Verification and Validation Report: Mechtronics Enigeering

Team 32, Wingman, SmartVault
Edward He
Erping Zhang
Guangwei Tang
Peng Cui
Peihua Jin

March 8, 2023

1 Revision History

Date	Version	Notes
2023/3/7	1.0	Finish the required parts
2023/3/8	1.1	Fix errors

2 Purpose

This document is intended to support the systematic plan for testing the functionality of the system. It meant to show the system has met the requirements in both software and hardware aspects mentioned in requirements document. In particular, this document will describe the testing results. By the end of testing process, it can be shown that the system is working properly and available for usage.

3 Scope

The document would pay attention to the different functionalities being discussed within the VnVPlan documentation. In addition, it would undergo the testing processes as if it was a black box, which will emphasis on the inputs and outputs of the system instead of the internal process and mechanics.

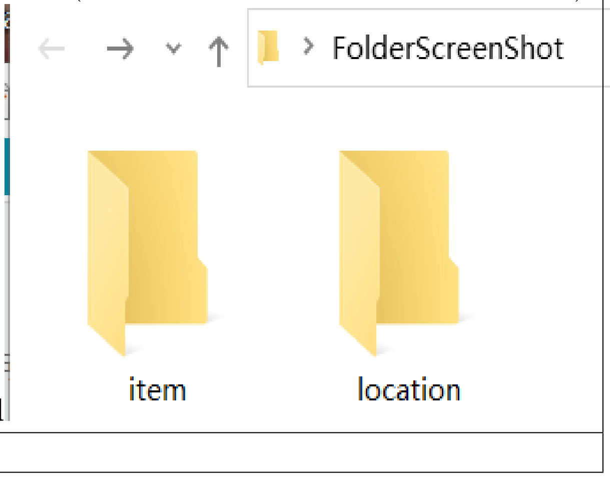
4 Background

SmartVault is designed to assist the user to remember where his/her belongings are and the most recent time the user had used or placed their belongings. The proposed system is capable of tracking and following human activities to position itself best for capturing any moving objects caused by the user. The system will identify each item that is being moved and record/update their corresponding positions. The user then has the ability to interact with our system through an interface and select which item the user is looking for. Given this information, our system would identify where that specific item is and assist the user to locate their belongings in a short time. This section will not be appropriate for every project.

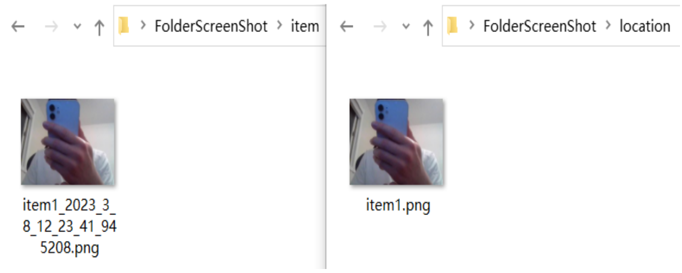
5 Functional Requirements Evaluation

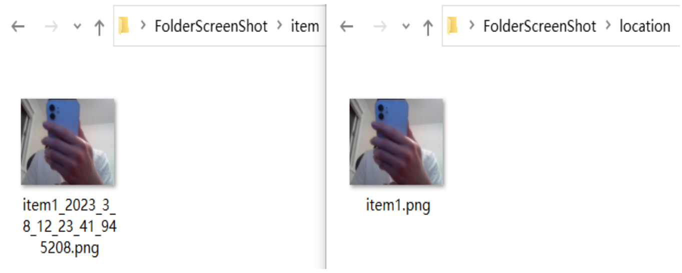
5.0.1 Area of Testing1

Manual Testing Testing shown:

