 Remove from tray - Not implemented	24
8.10 Remove All From Tray - not Implemented	24
8.11 Slide Through Images in Display - not implemented	25
8.12 Next/Previous Image in Display	25
8.13 Change Brightness	25
8.14 Change Contrast	25
8.15 Change Saturation	25
8.16 Resize Image	25
8.17 Select/Deselect copies	25
8.18 Buttons Press and Depress	26
8.19 Select All and Deselect All	26
8.20 Navigate through Directories	26
8.21 Invert Copy Colours	26
8.22 Restore Image	26

## **9 Testing Matrix**

# 1 Introduction

The Testing Plan outlines what types of tests we will be running and how we will be performing them each iteration. This document also contains the delta changes from the last iteration as well as information about our shift in testing and the alternatives that we have implemented. We outline the smoke tests, system tests, and integration tests along with how they are each being performed. Additionally, regression testing will be documented as issues in the code base arise and fixes are implemented. All Automated tests will be within the folder Assets/UnityTestTools/UnitTests/Editor because of how Unity Test Tools searches for tests in that specific folder. User testing practices are outlined in this document along with some of the results, findings, and solutions of our user testing sessions. Acceptance testing methods and practices are documented here along with any important notes or changes to them. Lastly, this document will contain a plan for a testing matrix.

All tests will be written in this document as to how we will be manually performing them and using the design document as a reference. However, due to the limitations of the Unity editor, we will not be implementing and automating any of these tests with the exception of smoke tests that we use to check dependencies. Instead, each tester will perform the manual tests from this document by stating their assumptions before the test and their results after the test. On top of this, we have began practicing more frequent manual acceptance testing as the developers work and recording the results of these on the appropriate wiki page. Lastly, we are following a more strict issue tracking procedure in that only a tester may close an issue once they have reviewed it and deem it closed.

## 2 Delta Changes

### 2.1 ID 2 Changes

Many tests in ID1 are no longer useful since architecture of our system changed as a result of how certain unity features behaved with the Oculus Rift. Tests that were removed are removed because they no longer fit the design of the system, and other tests are still relevant, but had to be updated to reflect newer methods. Many of the tests in the document this iteration are being left as //TODO for ID3 as they are not currently relevant to our system. Lastly, we updated the language of most every test to make it more clear on what we are testing and how we are testing it. Any other changes will be marked as such and given a description.

- Updated the Purpose description
- Added Delta Changes
- Added Testing Framework Issues
- Smoke Tests
  - Made verifying dependencies more detailed instead of just one line each
  - Deleted checking for the main menu
  - Check for registered default scene instead of workspace
  - Delete camera check because that is associated with the default scene check
  - Delete button alias check because that is associated with the Oculus libraries
- System Testing
  - Removed must/should/could have
  - Removed Initialize the program
  - Removed Quit from the menu
  - Removed load DICOM file
  - Moved exit workspace to acceptance testing
  - Updated select images to display
  - Updated remove images from display
  - Updated adjust contrast
  - Updated adjust brightness
  - Updated apply filter

- Updated zoom
- Updated move
- Added copy image
- Added load file from file browser
- Added next/prev image in display
- Integration Testing
  - Fixed grammar in introduction
  - Deleted everything else as a result of restructuring project. Will be in ID3 when we have pieces integrating.
- Regression
  - Added issues as regression tests
- Removed coverage testing for this iteration
- User Testing
  - Grammar
  - post -ask them after to see how they feel
  - look for studies and better questions
  - read Oculus warnings to them
- Coverage testing
  - Deleted for this iteration
- Acceptance Testing
  - Added methods for internal acceptance testing
- Testing Matrix
  - Deleted for this iteration

## 2.2 ID 3 Changes

- System Testing
  - Changed description and added test cases to Copy Image
  - Changed name of remove image to Remove copy
  - Updated description and added test cases to Remove copy
  - Changed description and added test cases to Load from file browser
  - Changed description and added test cases to select image in tray
  - Changed description and added test cases to remove image from tray
  - Deleted remove all images from tray
  - Changed description and added test cases to Next/prev Image in display
  - Updated description and added test cases to Change Contrast
  - Updated description and added test cases to Change brightness
  - Deleted move Copy tests
  - Updated description and added test cases to Change filter
  - Updated description and added test cases to Change zoom
- Integration Testing
  - Added all new tests to adhere to our new architecture

- Test cases still need to be written
- Regression Testing
  - Added 2 regression tests from tracked bug reports.
  - Test cases still need to be written
- Coverage Testing
  - Re-added Coverage Testing section for this ID
- Testing Matrix
  - Re-added Testing matrix section

## 2.3 ID 4 Changes

- Updated introduction for ID4 changes
  - Also added descriptions for how our testing file structure works
- Added “Testing Shift” section
- Added issue #8 to “Testing Framework Issues”
- Updated System tests for the new UML and implementation
- Updated Integration tests for the new UML and implementation
- Added regression tests for resolved issues
- Overhauled the User testing with best practices and moved the user script to a separate section
- Updated Coverage Testing with new information on the matter
- Added a description for the future implementation of a testing matrix

## 2.4 ID 5 Changes

- Edited the introduction to reflect ID5 goals
- Moved testing framework issues to a separate document with the coverage testing called Testing\_Issues\_With\_Unity
- System testing was overhauled to reflect the new UML and how we are manually testing instead of automating
- Integration tests were deleted because we cannot test the integration of components in a separate environment, thus every test we do needs to be considered a system test
- Acceptance tests were updated for the new requirements
- TESTING MATRIX WAS ADDED!!

