Acceptance Test Plan

Firmware Infirmary

Version 2.0

Defined By:

Billy Carroll, Jovan Kouakou, Mike Mariano, Nick Pelletier, Ryan Marcovecchio, Tom Card

Advisor:

Jeff Salvage

Stakeholder:

Drexel University

Table of Contents

1. Introduction

1.1. Purpose:

The idea of this Acceptance Test Plan is about having a plan for how to conduct the acceptance plan for the game Firmware Infirmary. This document follows a specific plan on how to test the game out and the expected outcomes of the playthrough tests along with the functionalities of the game outlined in the Firmware Infirmary Software Requirements Specifications or SRS. This plan also outlines tasks and how to progress throughout the game.

1.2. References:

Many of these can be attributed to the Firmware Infirmary SRS Document which outlines the game processes and what happens during each game scenario. Most of the formatting was referenced from Waved, a previous Senior design project, as allowed by Jeffrey Salvage.

1.3. Issue Tracking:

The purpose of the issue tracking system is to make the player aware of game bugs, glitches, display issues, fail states of mini games, progressing through the game, and more.

2. Definitions

Acceptance Test: Tests on a project to ensure all of its requirements and expectations are met.

Bug: Any glitch or unintended results within a game.

Collider: A component in Unity that defines an object's physical shape and allows it to interact with other colliders in the game world.

Coroutine: A type of function in Unity that allows code to be executed over multiple frames, useful for animating or updating objects over time.

HUD (Heads-Up Display): A type of user interface that displays information to the player on top of the game screen, without obscuring the view of the game world.

Interactable: A term used in game development to describe an object that can be interacted with by the player, typically through the use of buttons or other input methods.

LineRender: A component in Unity that allows developers to draw lines and shapes on the screen, typically used for visual effects or debugging.

NPC (Non-Player Character): A character in a game that is controlled by the computer, rather than by the player

Player: The person that is playing the game.

Prefab: A reusable game object template in Unity, typically used for creating multiple instances of the same object with the same properties.

Raycast: A technique used in game development to detect if an object is hit by a virtual ray, typically used for line of sight or aim detection.

Rigidbody: A component in Unity that allows an object to be affected by physics and interact with other objects in the game world.

SRS (Software Requirements Specification): A document that outlines the functional and nonfunctional requirements for a software system.

Test: Runthrough to ensure the requirements within the game are being met.

Trigger: A type of collision detection in Unity that occurs when two colliders come into contact, typically used for triggering events or interactions.

UI (User Interface): The part of a software system that interacts with the user, allowing them to input commands and receive feedback.

VFX (Visual Effects): Images created to enhance and simulate effects to create an enhanced experience.

3. Test Items

Builds: Each build of the game is intended to make improvements to the game. However, making those adjustments can lead to further problems that may have deeper roots to it which should also be addressed for each weekly build.

Controls: Ensure all the controls whether they are default or configured to the player's preferences are working properly.

Effects: When playing the game, you should observe how the effects impact the gameplay and figure out if any lag could affect the gameplay.

Framing: How well does the game hold up in certain resolutions? Is it compatible with a majority or resolution? How bright do the graphics get in certain resolutions? These are the questions to ask regarding framing within each build.

Gameplay: The gameplay needs to be tested to figure out game aesthetics, bugs, glitches, graphics, etc. and make the necessary modifications for the next build.

3.1. What to Test

Aspects to test include:

- Controls
- Frames
- Gameplay

- Interactions
- Minigames
- Resolutions

3.2. What not to Test

Aspects to not test include:

- Unity base functionality
- Visual Processing (pre, post etc)
- Look of the game
- Static objects

4. Testing Participants

This section describes the roles and responsibilities of the parties involved in the Acceptance Test plan, as well as the procedure for reporting the test results and any subsequent issues.

