X-HAB 2020 ACADEMIC INNOVATION CHALLENGE

NEXT GENERATION USER INTERFACES FOR GATEWAY AUTONOMOUS OPERATIONS



FINAL REPORT

May 20 2020

Project Team

Dr. J. Cecil (PI), Dr. Rajesh Krishnamurthy, Dr. Shelia Kennison,

Dr. Chris Crick, Dr. S. Aakur, Dr. Johnson Thomas

Student Teams Coordinator: Avi Gupta

Oklahoma State University

TABLE OF CONTENTS

SL.NO	CONTENTS	PAGE.NO
1.	Vision and Mission	6
2.	Background and Goal	6
3.	Overview of the three AR based user interfaces and environments	7
4.	System Design	7
	4.1 Cyber-Human design of AR based user interfaces	7
	4.2 Systems engineering based project implementation	8
	4.3 UML Activity diagram	10
	4.3.1 CONOPS activity diagram for RAM replacement scenario	10
	4.3.2 CONOPS activity diagram for payload storage scenario	11
	4.4 UML Sequence diagram	12
	4.4.1 RAM replacement scenario	13
	4.4.2 Payload storage scenario	14
5.	Requirements	14
	5.1 Internal / External interface requirements	15
	5.2 Functional requirements	15
6.	Team – 1 (RIDE)	16
7.	Team – 2 (PHOBOS)	17
8.	Team – 3 (ARTS)	18
9.	Physical Validation environment	20
10.	Testing Plan	20
	10.1 Test cases for unit testing and functional testing	21
	10.2 Testing feedback	22

10.2.1 Team RIDE 22
10.2.2 Team ARTS 23
10.2.3 Team PHOBOS 23
11. Risk management 24
12. Validation and Assessment 26
13. Dissemination and Outreach 31
14. Formative Report 32
15. Summative Report 32
16. Conclusion 32
17. References 33

LIST OF FIGURES

Fig. No	Figure Name	Page no.
3.1	Overview of the three AR based UIs and prototype environments	7
4.1	Role of Systems Engineering, Software Engineering and Cyber-Human Design principles in the process of Designing User Interfaces	8
4.2	The IDEF-0 model for project implementation	9
4.3	Work Breakdown Structure	10
4.4	ConOps - UML Activity diagram for Ram Replacement Scenario	11
4.5	ConOps - UML Activity diagram for Payload Storage Scenario	12
4.6	UML Sequence diagram for RAM Replacement Scenario	13
4.7	UML Sequence diagram for payload storage Scenario	14
6.1	Main Menu UI	17
6.2	View of the options for Augmented Reality Training Environment UI	17
6.3	View of Avatar based automated view of the training steps	17
6.4	View of the menu options for step-by-step overview	17
6.5	View of an Avatar showing the screw removal step for the step-by-step overview	17
7.1	Number code to be scanned to initiate the training	18
7.2	The view of the welcome scene after the number code is scanned	18
7.3	Help menu options in the automated segment of the UIs	18
7.4	View of the finger based Bloom Gesture	18
7.5	View of the finger based Pinch Gesture	18
7.6	One of the options in the AR interface under “Act” menu options	18
8.1	Main Menu UI	19
8.2	Training Environment UI	19

8.3	Help UI to familiarize users with Pinch and Bloom gestures	19
8.4	UI to scan the QR code	19
8.5	View of the AR based environment showing the location of payload and the storage	19
9.1	A user following the AR based instructions and replacing the laptop component	20
9.2	A user following the AR based instructions and storing the payload	20
10.1	Test Cases for Unit Testing	21
10.2	Test Cases for Functional Testing	22
11.1	Risk Matrix	26

1. VISION AND MISSION

NASA's Gateway seeks to establish an autonomous platform in support of Artemis mission (boots on ground 2024). It is used to refine and mature short and long-duration deep space exploration capabilities through the 2020s. It is expected to be assembled in a lunar orbit where it can be used as a staging point for missions to the Moon and other destinations in deep space. Gateway can evolve for different mission needs involving exploration, science, commercial and international partners. User interfaces for autonomous systems is an emerging area where knowledge and implementation concepts are still in their infancy.

2. BACKGROUND AND GOAL

OSU's Center for Cyber-Physical Systems is a leader in the design of software and IT based systems for a variety of domains such as manufacturing, medicine, and space systems.

The X-Hab Project Goal is to design Augmented Reality (AR) based Cyber-Human interfaces to support autonomous activities within the Gateway; specifically, our project objectives were to create AR based interfaces to assist astronauts in performing 2 activities within the Gateway:

(1) To perform a service task involving replacement of a RAM board on a laptop. Two AR user interface based prototypes were created using the Vive™ and HoloLens™ platforms respectively.

(2) To transfer a payload from one location to another within the Gateway. An AR based prototype was created using the HoloLens™ platform

The overall AR based user interface was designed with an emphasis on user centric design, adoption of next generation Augmented Reality technologies, Systems Engineering and Software Engineering principles.

The project activities were accomplished over a period of 2 semesters (fall 2019 and spring 2020) as part of course project activities in the department of Computer Science at Oklahoma State University. They were undertaken as part of course project activities involving a multi-disciplinary group of undergraduate and graduate students from computer science, electrical engineering, and computer engineering.

The planned activities were accomplished adopting a systems engineering (SE) approach while emphasizing design concepts based on software engineering principles. The design of the target user interfaces was based on Human-centric principles (formerly referred to as Human-Computer Interactions HCI) and explored adoption of next generation Virtual Reality techniques and technologies to facilitate immersive validation of proposed designs.