# 3 Testing Shift

During the project, our methods of testing have had to shift from automated testing to more manual testing as a result of our chosen frameworks being Unity and Visual Studio. In

summary, the Unity Editor has very strict permissions that do not allow outside sources to access information that is crucial for us to be able to make interact objects with each other that do not exist in a scene. This limits our ability to do any amount of System or Integration testing. Among other issues, this also means that we have no ways of coverage testing our code, because of security exceptions, which trigger when trying to call certain methods within Unity in debug mode. Lastly, Unity lacks tools to properly mock objects which we have overcome by creating our own mocking methods. These issues are described in detail in the `Testing_Issues_With_Unity` document as well as a copy of the message that we have send to Unity and any responses that we receive from them if we do before the end of the project.

## 4 Smoke Testing

These tests are aimed at ensuring the most basic functionality of the system. They are the first set of tests that will be run during a cloud build so that if they fail, the build fails early before it begins running more exhaustive testing. Smoke Tests Are our only tests that are automated as

### 4.1 Verify all dependencies (libraries, imports, etc)

Test 1: Check that OVR asset exists in the project

- Obtain the absolute path to the directory “Luxonic Project” by calling the `Directory.GetCurrentDirectory ()` function.
- Use the `Path.Combine()` function to add the path “Assets/OVR” to the absolute path.
- Assert that the directory exists (`Directory.Exists()`) and that it contains files (`Directory.GetDirectories(targetPath).Length > 0`) and it contains directories (`(Directory.GetFiles(targetPath).Length > 0)`).

Test 2: Check that OVR Avatar asset exists in the project

- Obtain the absolute path to the directory “Luxonic Project” by calling the `Directory.GetCurrentDirectory ()` function.
- Use the `Path.Combine()` function to add the path “Assets/OvrAvatar” to the absolute path.
- Assert that the directory exists (`Directory.Exists()`) and that it contains files (`Directory.GetDirectories(targetPath).Length > 0`) and it contains directories (`(Directory.GetFiles(targetPath).Length > 0)`).

Test 3: Check that OVR Avatar Settings exists in the project

- Obtain the absolute path to the directory “Luxonic Project” by calling the `Directory.GetCurrentDirectory ()` function.

- Use the Path.Combine() function to add the path “Assets/Resources” to the absolute path.
- Assert that the directory exists (Directory.Exists()) and that it contains files (Directory.GetDirectories(targetPath).Length > 0)).

Test 4: Check that OVR Gamepad Bundle Plugin exists in the project

- Obtain the absolute path to the directory “Luxonic Project” by calling the Directory.GetCurrentDirectory () function.
- Use the Path.Combine() function to add the path “Assets/Plugins/OVRGamepad.bundle/Contents” to the absolute path.
- Assert that the directory exists (Directory.Exists()) and that it contains files (Directory.GetDirectories(targetPath).Length > 0)) and it contains directories ((Directory.GetFiles(targetPath).Length > 0)).

Test 5: Check that ImageEffects standard asset exists in the project

- Obtain the absolute path to the directory “Luxonic Project” by calling the Directory.GetCurrentDirectory () function.
- Use the Path.Combine() function to add the path “Assets/Standard Assets/Editor/ImageEffects” to the absolute path.
- Assert that the directory exists (Directory.Exists()) and that it contains files (Directory.GetDirectories(targetPath).Length > 0)).

Test 6: Check that Effects standard asset exists in the project

- Obtain the absolute path to the directory “Luxonic Project” by calling the Directory.GetCurrentDirectory () function.
- Use the Path.Combine() function to add the path “Assets/Standard Assets/Effects” to the absolute path.
- Assert that the directory exists (Directory.Exists()) and that it contains files (Directory.GetDirectories(targetPath).Length > 0)) and it contains directories ((Directory.GetFiles(targetPath).Length > 0)).

\*NOTE: we are not checking for any testing framework dependencies as they are not critical to the system.

## 4.2 User gets into the default scene

Test1: Check that a default scene is present in the build

- Obtain the default scene from the list “Scenes in Build” in Unity by calling the EditorSceneManager.GetSceneByBuildIndex(0) function.
- Assert that returned scene isValid().