4.1 Roles and Responsibilities

I. For this test plan these people were used in the specified roles:
    - Test Team Leader: Nick Pelletier
    - Testers: GDAP students, Billy Carroll
    - Stakeholder: Jeff Salvage

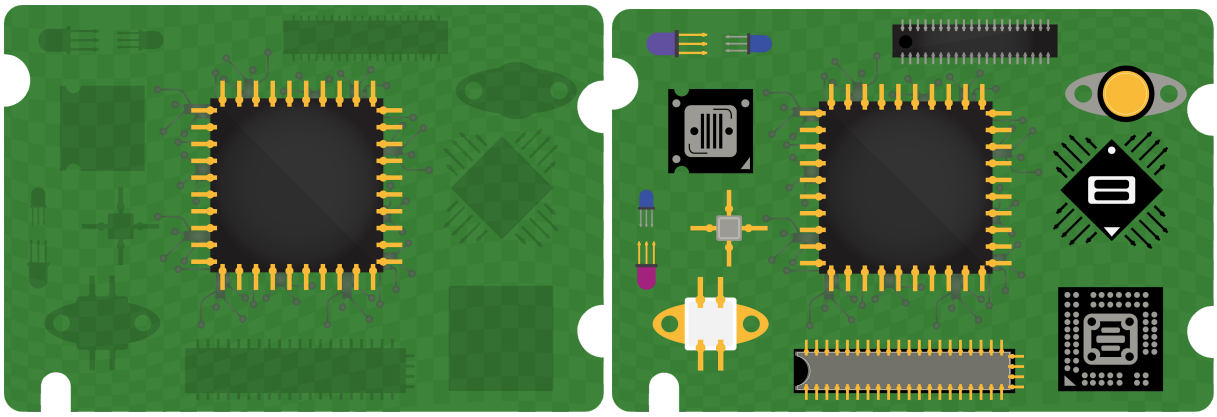
4.2 Training Requirements

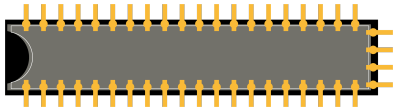
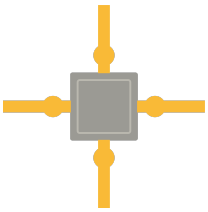
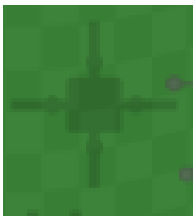
All parties involved should be familiar with the game Firmware Infirmary and how to play it. They should understand the tests and how to fulfill them. All reported issues should be viewed by the test team leader and producers in order to correctly send out work to fix said issues.

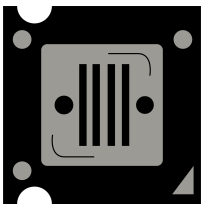
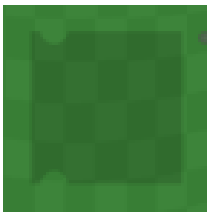
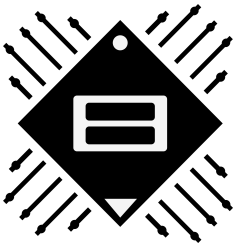
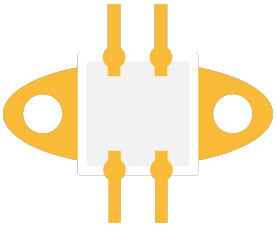
5. Activity Tables

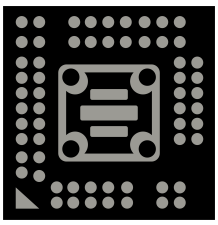
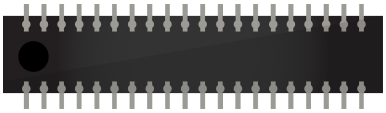
5.1 Repair Broken Activities