There were 2 primary outcomes or products of this X-Hab Project, which include:

I. Design and implementation of prototype software tools with associated graphics and immersive Augmented Reality (AR) capabilities could be used to serve as validation contexts for design and operation of the targeted set of user interfaces. An important component within this suite of tools would to be create 3D AR based environments, which

would be capable of interfacing with Immersive AR platforms such as the HTC Vive and Microsoft HoloLens; such a capability will provide an opportunity for crew and ground-support members to interact, provide feedback and validate the functionalities of the proposed user interfaces.

II. User assessment of the developed AR based environments and elaborate on the technical comparison between the two AR platforms (Vive and HoloLens).

3. OVERVIEW OF THE THREE AR BASED USER INTERFACES AND ENVIRONMENTS

The three student teams focused on creating AR based training simulators for RAM replacement and payload storage scenario. The designed innovative User Interfaces (UIs) supported autonomous operation of NASA's proposed gateway and the associated modules. Prototypes of the 3 User interfaces were created to highlight the design elements, menu options and avatar based interactions. These interfaces were captured as simulation environments to obtain user feedback and validation of the UIs. An overview of the three AR based environments was represented as shown in figure 3.1.

Teams	Simulation Environment	Platform	Interaction Mechanism
Team 1 (RIDE)	AR based simulator to train astronauts in replacing laptop components	Vive	Controller
Team 2 (ARTS)	AR based simulator to train astronauts in payload storage	HoloLens	Hand Gesture
Team 3 (PHOBOS)	AR based simulator to train astronauts in replacing laptop components	HoloLens	Hand Gesture

Fig 3.1 Overview of the three AR based UIs and prototype environment

4. SYSTEM DESIGN

The Unified Modeling Language (UML) based Sequence and activity diagrams were used to model temporal interactions and functional calls between the software objects used to build the UI. The initial software design included initial layouts of the menus, user options, etc. which were refined through many iterations during the periods between the SDR, the PDR and CDR deadlines. The 2D layouts of the user interfaces were used to create 3D layouts of user interface menus, buttons and user options, which were eventually used as a foundation to develop the AR based immersive environment of the UI designs proposed.

4.1 CYBER – HUMAN DESIGN OF AR BASED USER INTERFACES

For the design of User Interfaces (UIs), Cyber-Human design (which has roots in Human Computer Interaction HCI) principles were adopted, which emphasizes usability, user characteristics, environment and tasks. We created immersive 3D VR based digital mockups of the UI design using Unity and HTC Vive/ Microsoft HoloLens VR platforms. These next generation UIs will allow crew and ground support to interact immersively and accomplish various user interface functionalities. The student teams have also incorporated 3D avatars to facilitate the training process. The role of Systems Engineering, Software Engineering and Cyber-Human Design principles in the process of Designing User Interfaces AR based user interface was illustrated as shown in figure 4.1.