## 5 Manual System Testing

Every test that we are performing manually is considered a system test because we cannot isolate components to integrate them in an environment that is just that specific interaction.

### 5.1 Open File Browser

Pressing Load opens the file browser and everything is minimized

Test 1: There are no images loaded in the tray/display

**Setup:** No setup required

**Actions:** Press the load button

**Expected Results:** The file browser is displayed

**Result:** The file browser was displayed

Test 2: There is 1 image loaded in the tray/display

**Setup:** No setup required

**Actions:** Press the load button

**Expected Results:** The file browser is displayed

**Result:** The file browser was displayed

Test 3: There is a directory loaded in the tray/display

**Setup:** No setup required

**Actions:** Press the load button

**Expected Results:** The file browser is displayed

**Result:** The file browser was displayed

### 5.2 Close File Browser

Cancelling loading an image from the file browser closes the file browser and returns to workspace as it was

Test 1: There are no images loaded in the tray/display

**Setup:** Open the file browser

**Actions:** Press the cancel button

**Expected Results:** The Display and Tray are empty

**Result:** The Display and Tray were empty

## Test 2: There is 1 image loaded in the tray/display

**Setup:** Open the file browser when there is already one image loaded into the workspace

**Actions:** Press the cancel button

**Expected Results:** The Display and Tray are empty

**Result:** The Display and Tray were empty

## Test 3: There is a directory loaded in the tray/display

**Setup:** Open the file browser when there is already a directory loaded in the workspace

**Actions:** Press the cancel button

**Expected Results:** The Display and Tray are empty

**Result:** The Display and Tray were empty

# 5.3 Navigate through directories

## Test 1: Move to a different directory

**Setup:** Open the file browser

**Actions:** Press a directory

**Expected Results:** all directories and files shown belong to the selected directory

**Result:** All directories and files within the chosen directory is now shown in the file browser

## Test 2: Move through multiple directories in an instance

**Setup:** Open the file browser

**Actions:** Press multiple directories

**Expected Results:** All Directories and files shown belong to the last pressed directory

**Result:** All directories and files within the last chosen directory is now shown in the file browser

## Test 3: Move back to previous directory

**Setup:** Open the file browser, if not in a directory that has a parent directory, navigate to one

**Actions:** Press the back button

**Expected Results:** All directories and files shown belong to the parent directory

**Result:** All directories and files that were shown belonged to the parent directory

## Test 4: Move back past the root directory

**Setup:** Navigate to a non-root directory

**Actions:** Press back until no longer possible

**Expected Results:** User should not be able to go back past the root directory

**Result:** Root directory is shown when trying to move back past the root directory

## 5.4 Load DICOM Image from File Browser

Image(s) that are loaded are displayed once on the tray and once on the display. Images should be well sized and spaced.

Test 1: There are no images loaded in the tray/display

**Setup:** Open the file browser with an empty workspace

**Actions:** Press a file

**Expected Results:** One image appears in the display and tray

**Result:** The chosen file was the only one shown in the display and tray

Test 2: There is 1 image loaded in the tray/display

**Setup:** Open the file browser

**Actions:** Press a file

**Expected Results:** A file is loaded into the workspace and is shown in the display and tray

**Result:** The selected file was loaded into the workspace and shown in the display and tray

Test 3: There is a directory loaded in the tray/display

**Setup:** Open the file browser

**Actions:** Navigate to the directory to be loaded and press load directory

**Expected Results:** All DICOM files in the chosen directory is loaded into the workspace and shown in the display and tray

**Result:** All DICOM files that were in the chosen directory were loaded into the workspace and shown in the display and tray

## 5.5 Create a Copy from the Tray

Testing to ensure that image copies are created properly.

Test 1: Test that the copy exists

**Setup:** Tray needs to have at least one image loaded

**Actions:** Press the desired image in the tray

**Expected Results:** A copy should be added into the workspace

**Result:** A copy has been added to the workspace

Test 2: Test that the copy is the same image in the Display and Tray

**Setup:** At Least one image in tray and display

**Actions:** Create copy of image by pressing it in the tray, visually verify that its the required image

**Expected Results:** Copy image is the same as one pressed

**Result:** Copy image is the same as one pressed

Test 3: Test that the copy is a different object from the one in the Display and Tray

**Setup:** At Least one image loaded in tray and display

**Actions:** Press desired image in tray

**Expected Results:** New game object added in the hierarchy

**Result:** New game object added in the hierarchy

## 5.6 Move Copy

Moving a copy should only ever change its transform and can only move one image per hand.