5.1.1 Repair Microchip Table

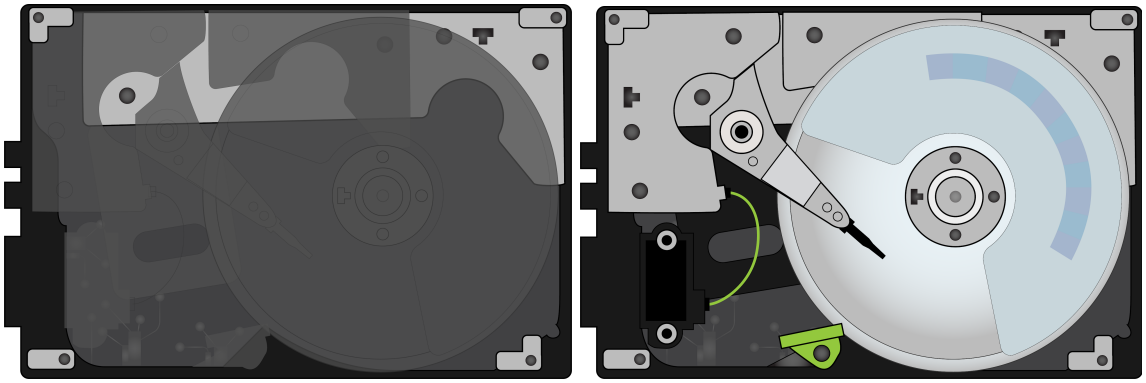


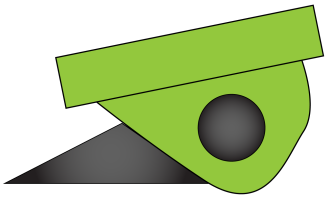
ID	Piece	Snap
01/snap1		
02/snap2		

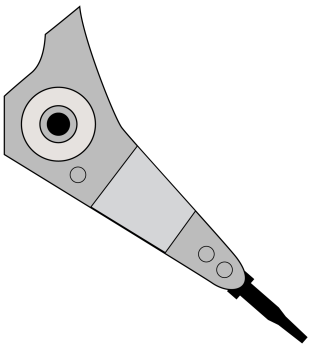
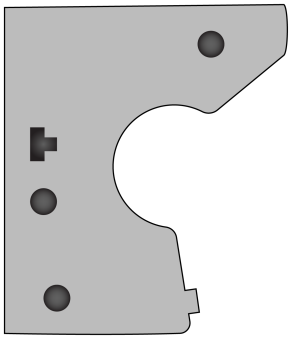
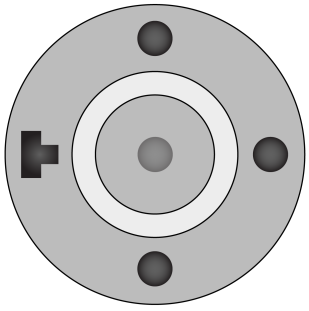
03/snap3		
04/snap4		
05/snap5		
06/snap6		
07/snap7		
08/snap8		
09/snap9		

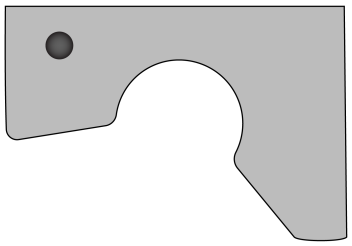
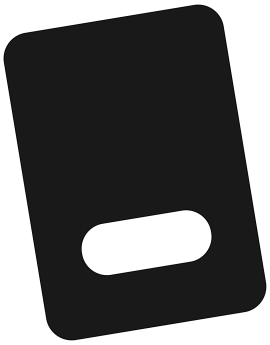
10/snap10		
11/snap11		
12/snap12		

5.1.2 Repair HardDrive Table



ID	Piece	Snap
01/snap1		
02/snap2		

03/snap3		
04/snap4		
05/snap5		
06/snap6		

07/snap7		
08/snap8		
09/snap9		

6. Test Cases

6.1 Introduction

I. The test cases below cover all functionality required by the Software Requirements Specification document. Each of the test cases includes the following information:
   - ID - An identification code for the test case
   - Name - A descriptive name of the test
   - Requirement(s) - The requirement number(s) from the Software requirements specification document
   - Description - A brief overview of the test purpose
   - Preconditions(s) - The expected state of the software before the actions are executed
   - Action(s) - The step(s) to be completed by the tester
   - Postcondition(s) - The expected state of the software after the actions are executed