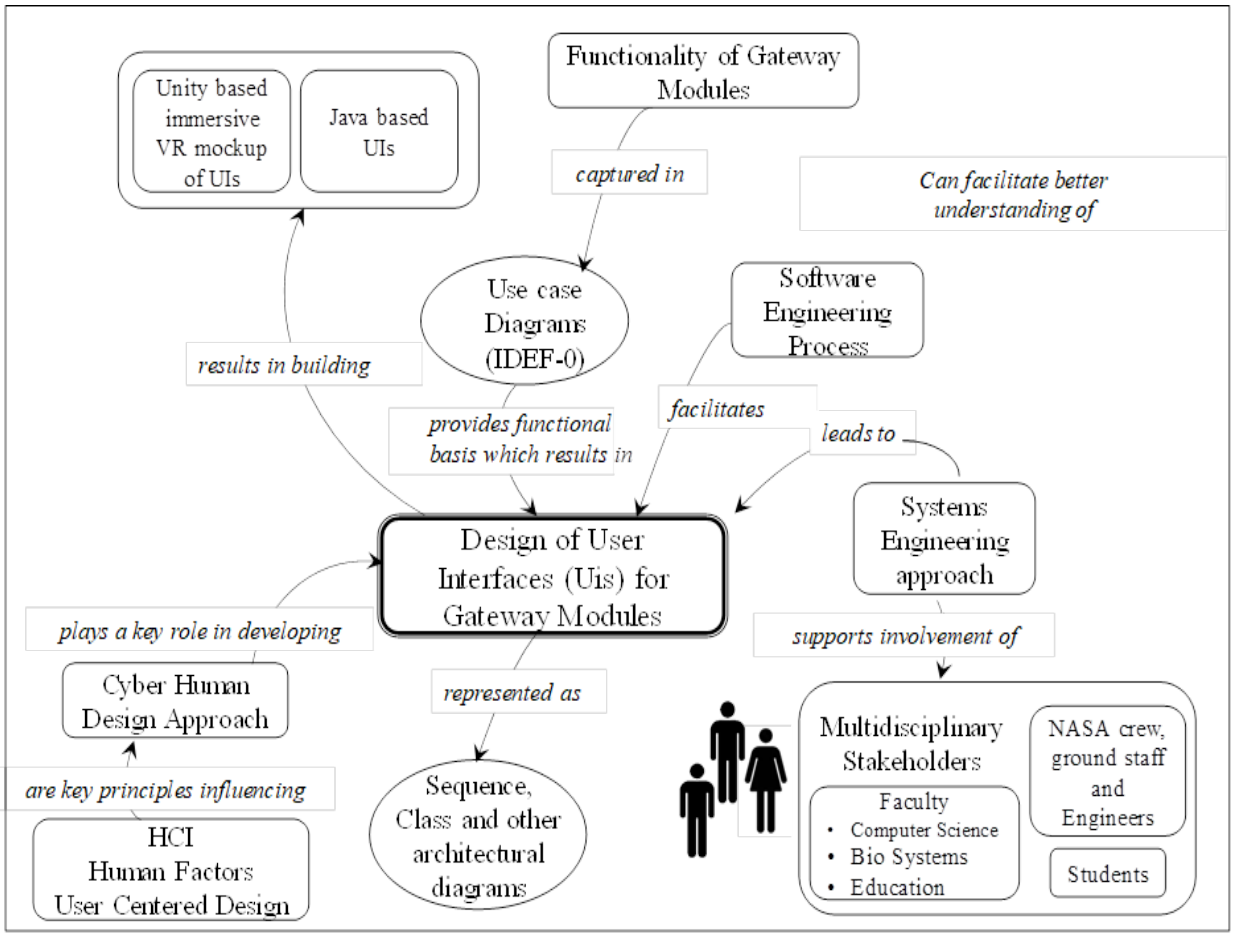


Fig 4.1 Role of Systems Engineering, Software Engineering and Cyber-Human Design principles in the process of Designing User Interfaces

4.2 SYSTEMS ENGINEERING BASED PROJECT IMPLEMENTATION

The activities in the project were planned and implemented based on systems engineering principles; an IDEF-0 model figure 4.2 was created to enable the student teams to plan and implement the project activities. The student teams adopted the IDEF-0 modeling method to highlight the adoption of systems engineering principles while delineating the functional flow of

the various activities. IDEF stands for Integrated Definition), which is an information modeling industry standards that also supports adoption of systems engineering principles.

IDEF-0 models, in general, are capable of graphically modeling a wide variety of software development, manufacturing and other processes at desired levels of detail (or abstraction). An IDEF-0 model consists of hierarchical series of diagrams, text, and glossary cross-referenced to each other.

In this project, the X-Hab project activities (completed by students) appear as functions (verb phrases inside rectangular boxes); the relevant data and objects that interrelate those functions are represented by arrows and fall into 4 categories called ICOMs (Inputs I, Controls C, Mechanisms M and Outputs O). The direction of the arrows (relationships) entering or leaving each activity box is shown in the legend inset in figure 4.2. The control attributes refer to concepts or data which control the performance of a given activity (e.g.: schedule, mission objectives, etc.) The inputs refer to physical or data/information inputs necessary to perform an activity (requirements, etc.). The outputs refer to activity outcomes (which can be completion of a design etc.). Mechanisms are resources (software, humans, equipment, etc.) needed to perform a given function. The key activities shown in IDEF 0 Model (figure 4.2) were undertaken by the students as part of course project activities include:

Activity 1 (A1): Problem and Domain Understanding (Inception Phase)
Activity 2 (A2): Define Requirements and Systems Design (Inception Phase)
Design of the User Interfaces (Elaboration Phase) involves 2 related activities:
Activity 3 (A3): Complete Preliminary Design of User Interfaces
Activity 4 (A4): Complete Critical Design of User Interfaces
Activity 5 (A5): Build, Test and Validate the User Interfaces (Construction and Transition Phase)
Activity 6 (A6): Completion of Reports (Construction and Transition Phase)