Test 1: Copy can be grabbed and moved to a different transform

**Setup:** Create a copy

**Actions:** Grab the image and move hand to another location and drop the copy

**Expected Results:** The copy is now located where the hand was last located with the grabbing motion

**Result:** The copy was moved to where the hand moved and dropped the copy

Test 2: Copies can be grabbed in each hand and moved to a different transform

**Setup:** Create a copy

**Actions:** Move the copy using each hand

**Expected Results:** The location of the copy should be moved by either hand

**Result:** The copy was moved to a different location using each hand

Test 3: Copy cannot be grabbed with a second hand when already being grabbed

**Setup:** Create a copy and grab the copy

**Actions:** Grab the grabbed copy with the other hand

**Expected Results:** The grabbed copy remains in the first hand that grabbed the copy

**Result:** Grabbing the copy with the second hand does not work and the copy remains in the first hand

Test 4: Only one copy can be grabbed per hand

**Setup:** Stack a bunch of copies on the same transform

**Actions:** Try to grab the entire stack of images

**Expected Results:** Only one image is grabbed

**Result:** Only one image was grabbed

## 5.7 Resize Copy

### Test 1: One copy is selected

**Setup:** At Least one copy exists in workspace

**Actions:** Select resize after selecting the image. Move right thumbstick

**Expected Results:** Image size changes

**Result** Image gets bigger and smaller

### Test 2: More than one copy is selected

**Setup:** At least two copies exist in workspace

**Actions:** Select resize after selecting at least two of the images. Move right thumbstick

**Expected Results:** Images size changes simultaneously

**Result** Images get bigger and smaller simultaneously

### Test 3: All copies in the workspace are selected

**Setup:** At least two copies exist in workspace

**Actions:** Select resize after selecting all the images. Move right thumbstick

**Expected Results:** Images size changes simultaneously

**Result** Images get bigger and smaller simultaneously

### Test 4: No copies are selected

**Setup:** At least two copies exist in workspace

**Actions:** Select resize after deselecting all the images. Move right thumbstick

**Expected Results:** Images size do not change

**Result** Images don't change in size

### Test 5: Copy has been selected then deselected

**Setup:** At least two copies exist in workspace and are selected

**Actions:** Select resize after deselecting all the images. Move right thumbstick

**Expected Results:** Images size do not change

**Result** Images don't change in size

### Test 6: Copy has a minimum size

**Setup:** At least one copy exists in workspace

**Actions:** Select resize after selecting all the images. Move right thumbstick to the left

**Expected Results:** Images get smaller and stop at a fixed size

**Result** Images do not get smaller than specified size

## Test 7: Copy has a maximum size

**Setup:** At least one copy exists in workspace

**Actions:** Select resize after selecting all the images. Move right thumbstick to the right

**Expected Results:** Images get larger and stop at a fixed size

**Result** Images do not get larger than specified size

## 5.8 Change Copy Brightness

### Test 1: One copy is selected

**Setup:** Create at least one copy and select an image

**Actions:** Press the brightness button and use the right analog stick to adjust the brightness (left decreases, right increases)

**Expected Results:** Brightness value of the copy has been changed

**Result:** The brightness of the copy decreases as the analog stick is held to the left and increases to the right

### Test 2: More than one copy is selected

**Setup:** Create more than one copy and select more than one image

**Actions:** Press the brightness button and use the right analog stick to adjust the brightness (left decreases, right increases)

**Expected Results:** Brightness value of the copy has been changed

**Result:** The brightness of the copy decreases as the analog stick is held to the left and increases to the right

### Test 3: all copies in workspace are selected

**Setup:** Create copies and select all images

**Actions:** Press the brightness button and use the right analog stick to adjust the brightness (left decreases, right increases)

**Expected Results:** Brightness value of the copy has been changed

**Result:** The brightness of the copy decreases as the analog stick is held to the left and increases to the right

### Test 4: no copies are selected

**Setup:** Create at least one copy and do not select any images

**Actions:** Press the brightness button and use the right analog stick to adjust the brightness (left decreases, right increases)

**Expected Results:** No copies have been adjusted

**Result:** No copies were adjusted

### Test 5: Copy has been selected then deselected

**Setup:** Create at least one copy, select it, and then deselect it

**Actions:** Press the brightness button and use the right analog stick to adjust the brightness (left decreases, right increases)

**Expected Results:** No copies have been adjusted

**Result:** No copies were adjusted

### Test 6: Copy has a minimum brightness

**Setup:** Create a copy and select it

**Actions:** Press the brightness button and hold the analog stick to the left

**Expected Results:** When the brightness value reaches zero in the inspector, it does not go any further

**Result:** The brightness does not go below zero (some room for leeway)

### Test 7: Copy has a maximum brightness

**Setup:** Create a copy and select it

**Actions:** Press the brightness button and hold the analog stick to the right

**Expected Results:** When the brightness value reaches zero in the inspector, it does not go any further

**Result:** The brightness does not go above 2 (some room for leeway)

## 5.9 Change Copy Contrast

### Test 1: One copy is selected

**Setup:** Create at least one copy and select an image

**Actions:** Press the contrast button and use the right analog stick to adjust the contrast (left decreases, right increases)