6.2. Baseline Interactions

ID	0
Name	Interact with Dialogue
Requirement(s)	R2.2.1.1
Description	Regular Dialogue Interactions
Precondition(s)	Reached Dialogue section, npc dialogue displays at top.
Action(s)	1. The Player can highlight selections at the bottom (center in RingCamera and right in Activities) 2. The player selects an option and the correct dialogue displays in the npc panel according to the ink-scripts. 3. The dialogue completely displays before any other dialogue overwrites it, if any. Dialogue choices display again. 4. The player continues to interact via Actions 1-3. 5. Dialogue freezes where Activities must continue, If in an activity, complete the activity according to the correct ID number. restart

	at Action 1 when activity is complete, ignore this action if the activity is complete.
Postcondition(s)	● If Activity or RingCamera dialogue interactions complete (according to the relevant ink-script) exit or start button should be enabled by the last interaction or text completing..

ID	1
Name	Pull the lever
Requirement(s)	R2.2.1.2
Description	Player clicks on the lever to turn on the open sign.
Precondition(s)	Started the game from the main menu.
Action(s)	1. Players walks forwards with W, backwards with S, left with A and right with D 2. Player can look around via moving the mouse 3. Player pulls up the escape menu with esc key and closes it with resume button or esc key 4. Player looks at the lever and it highlights 5. Player interacts with the lever by using the Left Mouse Button. 6. Waits 5 seconds
Postcondition(s)	● A customer arrives at the shop, indicated via the intercom system through a light and audio queue as well as an ‘OPEN’ sign turning on. ● A phone ringing/beeping is the audio cue once a customer arrives

ID	2
Name	Talk to patient outside

Requirement(s)	R2.2.1.3
Description	Player clicks on the intercom box next to the door and interacts with the patient via dialogue
Precondition(s)	Completed Test Case ID: 1
Action(s)	1. Player looks at the intercom box and it highlights 2. Players interact with the intercom system by using the left mouse button on the intercom box 3. Players enter the intercom system screen, which is a 2D “video feed” of the customer 4. Player follows ID 0. 5. Player clicks exit once done.
Postcondition(s)	● Player exits the ring camera 2d display and is back in the main scene. ● A transition plays where the player cannot move for a couple seconds.

ID	3
Name	Entering Activity
Requirement(s)	R2.2.1.4
Description	Player clicks on the buttons to enter the first Activity.
Precondition(s)	Completed Previous Test Case IDs
Action(s)	1. Player clicks buttons on the left hand side of the patient tube that the patient is now in.

Postcondition(s)	● Player enters the first activity.

ID	3.1
Name	Finishing an activity successfully
Requirement(s)	R2*
Description	The Player either finished the final activity or the first one of the day.
Precondition(s)	Completed activity successfully
Action(s)	1. The Player clicks the exit button
Postcondition(s)	● If it's the day's first activity: ○ The next activity loads ○ win condition dialogue displays. ● If it's the day's second activity: ○ See ID 16.

ID	3.2
Name	Finishing an activity and partially failed
Requirement(s)	R2*
Description	The Player either finished the final activity or the first one of the day.
Precondition(s)	Completed activity.
Action(s)	1. The Player clicks the exit button
Postcondition(s)	● If it's the day's first activity:

	○ The next activity loads ○ Partial fail condition dialogue displays. ● If it's the day's second activity: ○ See ID 16.

ID	3.3
Name	Finishing any activity unsuccessfully
Requirement(s)	R2*
Description	The day is restarting
Precondition(s)	Failed an activity
Action(s)	1. None needed by the player.
Postcondition(s)	● Fail condition text displays (akin to you shocked the patient) ● Same day the player was on reloads and player must start at ID1

6.3. Repair Broken Devices Activities

ID	4.1
Name	Playing Repair Broken Devices MicroChip and Succeeding
Requirement(s)	R2.3.1.1 R2.3.1.2 R2.3.1*
Description	Player interacts with the activity
Precondition(s)	Complete ID 1-3.
Action(s)	1. Complete ID 0. 2. The player selects the outlined object