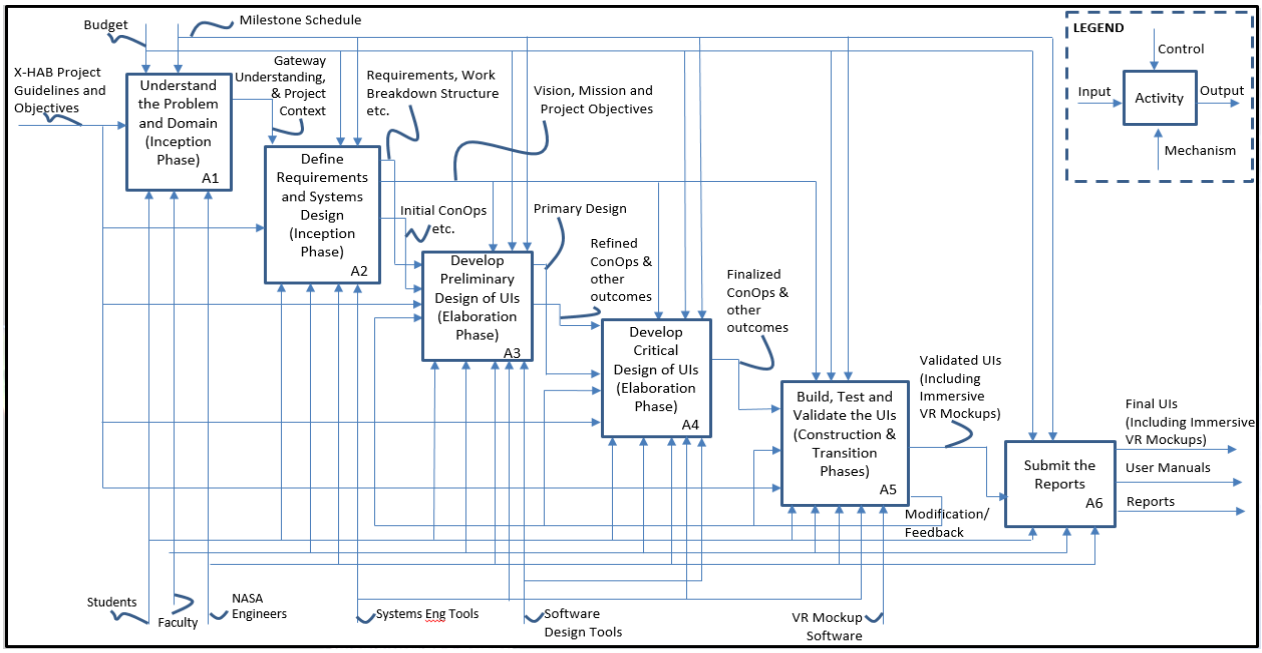


Fig 4.2 The IDEF-0 model for project implementation

4.3 UML ACTIVITY DIAGRAM

The Concept of operations (CONOPS) denoting the system characteristics during the entire activity, functional requirements and the result of conceptual analysis was represented as shown in the figure 4.3. The UML Activity diagram describes the workflow behavior of a system, co-ordination of the activities and the service at different levels of abstraction.

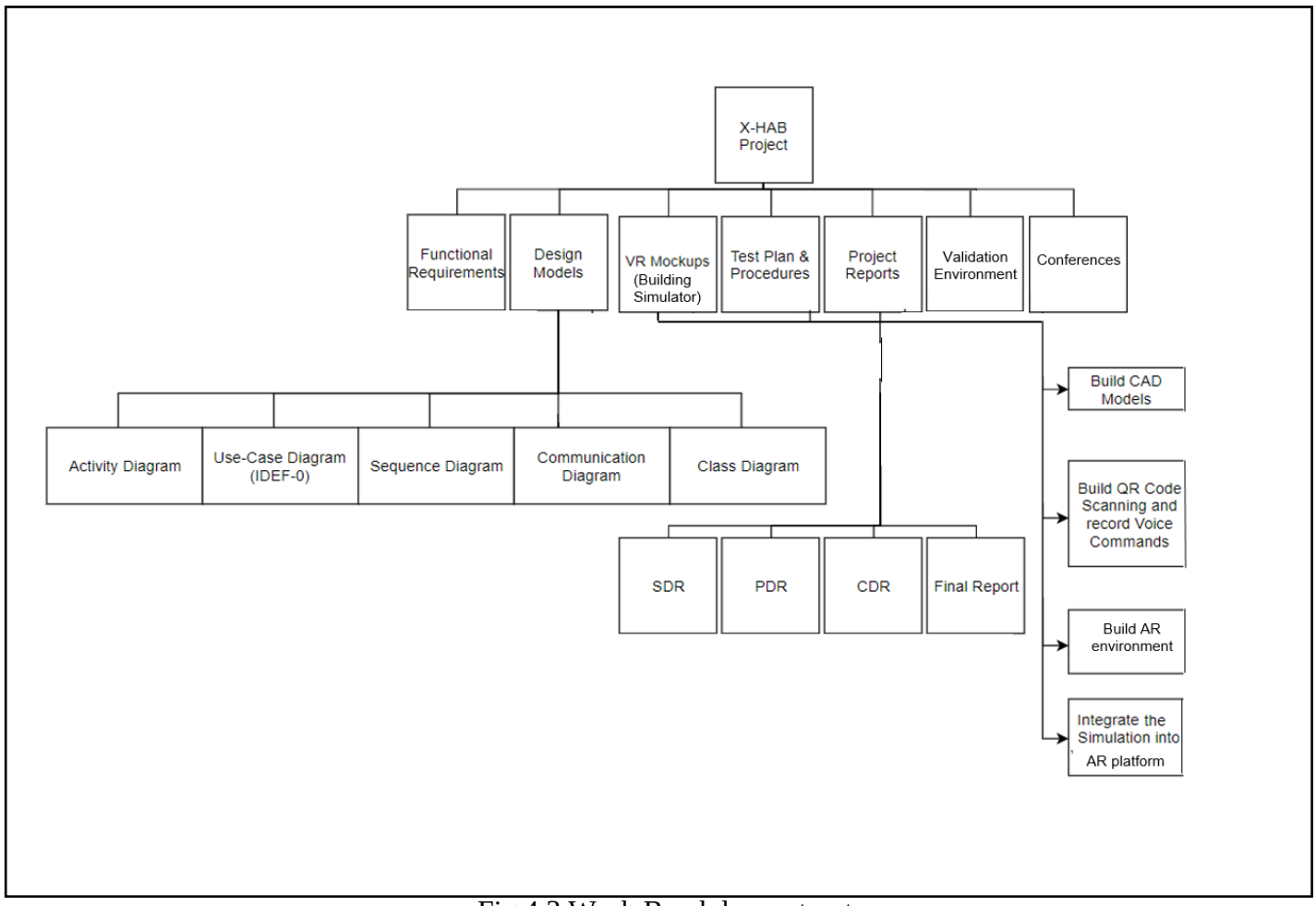


Fig 4.3 Work Breakdown structure

The work flow analysis involved in the RAM replacement scenario and the payload storage scenario were represented as UML activity diagram. The dependencies between the activities were captured in these UML based activity diagrams.

4.3.1 CONOPS Activity diagram for RAM Replacement Scenario

The sequence of activities and the workflow involved in the RAM Replacement scenario was represented as an activity diagram in figure 4.4.