**Expected Results:** Contrast value of the copy has been changed

**Result:** The contrast of the copy decreases as the analog stick is held to the left and increases to the right

### Test 2: More than one copy is selected

**Setup:** Create more than one copy and select more than one image

**Actions:** Press the contrast button and use the right analog stick to adjust the contrast (left decreases, right increases)

**Expected Results:** Contrast value of the copy has been changed

**Result:** The contrast of the copy decreases as the analog stick is held to the left and increases to the right

### Test 3: All copies in workspace are selected

**Setup:** Create copies and select all images

**Actions:** Press the contrast button and use the right analog stick to adjust the contrast (left decreases, right increases)

**Expected Results:** Contrast value of the copy has been changed

**Result:** The contrast of the copy decreases as the analog stick is held to the left and increases to the right

### Test 4: No copies are selected

**Setup:** Create at least one copy and do not select any images

**Actions:** Press the contrast button and use the right analog stick to adjust the contrast (left decreases, right increases)

**Expected Results:** No copies have been adjusted

**Result:** No copies were adjusted

### Test 5: Copy has been selected then deselected

**Setup:** Create at least one copy, select it, and then deselect it

**Actions:** Press the contrast button and use the right analog stick to adjust the contrast (left decreases, right increases)

**Expected Results:** No copies have been adjusted

**Result:** No copies were adjusted

### Test 6: Copy has a minimum contrast

**Setup:** Create a copy and select it

**Actions:** Press the contrast button and hold the analog stick to the left

**Expected Results:** When the contrast value reaches zero in the inspector, it does not go any further

**Result:** The contrast does not go below zero (some room for leeway)

### Test 7: Copy has a maximum contrast

**Setup:** Create a copy and select it

**Actions:** Press the contrast button and hold the analog stick to the right

**Expected Results:** When the contrast value reaches zero in the inspector, it does not go any further

**Result:** The contrast does not go above 2 (some room for leeway)



## 5.10 Invert Copy Colours

### Test 1: One copy is selected

**Setup:** Copy exists and is selected

**Actions:** Press invert button until press animation appears

**Expected Results:** Image colors invert

**Result:** Image colors invert

### Test 2: More than one copy is selected

**Setup:** At least two copies exist and are selected

**Actions:** Press invert button until press animation appears

**Expected Results:** Image colors invert

**Result:** Image colors invert

### Test 3: More than one copy is selected with different states of inversion

**Setup:** At least two copies exist and are selected. Some but not all images are inverted

**Actions:** Press invert button until press animation appears

**Expected Results:** Images which are not inverted invert, opposite happens for inverted images

**Result:** Images which are not inverted invert, opposite happens for inverted images

### Test 4: All copies in workspace are selected

**Setup:** At least two copies exist and are all selected

**Actions:** Press invert button until press animation appears

**Expected Results:** All image colors invert

**Result:** All image colors invert

### Test 5: No copies are selected

**Setup:** At least two copies exist and are all deselected

**Actions:** Press invert button until press animation appears

**Expected Results:** All image colors remain the same

**Result:** All image colors remain the same

### Test 6: Copy has been selected then deselected

**Setup:** At least two copies exist and are all selected

**Actions:** Press invert button until press animation appears after deselect all has been pressed

**Expected Results:** All image colors remain the same

**Result:** All image colors remain the same

## 5.11 Change copy saturation

### Test 1: One copy is selected

**Setup:** Create at least one copy and select an image

**Actions:** Press the saturation button and use the right analog stick to adjust the contrast (left decreases, right increases)

**Expected Results:** Saturation value of the copy has been changed

**Result:** The saturation of the copy decreases as the analog stick is held to the left and increases to the right

### Test 2: More than one copy is selected

**Setup:** Create at least two copies and select at least two copies

**Actions:** Press the saturation button and use the right analog stick to adjust the contrast (left decreases, right increases)

**Expected Results:** Saturation value of the copies has been changed

**Result:** The saturation of the copies decreases as the analog stick is held to the left and increases to the right

### Test 3: All copies in workspace are selected

**Setup:** Create at least two copies and select them all

**Actions:** Press the saturation button and use the right analog stick to adjust the contrast (left decreases, right increases)

**Expected Results:** Saturation value of the copies has been changed

**Result:** The saturation of the copies decreases as the analog stick is held to the left and increases to the right

### Test 4: No copies are selected

**Setup:** Create at least one copy and do not select any copies

**Actions:** Press the saturation button and use the right analog stick to adjust the contrast (left decreases, right increases)

**Expected Results:** Saturation value of the copy has not been changed

**Result:** The saturation of the copy does not change

### Test 5: Copy has been selected then deselected

**Setup:** Create at least one copy and select an image

**Actions:** Press the saturation button and use the right analog stick to adjust the contrast (left decreases, right increases) after images have been deselected