	3. Empty microchip image from 5.1.1 appears. 4. From here on the player is following the Table in 5.1.1 and left clicks the object id 01. 5. The player moves the object around and the green slider starts to decrease in size. 6. An audible sound can be heard when left clicking again to drop it. 7. The player moves component ID 01 to the snap02 ID in the table and drops it using left click. A different sound plays 8. The player left clicks on the component and it gets picked back up. 9. The player moves component ID 01 to the snap01 ID in the table and drops it using left click. A third sound plays. 10. The player left clicks on the component and cannot pick it back up. Mark pass if this is the case, fail if you can pick it back up. 11. The player completely moves all components in place following the rest of the table. 12. The player fills out the check-list under Test in the test-case's ID according to the snap numbers. 13. The player again complete's ID 0.
Postcondition(s)	● An Exit button appears. ● ID 3.1 occurs

Test Results	01	02	03	04	05	06	07	08	09	10	11	12
Pass/Fail	P	P	P	P	P	P	F	P	P	P	P	P

ID	4.2
Name	Playing Repair Broken Devices MicroChip and soft failing
Requirement(s)	R2.3.1.1 R2.3.1.2 R2.3.1*
Description	Player interacts with the activity

Precondition(s)			Complete ID 1-3.									
Action(s)			1. The player follows ID 4.1 until action 5. 2. The player moves the components from ID 01-12 into snaps 01-12 3. The player can choose to have either one or all components in the wrong snaps. 4. Mark pass and fail in test results to show which components were out of place. 5. Once all components are placed complete ID0.									
Postcondition(s)			● An Exit button appears. ● ID 3.2 occurs									
Test Results	01	02	03	04	05	06	07	08	09	10	11	12
Pass/Fail	P	P	P	P	P	P	P	P	P	P	P	P

ID	5.1
Name	Playing Repair Broken Devices HardDrive and Succeeding
Requirement(s)	R2.3.1.1 R2.3.1.2 R2.3.1*
Description	Player interacts with the activity
Precondition(s)	Complete ID 1-3.
Action(s)	1. Complete ID 0. 2. The player selects the outlined object 3. Empty harddrive image from 5.1.2 appears. 4. From here on the player is following the Table in 5.1.2 and left clicks the object id 01. 5. The player moves the object around and the green slider starts to decrease in size.

	6. An audible sound can be heard when left clicking again to drop it. 7. The player moves component ID 01 to the snap02 ID in the table and drops it using left click. A different sound plays 8. The player left clicks on the component and it gets picked back up. 9. The player moves component ID 01 to the snap01 ID in the table and drops it using left click. A third sound plays. 10. The player left clicks on the component and cannot pick it back up. Mark pass if this is the case, fail if you can pick it back up. 11. The player completely moves all components in place following the rest of the table. 12. The player fills out the check-list under Test in the test-case's ID according to the snap numbers. 13. The player again complete's ID 0.
Postcondition(s)	● An Exit button appears. ● ID 3.1 occurs

Test Results	01	02	03	04	05	06	07	08	09
Pass/Fail	P	P	P	P	P	P	P	P	P

ID	5.2
Name	Playing Repair Broken Devices HardDrive and soft failing
Requirement(s)	R2.3.1.1 R2.3.1.2 R2.3.1*
Description	Player interacts with the activity
Precondition(s)	Complete ID 1-3.

Action(s)			6. The player follows ID 5.1 until action 5. 7. The player moves the components from ID 01-12 into snaps 01-12 8. The player can choose to have either one or all components in the wrong snaps. 9. Mark pass and fail in test results to show which components were out of place. 10. Once all components are placed complete ID0.									
Postcondition(s)			● An Exit button appears. ● ID 3.2 occurs									
Test Results	01	02	03	04	05	06	07	08	09	10	11	12
Pass/Fail	P	P	P	P	P	P	P	P	P	P	P	P

ID	6
Name	Failing any Repair Broken Devices Activity
Requirement(s)	R2.3.1*
Description	Player interacts with the activity and fails
Precondition(s)	Completed Test Case ID1-3 ID 4.1 until Action 6
Action(s)	1. The player moves any component around until the green slider disappears. 2. The component automatically drops.
Postcondition(s)	● ID 3.3 occurs