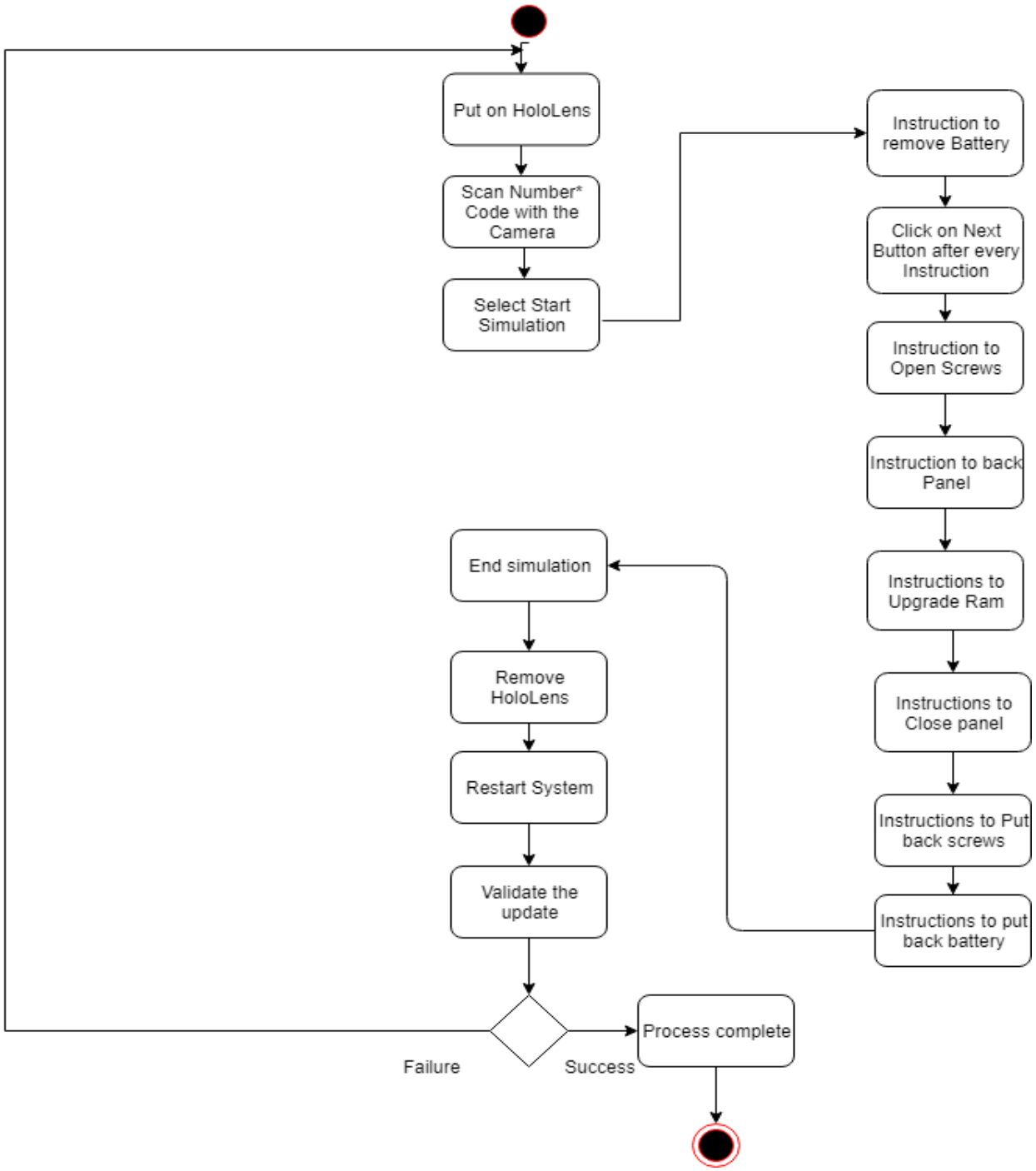


Fig 4.4 ConOps - UML Activity diagram for Ram Replacement Scenario

4.3.2 ConOps Activity diagram for Payload Storage Scenario

The order of activities that take place in the payload storage scenario was represented in the form of a UML activity diagram as shown in figure 4.5.