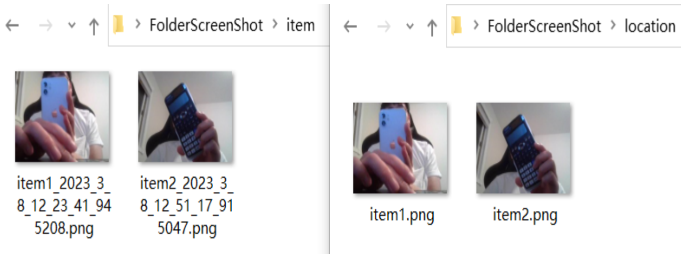
Test Number	IPR4-1
Requirement Reference	IPR4
Requirement	To create 3 folders sequentially
Input	createFolder() being called
Desired Output	3 folders (FolderScreenShot, item, location) created
Actual Output	3 folders (FolderScreenShot, item, location)  created
Conclusion	Pass

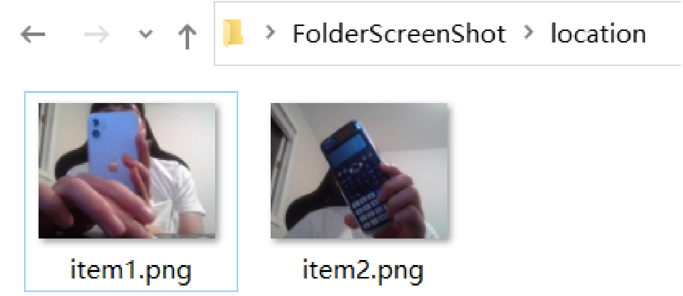
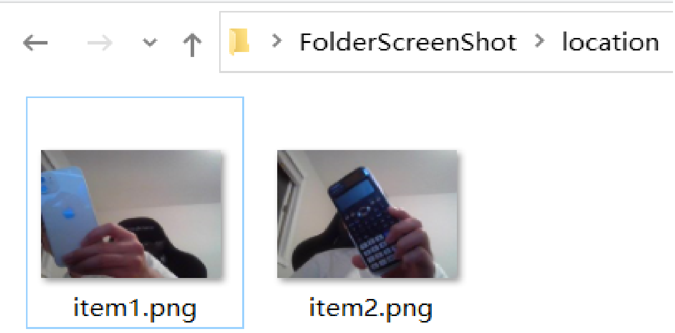
Test Number	IPR4-2
Requirement Reference	IPR4
Requirement	Do nothing if they have already existed
Input	createFolder() being called
Desired Output	No change
Actual Output	No change
Conclusion	Pass

Test Number	IPR5-1
Requirement Reference	IPR5
Requirement	To store the initial frame
Input	(1, ’i’)
Desired Output	Adding item1_{date and time}.png, item1.png
Actual Output	Added as:  FolderScreenShot › item item1_2023_3_8_12_23_41_945208.png FolderScreenShot › location item1.png
Conclusion	Pass

Test Number	IPR5-2
Requirement Reference	IPR5
Requirement	To check whether the frame is stored in the correct path
Input	(1, ’i’)
Desired Output	item{num}_{date and time}.png is stored in ’item’, item{num}.png is stored in ’location’
Actual Output	item1_2023_3_8_12_23_41_945208.png is within ’item’, item1.png is inside ’location’  FolderScreenShot › item item1_2023_3_8_12_23_41_945208.png FolderScreenShot › location item1.png
Conclusion	Pass

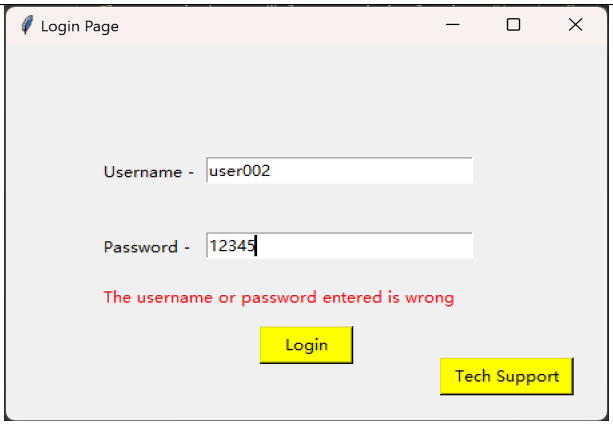
Automatic Testing Testing shown:

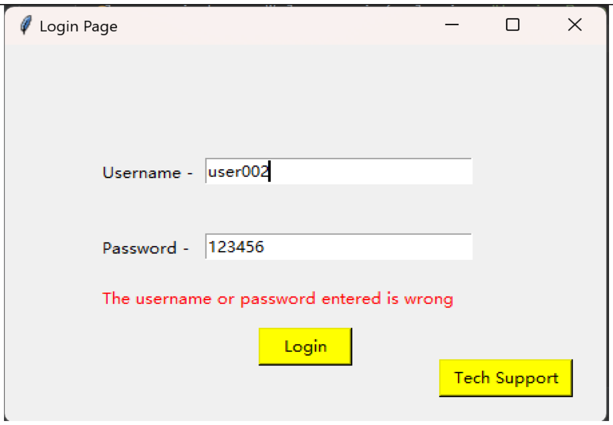
Test Number	IPR6-1
Requirement Reference	IPR5, IPR6
Requirement	To check whether the frame for the second item is captured
Input	(2, ’i’)
Desired Output	Adding item2_{date and time}.png, item2.png
Actual Output	Added as:  FolderScreenShot › item item1_2023_3_8_12_23_41_945208.png item2_2023_3_8_12_51_17_915047.png FolderScreenShot › location item1.png item2.png
Conclusion	pass

Test Number	IPR6-2
Requirement Reference	IPR4, IPR6
Requirement	To check whether the location frame for the first item is updated, meanwhile the second item won't get affected
Input	(1, 'u')
Desired Output	item1_{date and time}.png should remain, item1.png shall be updated
Actual Output	Only item1.png get updated Comparison shown:  
Conclusion	Pass

5.0.2 UI Interface Menu

Manual Testing Testing shown:

Test Number	UIR1-1
Requirement Reference	UIR1
Requirement	The UI should notify the user when the user has a wrong password input
Input	The wrong input of the password
Desired Output	There should be a text notification shown on the window
Actual Output	 Username - user002 Password - 12345 The username or password entered is wrong
Conclusion	The test is successful

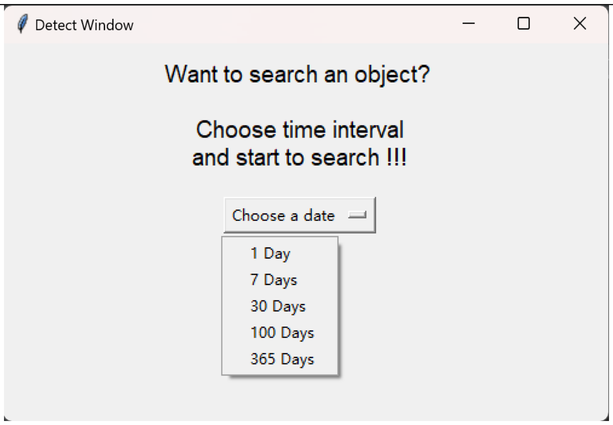
Test Number	UIR1-2
Requirement Reference	UIR1
Requirement	The UI should notify the user when the user has a wrong username input
Input	The wrong input of the username
Desired Output	There should be a text notification shown on the window
Actual Output	 Username - user002 Password - 123456 The username or password entered is wrong
Conclusion	The test is successful

Test Number	UIR2-1
Requirement Reference	UIR2
Requirement	The UI should be able to let the user to switch the pictures shown in the window
Input	The next button is clicked
Desired Output	A different picture is shown
Actual Output	A different picture is shown in the window
Conclusion	The test is successful