6.4. PaintJob Activity Actions

ID	7.0
Name	Selecting Color
Requirement(s)	R2.3.2* R2.3.2.1
Description	Player sets color to follow given color
Precondition(s)	Completed Test Case IDs 1-3 and in Paintjob Activity Test Case under section 6.4
Action(s)	1. The player interacts with the red color splotch 2. Current color's R value increases by 2 3. The player interacts with the green color splotch 4. Current color's G value increases by 2 5. The player interacts with the blue color splotch 6. Current color's B value increases by 2 7. The Player interacts with the white color splotch 8. Current color's R, G, and B values increase by 2 each 9. The Player interacts with the black color splotch 10. Current color's R, G, and B values decrease by 2 each 11. The player left clicks accept paint and nothing occurs. 12. The Player left clicks the reset button and all of Current Color's values reset to 0. 13. The player selects the Red color splotch until Current Color's R value is the same as Desired Color's R value. 14. The player selects the Green color splotch until Current Color's G value is the same as Desired Color's G value. 15. The player selects the Blue color splotch until Current Color's B value is the same as Desired Color's B value. 16. The player clicks the accept paint button.
Postcondition(s)	● Activity moves to the spraypainting phase.

ID	7.1
Name	Playing Paintjob Activity
Requirement(s)	R2.3.2* R2.3.2.1
Description	Player interacts with the activity and dialogue
Precondition(s)	Completed Test Case IDs 1-3
Action(s)	1. Player can choose to complete Test Case ID0 2. Player completes Test Case ID 7.0 3. The player sprays the paint via left click onto the component 4. The amount of spraypaint on the left drops as the player sprays 5. The player can left click finish painting at any point to finish painting. 6. The player completes Test Case ID0 after clicking finish painting 7. The player can choose to paint until they run out of paint, in which the player is sent straight to Test Case ID0.
Postcondition(s)	● Test Case ID 3.1 runs always

6.5. Upgrades Game Activity Actions

ID	10
Name	Upgrades Game Screwdriver Test
Requirement(s)	R2.3.3*
Description	Player uses ScrewDriver Tool
Precondition(s)	Player has started an Activity TestCase under section 6.5
Action(s)	1. Player left clicks on the red screwdriver in the toolbar. 2. Player moves mouse around and the screwdriver follows

	3. Player left clicks anywhere on the object and the green health bar at the top decreases. 4. Player left clicks a screw on the floating 3D object and the screw attaches to the screwdriver. 5. Player left clicks anywhere on the object and the green health bar at the top decreases. 6. Player right clicks in the toolbar and the screw drops into the toolbar 7. Player left clicks the screw and it gets picked back up 8. Player tries to right click to place the screw anywhere outside of the toolbar and fails 9. Player right clicks to place the screw back down in the toolbar. 10. Player tries to right click to place the screwdriver anywhere outside of the toolbar and fails 11. The player right clicks to drop the screwdriver back in the toolbar
Postcondition(s)	● Game is still playable, player can still pick up the screwdriver

ID	10.1
Name	Upgrades game Forceps Test
Requirement(s)	R2.3.3*
Description	The Player uses the Forceps Tool
Precondition(s)	Player has started an Activity TestCase under section 6.5
Action(s)	1. Player left clicks on the gray forceps in the toolbar. 2. Player moves mouse around and the forceps follows 3. The Player left clicks anywhere on the object outside of the toolbar and the green bar decreases at the top.

	4. Player left clicks the square component in the section of the 3d object that was revealed previously in the Activity test case occurring under section 6.5. 5. The component attaches to the forceps 6. The component follows the forceps 7. Player right clicks to drop the component in the toolbar. 8. Player right clicks to drop the forceps 9. Player left clicks on the forceps to pick up the forceps 10. Player left clicks on the newer-looking square component to pick it up 11. Player right clicks to drop the newer looking component 12. Player right clicks to drop the forceps.
Postcondition(s)	● Game is still playable, player can still pick up the forceps