**Expected Results:** Saturation value of the copy has not been changed

**Result:** The saturation of the copy does not change

### Test 6: Copy has a minimum saturation

**Setup:** Create at least one copy and select an image

**Actions:** Press the saturation button and move the right analog stick to the left

**Expected Results:** Saturation value of the copy reduces and stops after a certain point

**Result:** The saturation of the copy decreases and stops at a min value

### Test 7: Copy has a maximum saturation

**Setup:** Create at least one copy and select an image

**Actions:** Press the saturation button and move the right analog stick to the right

**Expected Results:** Saturation value of the copy increases and stops after a certain point

**Result:** The saturation of the copy increases and stops at a min value

## 5.12 Select/Deselect image

### Test 1: Image can be selected

**Setup:** Create a copy

**Actions:** Press a copy

**Expected Results:** Copy pressed should be selected, indicated with a blue border

**Result:** Copy pressed was selected and had a blue border

### Test 2: Image can be deselected

**Setup:** Create a copy, select the copy

**Actions:** Press the copy

**Expected Results:** Copy pressed should be deselected, indicated with no blue border

**Result:** Copy pressed was deselected and did not have a blue border

### Test 3: Image can be selected again after a deselect

**Setup:** Create a copy, select and then deselect copy

**Actions:** Press the copy

**Expected Results:** Copy pressed should be selected, indicated with a blue border

**Result:** Copy pressed was selected and had a blue border

### Test 4: Multiple images can be selected at once

**Setup:** Create multiple copies

**Actions:** Press multiple copies

**Expected Results:** Copies pressed should be selected, indicated with a blue border

**Result:** Copies pressed were selected and had a blue border

#### Test 5: Multiple images can be deselected a tonce

**Setup:** Create multiple copies, select the copies

**Actions:** Press the copies

**Expected Results:** Copies pressed should be deselected, indicated with no blue border

**Result:** Copies pressed were deselected and did not have a blue border

### 5.13 Select/Deselect all images

#### Test 1: Select all does nothing when there are no images

**Setup:** No copies loaded into the workspace

**Actions:** Press select all button

**Expected Results:** Nothing happens

**Result:** Nothing happens

#### Test 2: Deselect all does nothing when there are no images

**Setup:** No copies loaded into the workspace

**Actions:** Press deselect all button

**Expected Results:** Nothing happens

**Result:** Nothing happens

#### Test 3: Select all selects one image when only one image exists that is deselected

**Setup:** One copy loaded into workspace. Copy is not selected

**Actions:** Press select all button

**Expected Results:** Image gets selected

**Result:** Image is selected

#### Test 4: Select all does nothing when only one image exists and it is selected

**Setup:** One copy loaded into workspace. Copy is selected

**Actions:** Press select all button

**Expected Results:** Image remains selected

**Result:** Image remains selected

Test 5: Deselect all deselects one image when only one exists that is selected

**Setup:** One copy loaded into workspace. Copy is selected

**Actions:** Press deselect all button

**Expected Results:** Image gets deselected

**Result:** Image is deselected

Test 6: Deselect all does nothing when only one image exists and it is deselected

**Setup:** One copy loaded into workspace. Copy is not selected

**Actions:** Press deselect all button

**Expected Results:** Image remains deselected

**Result:** Image remains deselected

Test 7: Select all selects all images when multiple images exists that are deselected

**Setup:** Load multiple copies into workspace. Make sure copies are deselected

**Actions:** Press select all button

**Expected Results:** All images are selected

**Result:** All the images are selected

Test 8: Select all selects all images when multiple images exists that are in varying states of selected and deselected

**Setup:** Load multiple copies into workspace. Make sure copies are deselected and some are selected

**Actions:** Press select all button

**Expected Results:** All images are selected

**Result:** All the images are selected

Test 9: Deselect all deselects all images when multiple exist that are selected

**Setup:** Load multiple copies into workspace. Make sure copies are deselected

**Actions:** Press deselect all button

**Expected Results:** All images are deselected

**Result:** All the images are deselected

Test 10: Deselect all deselects all images when multiple exist that are in varying states of selected and deselected

**Setup:** Load multiple copies into workspace. Make sure copies are deselected and some are selected

**Actions:** Press deselect all button

**Expected Results:** All images are deselected

**Result:** All the images are deselected

## 5.14 Close a Copy

Test 1: Test that copy is no longer in the workspace and that the original exists in the tray, display, and file system

**Setup:** Create a copy and select it

**Actions:** Press the close button

**Expected Results:** Copy closed is no longer in the workspace but still remains in the display and tray

**Result:** Copy that was closed is no longer in the workspace but remained in the display and tray

## 5.15 Restore Copy to Default

Test 1: Copy is at default and restore does nothing

**Setup:** Create a copy and make no adjustments, select adjusted copy

**Actions:** Press the restore button

**Expected Results:** Copy should be in the original state

**Result:** Copy remains in the default state

Test 2: Image's brightness, contrast, saturation, and inversion are changed and restore sets it to default

**Setup:** Create a copy and adjust multiple settings of the copy, select the adjusted copy

**Actions:** Press the restore button

**Expected Results:** Copy should be in the original state

**Result:** Copy restored reverted back to default state

Test 3: Multiple images' brightness, contrast, saturation, and inversion are changed and restore sets them to default

**Setup:** Create multiple copies and adjust multiple settings of the copies, select all adjusted

**Actions:** Press the restore button

**Expected Results:** Copies should be in the original state

**Result:** Copies restored reverted back to default state

Test 4: Multiple images' are in varying states of select with brightness, contrast, saturation, and inversion all changed and restore sets the selected to default.