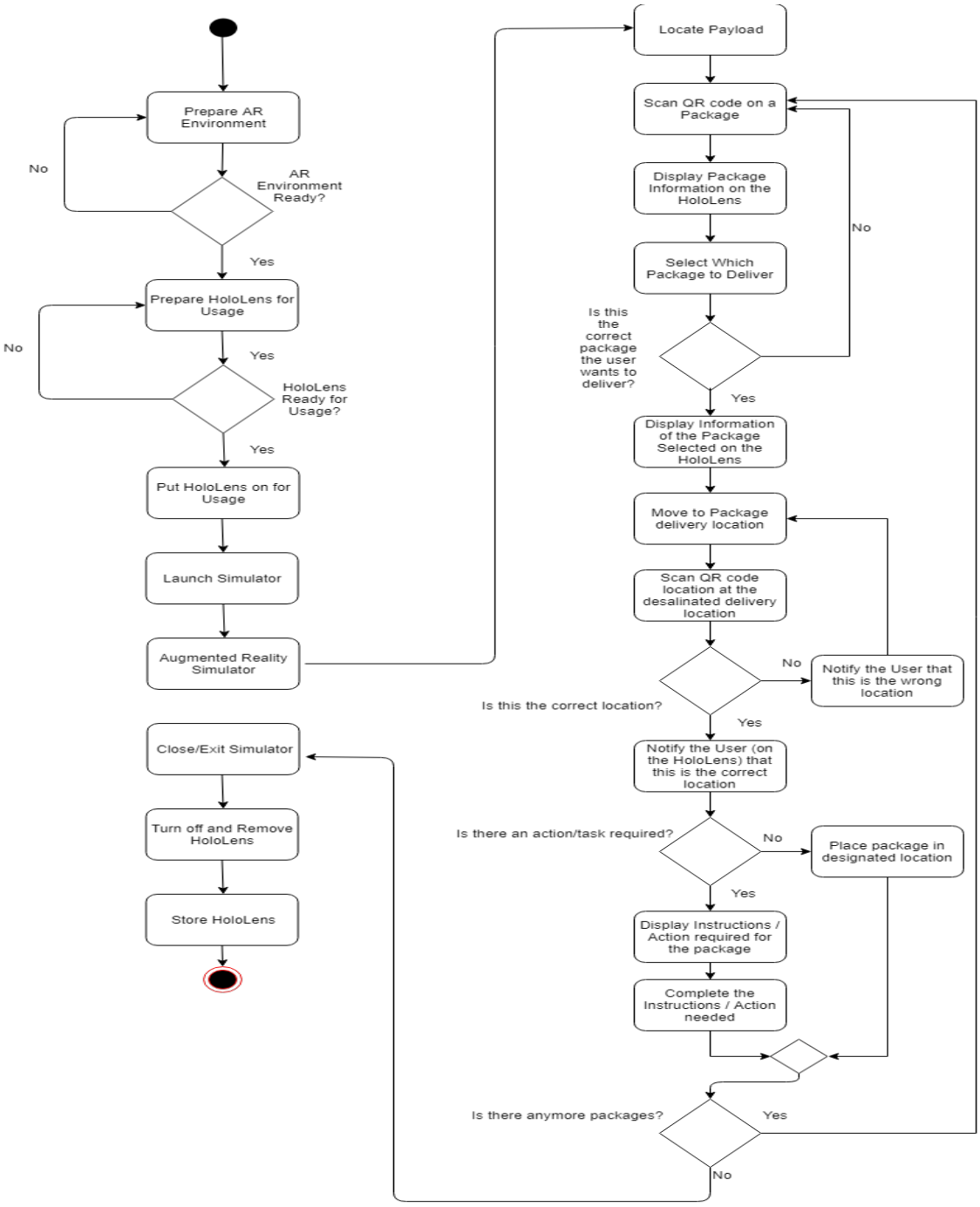


Fig 4.5 ConOps - UML Activity diagram for Payload Storage Scenario

4.4 UML SEQUENCE DIAGRAM

The UML Sequence diagram describes the interactions in the system. The model represents how objects operate with one another and the order of their operation. The object interactions are arranged in time sequence. The design, analysis and the logical flow of the system was diagrammatically represented in the order of interaction as a UML sequence diagram. The sequence diagrams were created understand the interactions of the 2 activities in the Gateway viz

(i) RAM replacement activities on a laptop and (ii) transfer payload from one location to another within the Gateway.

4.4.1 RAM Replacement Scenario

The sequence of steps involved in the RAM replacement scenario was represented as a UML sequence diagram as shown in figure 4.6. The order of interaction and the logical flow involved are:

Step 1 – Removal of the battery
Step 2 – Removal of the 2 screws
Step 3 – Detachment of the back panel
Step 4 – RAM upgrade
Step 5 – Attachment of the back panel
Step 6 – Placing the screws back
Step 7 – Reinserting the battery