Test Number	UIR2-2
Requirement Reference	UIR2
Requirement	The UI should be able to let the user to switch the pictures shown in the window
Input	The previous button is clicked
Desired Output	A different picture is shown
Actual Output	A different picture is shown in the window
Conclusion	The test is successful

Test Number	UIR3-1
Requirement Reference	UIR3
Requirement	The UI should be able to provide information about the location of the item
Input	The user select the item picture
Desired Output	The location of the picture is shown in a new window
Actual Output	
Conclusion	The test is successful

Test Number	UIR3-2
Requirement Reference	UIR3
Requirement	The UI should be able to provide information about the location of the item
Input	The user select the item picture
Desired Output	The UI should notify the user that the item has been taken out of the room
Actual Output	
Conclusion	The test is successful

Test Number	UIR4-1
Requirement Reference	UIR4
Requirement	The UI should be able to let the user to choose the information input
Input	The user select the choose box
Desired Output	The UI provides choices to the user
Actual Output	 Want to search an object? Choose time interval and start to search !!! Detect Window Choose a date 1 Day 7 Days 30 Days 100 Days 365 Days
Conclusion	The test is successful

6 Nonfunctional Requirements Evaluation

6.1 Usability and Humanity Requirements

Test Number	APR1-1
Requirement Reference	APR1
Requirement	The User is able to launch the program without help
Input	The servy paper
Desired Output	An average of high rating shown on the paper
Actual Output	
Conclusion	The test is successful

Test Number	EUR1-1
Requirement Reference	EUR1
Requirement	The User is able to use the hardware without help
Input	The servy paper
Desired Output	An average of high rating shown on the paper
Actual Output	
Conclusion	The test is successful

Test Number	EUR2-1
Requirement Reference	EUR2
Requirement	The User is able to find the desired item without help
Input	The servy paper
Desired Output	An average of high rating shown on the paper
Actual Output	
Conclusion	The test is successful

Test Number	LER1-1
Requirement Reference	LER1
Requirement	The User is able to install the software without help
Input	The servy paper
Desired Output	An average of high rating shown on the paper
Actual Output	
Conclusion	The test is successful

Test Number	LER2-1
Requirement Reference	LER2
Requirement	The program can take pictures after the user has been leave the room
Input	The user leave the room
Desired Output	Pictures are taken
Actual Output	
Conclusion	The test is successful

Test Number	UPR1-1
Requirement Reference	UPR1
Requirement	The user is able to see each picture clearly
Input	The servy paper
Desired Output	An average of high rating shown on the paper
Actual Output	
Conclusion	The test is successful

7 Changes Due to Testing

8 Traceability Matrices

8.1 Traceability for Functional Requirements

Table 1: Traceability for Area of Testing 1

Test Method	Requirement	Test Number
Manual	IPR4	IPR4-1
Manual	IPR4	IPR4-2
Manual	IPR5	IPR5-1
Manual	IPR5	IPR5-2
Automatic	IPR5, IPR6	IPR6-1
Automatic	IPR4, IPR6	IPR6-2

Table 2: Traceability for UI Interface Menu

Test Method	Requirement	Test Number
Manual	UIR1	UIR1-1
Manual	UIR1	UIR1-2
Manual	UIR2	UIR2-1
Manual	UIR2	UIR2-2
Manual	UIR3	UIR3-1
Manual	UIR3	UIR3-2
Manual	UIR4	UIR4-1

8.2 Traceability for Nonfunctional Requirements

Table 3: Traceability for Usability and Humanity Requirements

Test Method	Requirement	Test Number
Manual	APR1	APR1-1
Manual	EUR1	EUR1-1
Manual	EUR2	EUR2-1
Manual	LER1	LER1-1
Manual	LER2	LER2-1
Manual	UPR1	UPR1-1
Manual	UIR4	UIR4-1

Appendix — Reflection

The information in this section will be used to evaluate the team members on the graduate attribute of Lifelong Learning. Please answer the following questions:

1.

2.