**Setup:** Create multiple copies and adjust multiple settings of the copies to varied states, select all adjusted

**Actions:** Press the restore button

**Expected Results:** Copies should be in the original state

**Result:** Copies restored reverted back to default state

Test 5: There are no images and the button does nothing

**Setup:** Create a workspace with no copies

**Actions:** Press the restore

**Expected Results:** Nothing has changed

**Result:** Nothing changed

## 5.16 Quit the Workspace

Test 1: Press the quit button

**Setup:** User is in the workspace

**Actions:** Press the quit button

**Expected Results:** The application is closed

**Result:** The application closed

## 5.17 Minimize Display/Dashboard

Test 1: Minimize

**Setup:** User is in workspace

**Actions:** Select minimize button

**Expected Results:** Dashboard minimizes, minimize button renamed to maximize, and display/dashboard is no longer visible

**Result:** Dashboard minimizes and minimize button renamed to maximize, buttons are no longer visible

## 5.18 Maximize Display/Dashboard

### Test 1: Maximize

**Setup:** User is in the workspace and the dashboard/display is minimized

**Actions:** Press the maximize button

**Expected Results:** The dashboard and display should reappear and the maximize button is renamed to minimize

**Result:** Maximize button is renamed to minimize and buttons reappear

## 5.19 Move Left and Right Through Images in Display

Test 1: Test that going to the left image in the display updates the display Images order and the selected thumbnail in the tray.

**Setup:** Images loaded into display and tray (At Least 4 images need to be loaded)

**Actions:** Click the left button

**Expected Results:** Images in display shift one position to the left. Visually check that the new image in the display was the same as the tray

**Result:** Images in display shift one position to the left.

Test 2: Test that going to the right image in the display updates the display Images order and the selected thumbnail in the tray

**Setup:** Images loaded into display and tray (At Least 4 images need to be loaded)

**Actions:** Click the right button

**Expected Results:** Images in display shift one position to the right. Visually check that the new image in the display was the same as the tray

**Result:** Images in display shift one position to the right.

Test 3: Test that pressing left does loops when you are at the start of the tray

**Setup:** Images loaded into display and tray (At Least 4 images need to be loaded). Make sure that images in the display are the first three images in tray

**Actions:** Click the left button

**Expected Results:** Images in display shift one position to the left. Visually check that the new image in the display was the same as the last image in the tray

**Result:** Left image in display is the last image in the tray



Test 4: Test that pressing right loops to the beginning when you are at the end of the tray

**Setup:** Images loaded into display and tray (At Least 4 images need to be loaded). Make sure that images in the display are the last three images in tray

**Actions:** Click the right button

**Expected Results:** Images in display shift one position to the right. Visually check that the new image in the display was the same as the first image in the tray

**Result:** Right image in display is the first image in the tray

## 5.20 Button Presses/Depresses

Test 1: Button presses when pressed and no copy is selected

**Setup:** User is in the workspace and there are no copies selected

**Actions:** Press any button

**Expected Results:** The button should press in and depress

**Result:** The button presses and depresses

Test 2: Button presses when pressed and stays pressed until another button is pressed when a copy is selected

**Setup:** User is in the workspace and a copy is selected

**Actions:** Press any button

**Expected Results:** The button should stay pressed until another button is pressed

**Result:** The button stays pressed until another button is pressed

## 6 Manual Regression testing

As a result of us running manual testing, we will not be recording our regression tests in this document. Instead, the testers have taken a more active role in the issue tracking process and each issue that gets closed has a description of what tests were performed to ensure that it was fixed. If we wish to Manually perform regression tests, then we just have to look at our github issue log and follow the manual steps that we took the first time that we solved each issue.