ID	11
Name	Playing Upgrades Game successfully
Requirement(s)	R2.3.3* R2.3.3.1 R2.3.3.2
Description	The Player interacts with the activity and dialogue
Precondition(s)	Completed Previous Test Case IDs
Action(s)	1. Player completes Test Case ID 0 2. Player completes Test Case ID 10 3. The player follows actions 1-4 in Test Case ID 10 for the last 3 screws. 4. The player drops the screwdriver by right clicking in the tool area 5. The player left clicks on and moves the cover the screws were previously on 6. After the cover is completely moved away the Player follows Test Case ID 10.1 7. The player picks up the gray forceps via left click

	8. The player picks up the newer looking component that was already in the toolbar via left click 9. The player right clicks to drop the newer component into the place where the old one started in the 3D object 10. The player drops the forceps via right click back onto the toolbar when hovering over it 11. The player left clicks the cover and moves it back over the open section 12. The player drops the cover with left click 13. The player left clicks the screwdriver to pick it up 14. The player left clicks a screw to pick it up 15. The player right clicks to drop it into the screw outline on one of the four corners of the cover. 16. The player completes actions 14-15 for the last 3 screws 17. The player right clicks in the toolbar to drop the screwdriver back into the toolbar 18. Player completes Test Case ID 0 again.
Postcondition(s)	● If the cybernetic stability is above 40% Test case 3.1 applies ● If the cybernetic stability is between 20% and 40% test case 3.2 applies.

ID	12
Name	Playing Upgrades game
Requirement(s)	R2.3.3*
Description	Player interacts with the activity and fails
Precondition(s)	Completed Previous Test Case IDs

Action(s)	1. Player completes Test Case ID 0 2. Player completes Test Case ID 10 3. The player follows actions 1-4 in Test Case ID 10 for the last 3 screws. 4. The player drops the screwdriver by right clicking in the tool area 5. The player left clicks on and moves the cover the screws were previously on 6. After the cover is completely moved away the Player follows Test Case ID 10.1 7. The player picks up the gray forceps via left click 8. The player picks up the newer looking component that was already in the toolbar via left click 9. The player right clicks to drop the newer component into the place where the old one started in the 3D object 10. The player drops the forceps via right click back onto the toolbar when hovering over it 11. The player left clicks the cover and moves it back over the open section 12. The player drops the cover with left click 13. The player left clicks the screwdriver to pick it up 14. The player left clicks a screw to pick it up 15. The player right clicks to drop it into the screw outline on one of the four corners of the cover. 16. The player completes actions 14-15 for the last 3 screws 17. The player right clicks in the toolbar to drop the screwdriver back into the toolbar 18. Player completes Test Case ID 0 again. 19. At any point in time the player may use a tool to left click on an object and lose all health before completing the game. This is the last point the player can lose
Postcondition(s)	● Test Case ID 3.3 applies

6.6. Memory Solver Activity Actions

ID	14
Name	Playing Memory Solver
Requirement(s)	R2.3.4* R2.3.4.1 R2.3.4.2
Description	Player interacts with the activity and dialogue
Precondition(s)	Completed Test Case IDs 1-3
Action(s)	1. Player completes Test case ID0. 2. The player select the start button to start the memory solver game 3. The player clicks one of the three Serum buttons and the values on the relevant sine waves increase or decrease 4. The player keeps the values low enough for the text numerics to not be red but high enough for it to not be blue 5. When the text numerics aren't either color the Stabilizing Rate should count down from a preset value dependent on the day 6. The player should left click the serum buttons to make the text numerics either red or blue 7. The System Shutdown timer should appear and start from a preset value dependent on the day 8. The player should select the serums until the text stabilizes to black 9. The Stabilizing Rate should continue counting down from where it stopped 10. The player should select the serums until the colored text returns 11. The System Shutdown timer should restart from the preset value for the day 12. The player re-stabilizes the patient by selecting the correct serums to return the values colors to black

	13. The player should keep the patient stabilized until the timer runs out 14. The activity will stop 15. Player completes Test Case ID0
Postcondition(s)	● Test Case ID 3.1 runs