## 7 User testing

User testing will require pre-test and post-test documentation to ensure the user understands the purpose of their testing of our software. All questions will be optional for the user to answer. These questions are formatted and explained in the "User Testing Form". Users are ideally

people without experience in computer science as we are ideally looking for user interaction glitches, not testing for bugs, though we will still get some computer science users to test for us to add robustness to our tests. For radiologists, we will come up with session specific questions that related to our programs current state to help steer the direction of the program according to their feedback

- Before The users show up the lab should be properly set up with the equipment ready to go and at least 2 team members should be present for the test.
- When the user arrives, explain to them our project using the following key points:
  - It is a Virtual reality application
  - It is meant to be used by radiologists
  - The purpose is for the viewing of DICOM images
  - The purpose of them being here testing is to help us ensure that our program flows smoothly when it is being used by somebody who is unfamiliar with it
- Take the user's Pre documentation as per the 'User Testing Form'
- Inform the user that:
  - they must use hand sanitizer that is supplied
  - they must accept assistance in taking the headset and controllers on and off
  - We will want their first impressions when the system starts up
  - We want them to walk us through their actions and what they think is going to happen before and after they do something
  - Let them know that if at anytime they need out of the headset to say so and the tester will come help them out
- During the testing the tester is to:
  - Ask the user questions
  - Navigate the user to missed material once the user thinks that they are done
  - Take notes on their responses and comments
- Take the users post documentation
- Show the user out
- Discuss the Results: and take notes
- Clean the headset and controllers
- The results of the user tests will live in text files in our repository
- After each series of user testing, the results will be in a separate document
  - Currently they are located in Medical\_Professional\_User\_Testing\_Results and User\_Testing\_Form

## 8 Acceptance Testing

Manual tests to be done by the appropriate member of our team in the following cases as a result of us not having automated tests:

- By developers after working on an affected feature.
- By testers after a merge of a feature branch into the development branch

- By testers after a development freeze
- By the Project Manager just prior to turning in a deliverable build

Record of the test being performed, and its results, will be recorded in the project wiki.

## 8.1 Quit Workspace

- Ensure that clicking on the “Quit” button closes the workspace at anytime.

## 8.2 Move Copy

- Create a copy
- Grab the copy
- Move the copy
- Let go of the copy

## 8.3 Show/Hide Dashboard

Test that when the “Show Dashboard” button is pressed that the dashboard is displayed with all elements which include:

- All button except for “Maximize”
- The Tray with any images it had on it still loaded and highlighted
- The Display with any images it had still on it that match the highlighted images in the Tray

Test that when “Minimize” button is pressed that all of the above elements are hidden and the only element that remains is the “Show Dashboard” button

## 8.4 Create Copy

- Load an image into the tray
- Poke the image
- A copy should be placed in the workspace

## 8.5 Close Copy

- Select a copy
- Press the close button
- The copy should no longer exist

## 8.6 Show File Browser

- Press the load button
- The File Browser should display

## 8.7 Close File Browser

- Open the File Browser
- Press the “cancel” button
- The File Browser should no longer be displayed

## 8.8 Load from File Browser

- Open the File Browser
- Poke a DICOM formatted file
- The file should appear in the tray and display

## 8.9 Remove from tray - Not implemented

- Select image in the display
- Select close
- Image should no longer appear in the tray or display

## 8.10 Remove All From Tray - not Implemented

- Select image in the display
- Select close all
- Images should no longer appear in the tray or display

## 8.11 Slide Through Images in Display - not implemented

- Grab the slider
- Move it back and forth to browse through images in the tray.
- Ensure that the tray’s selection indicator is updated to reflect this.

## 8.12 Next/Previous Image in Display

- Load images until the left and right buttons appear
- Press left or right
- Check that the images in the display properly changed according to the display

## 8.13 Change Brightness

- Select a copy
- Poke the brightness button
- Use the thumbstick to change brightness

## 8.14 Change Contrast

- Select a copy
- Poke the contrast button
- Use the thumbstick to change contrast

## 8.15 Change Saturation

- Select a copy
- Poke the saturation button
- Use the thumbstick to change Saturation

## 8.16 Resize Image

- Select a copy
- Poke the resize button
- Use the thumbstick to change Copy size

## 8.17 Select/Deselect copies

- Poke a copy to see that it selects
- Poke the copy again to see it deselect

## 8.18 Buttons Press and Depress

- Poke a button to see it press
- Poke another button to see the previous button depress

## 8.19 Select All and Deselect All

- Load copies into the workspace
- Press select all and see that all copies are selected
- Press deselect all and see that all copies are deselected

## 8.20 Navigate through Directories

- Open the file browser
- Poke a directory to navigate into it
- Press back to navigate out of the directory

## 8.21 Invert Copy Colours

- Select a copy
- Poke the invert button
- Use the thumbstick to invert the colours

## 8.22 Restore Image

- Select an image
- Change a property
- Poke restore to change the changed property back to default

# 9 Testing Matrix

We have 2 testing matrices to keep it manageable. The first is for our acceptance tests and the second is for our Manual system tests. Both matrices can be found here on our wiki:

<https://wiki.usask.ca/display/C371T3/Testing+Matrix>

A few notable features about the matrices:

- Handling Dicom Files to parse image and patient data takes place in the majority of tests because you must navigate through this step to create a file to actually do anything in our system
- Minimize and maximize happen every time that the file browser is opened or closed, so they are frequented in the matrices
- We have tests for a few features that are not requirements, but we implemented them as they were very necessary usability features such as buttons pressing and depressing and the restore button