ID	14.1
Name	Playing Memory solver and failing
Requirement(s)	R2.3.4*
Description	Player interacts with the activity and fails
Precondition(s)	Completed Test Case IDs 1-3
Action(s)	1. Player Completes ID0. 2. The player select the start button to start the memory solver game 3. The player clicks one of the three Serum buttons and the values on the relevant sine waves increase or decrease 4. The player gets the text into the blue or red and keeps it there 5. System Shutdown timer runs out 6. Game stops
Postcondition(s)	● Test Case ID 3.3 runs

6.7. Day-Specific Activity Differences

ID	15
Name	Playing Memory solver on day 5

Requirement(s)	R2.3.4*
Description	Player interacts with the activity and fails
Precondition(s)	Completed Previous Test Case IDs
Action(s)	1. Player completes ID 0 2. The player select the start button to start the memory solver game 3. The player clicks one of the three Serum buttons and the values on the relevant sine waves increase or decrease 4. The player gets the text into the blue or red and keeps it there 5. System Shutdown timer runs out 6. Player completes ID 0 7. Exit button displays
Postcondition(s)	● Player returns to their body in 3d environment ● Player can move and interact with interactable objects ● YPZ shouldn't show up on day 7 ● Twinkle will show up on day 7

6.8. Exiting Activity Actions

ID	16
Name	Finishing The Day
Requirement(s)	R2.2.1.6 R2.2.1.7
Description	Player returns to 3D camera and goes to next day
Precondition(s)	Completed Previous Test Case IDs
Action(s)	1. The player turns and left clicks on the monitor that is lit up
Postcondition(s)	● Next Day and number is displayed correctly ● The player starts in the same area as the first day

	• All dialogue and functionality is set up for the next day based on results from second activity returning to either ID 3.1 or 3.2

7. Traceability Matrix

Requirement	Description	Test Cases(ID)
R2.1*	**Controls**	**Tests Below**
R2.1.1*	Default Controls.	No Tests Included
R2.1.2*	Activity and Ring Camera Controls.	ID 0,2 and within activity ID's.
R2.2*	**Gameplay Loop**	**Tests Below**
R2.2.1*	Day Cycle	3.1, 3.2, 3.3
R2.2.1.1	Start Workday	1
R2.2.1.2	Customer Arrives	2
R2.2.1.3	Diagnostic Phase	2
R2.2.1.4	Service Chamber	3
R2.2.1.5	Activity Fixes	4-15
R2.2.1.6	Customer Exits	16
R2.2.1.7	End Workday	16
R2.3*	**Activities**	**Tests Below**
R2.3.1*	Repair Broken Devices	4.1, 4.2, 5.1, 5.2, 6
R2.3.2*	Paint Job	7.1, 7.2
R2.3.3*	Upgrades, People, Upgrades	10, 10.1, 11, 12
R2.3.4*	Memory Solver	14, 14.1, 15
R2.6*	**User Interface**	**Tests Below**

Requirement	Description	Test Cases(ID)
R2.1*	**Controls**	**Tests Below**
R2.1.1*	Default Controls.	No Tests Included
R2.6.1*	Main Menu	No Tests Included
R2.6.1.1	Start button	No Tests Included
R2.6.1.2	Options button	No Tests Included
R2.6.1.3	Exit Button.	No Tests Included
R2.6.2*	Pause Menu	No Tests Included
R2.6.2.1	Continue button	No Tests Included
R2.6.2.2	Options button	No Tests Included
R2.6.2.3	Exit to Menu button	No Tests Included

8. Tools

- Clickup
- Game Assets
- Game Manager
- Game Website
- Miro
- Perforce
- Unity

References

Acceptance Testing Documentation With Real-TIme Scenarios. (2023, March 21). *Software Testing Help.* Retrieved from: https://www.softwaretestinghelp.com/acceptance-test-plan/

Lee, D. (2023, February 3). How To Write a Test Plan (With Definition and Importance). Indeed. Retrieved from: https://www.indeed.com/career-advice/career-development/how-to-write-test-plan

Pelletier, N., et al. (2023). Software Requirements Specifications. *Firmware Infirmary.* Retrieved from: https://docs.google.com/document/d/1Sxk-EWGOqNRismxYbD2RH_CtleNII2QCIbQ-939iZ0E/edit#