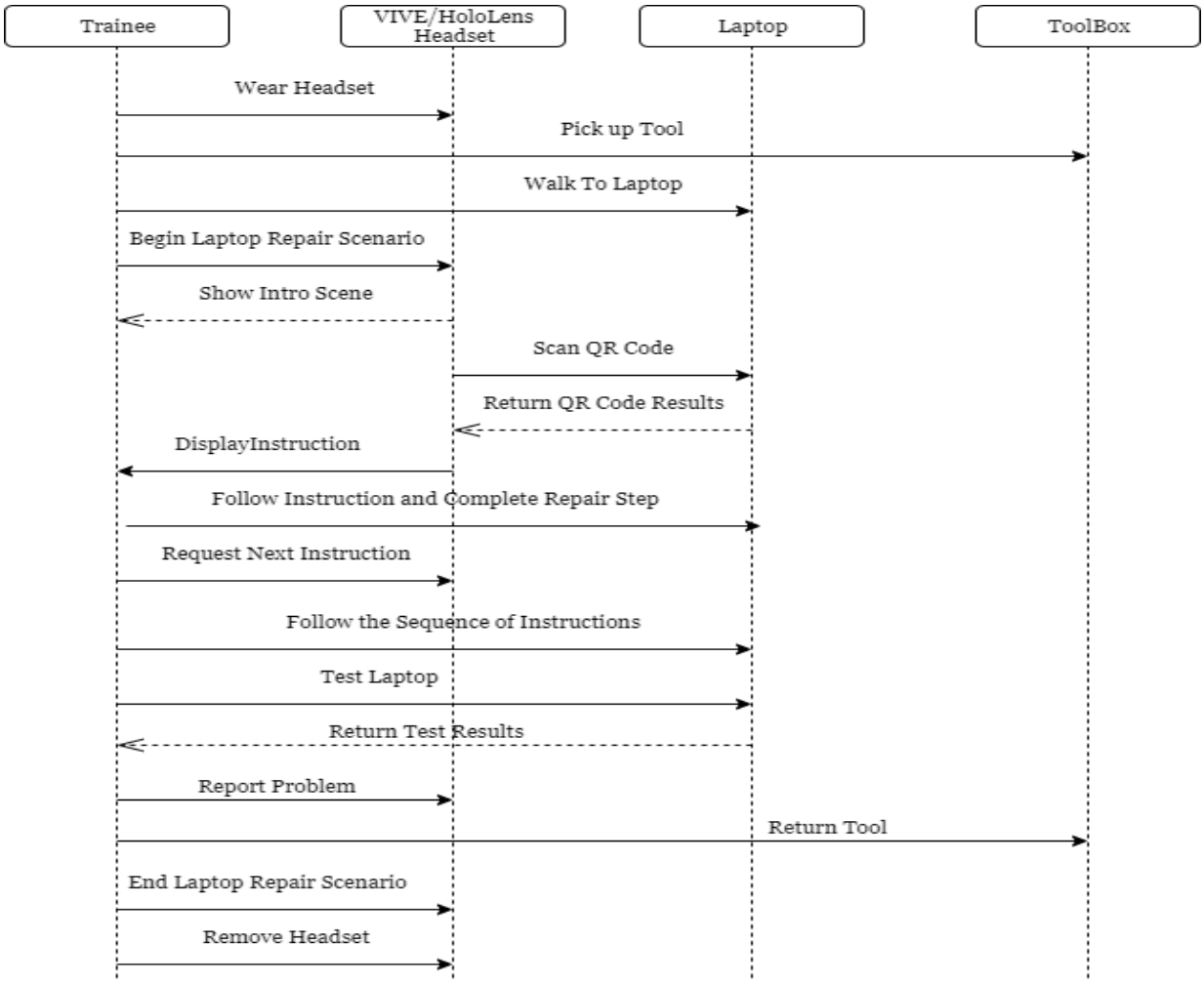


Fig 4.6 UML Sequence diagram for RAM Replacement Scenario

4.4.2 Payload Storage Scenario

The order of interactions that occur in the Payload Storage Scenario was represented as a sequence diagram as shown in figure 4.7. The sequence of steps involved in payload storage are:

Step 1 – Locating the Payload
Step 2 – Scanning code of the payload information
Step 3 – Locating the storage location
Step 4 – Unloading the payload for storage
Step 5 – Storing payload at the designated location

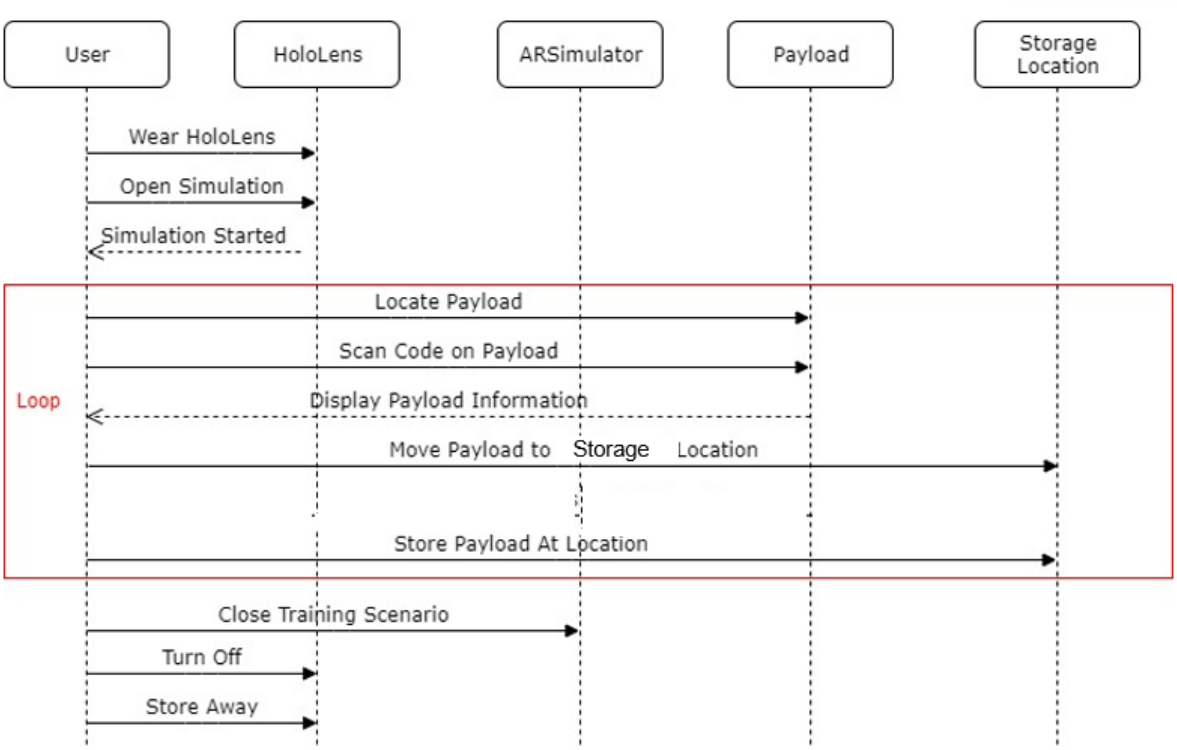


Fig 4.7 UML Sequence diagram for payload storage Scenario

5. REQUIREMENTS

The features required for the simulator to perform tasks as intended are categorized into Internal / External interface requirements and functional requirements. The Analysis tools used for building the system to perform simulation are as follows:

(i) Design tools:

- AI0 WIN - for IDEF-0 model development
- Microsoft Visio 2019 (V16.0) - for creating UML diagrams
- SolidWorks, Sketchup, FormZ, AutoCAD, etc... - for modeling the required 3D objects

- Microsoft Visual Studio – For built-in software design tools

(ii) Development tools:

- Unity - for 3D modeling and creation of simulation environment
- Microsoft Visual Studio – For a more dynamic coding environment
- SteamVR ToolKit, Mixed Reality Tool Kit - to aid in developing augmented reality application

(iii)Hardware devices:

- Microsoft HoloLens and HTC Vive

5.1. INTERNAL / EXTERNAL INTERFACE REQUIREMENTS

Interface Requirements	
AR Platforms	The AR simulator will be integrated with either of the following interfaces during training: 1. HTC VIVE 2. HoloLens The user interactions will be in the form of text, voice or avatar-based commands
Software Interfaces	• The AR simulator will be implemented on a windows operating system. • The AR simulator will be developed on the Unity 3D engine. • The simulator needs to support the camera-based interface to understand the trainee's gestures and provide appropriate response.

5.2 FUNCTIONAL REQUIREMENTS

The functional requirements describe the characteristic of the system behavior. The response of the system was based on the functionalities designed. The simulator has to meet the functional requirements for the astronaut to perform the intended task.

No.	Description	Rationale
1	The simulator will need to help the astronaut in becoming familiar and better trained in a) service tasks such as replacement of processor or some other board of a laptop inside the Gateway b) the process of storage of cargo inside the Gateway	Such a simulator will allow the astronauts to practice the targeted activities in an intuitive manner as well as allow them to repeat those targeted activities until they are sufficiently trained
2	The simulators need to be user friendly and designed using HCI principles	Use of HCI principles will enable the astronauts to interact and be trained through user friendly interfaces, enhancing the effectiveness of the training process. The following HCI design heuristics will be followed. • Affordance • Ease of Interaction • Reduction of Cognitive Overhead
3	The simulators need to support camera based recognition of targeted physical objects as part of the cyber-physical interactions during training activities	Camera based recognition is necessary to identify objects in the physical environments (code) which in turn will serve as triggers to display the appropriate physical instructions for the astronauts to follow vis-à-vis the targeted service or other activities inside the Gateway.

6. TEAM 1 (RIDE)

The AR based environment for Laptop Repair scenario was implemented on the Vive Pro. The implementation part included the Virtual Reality(VR) / Augmented Reality(AR) mode. In the VR Mode the option for visualizing the full tutorial (Automated training scenario) and the step-by-step tutorial were included. In the AR Mode the module comprised of player interactions and Laptop replacement via player. The Layout of the modules in the simulation environment is indicated in the figure 6.1 to figure 6.4.



Fig 6.1. (left) Main Menu UI; Fig 6.2. (center) View of the options for the Augmented Reality Training Environment UI and Fig 6.3. (right) View of the Avatar based Automated overview of the training steps (developed by Team 1 RIDE)



Fig 6.4. (left) View of the menu options for step-by-step overview and Fig 6.5. (right) View of an Avatar showing the screw removal step for the step-by-step overview (developed by Team 1 RIDE)

7. TEAM 2 (PHOBOS)

The AR based Environment for Laptop repair scenario was developed and implemented using the HoloLens. The simulation environment is immersive and interactive. The different modules with Help Menu, Animations, Custom built 3D models and step by step instructions were included in the simulation environment as shown in figure 7.1 to figure 7.6.

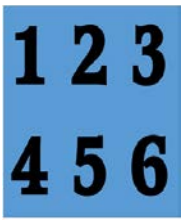
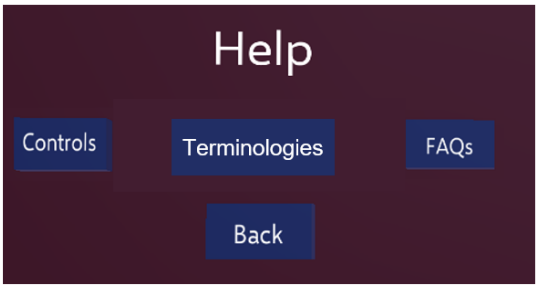
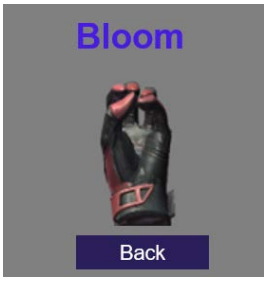




Fig 7.1. Number code to be scanned to initiate the training (left) and Fig 7.2. The view of the welcome scene after the number code is scanned (right)





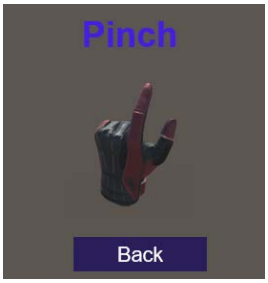


Fig 7.3. (left) Help menu options in the automated segment of the UIs (this is used to give the astronaut an automated overview of the entire set of service tasks they will perform later using the AR interfaces); Fig 7.4. (center) view of the finger based Bloom Gesture and Fig 7.5 (right) view of the finger based Pinch Gesture

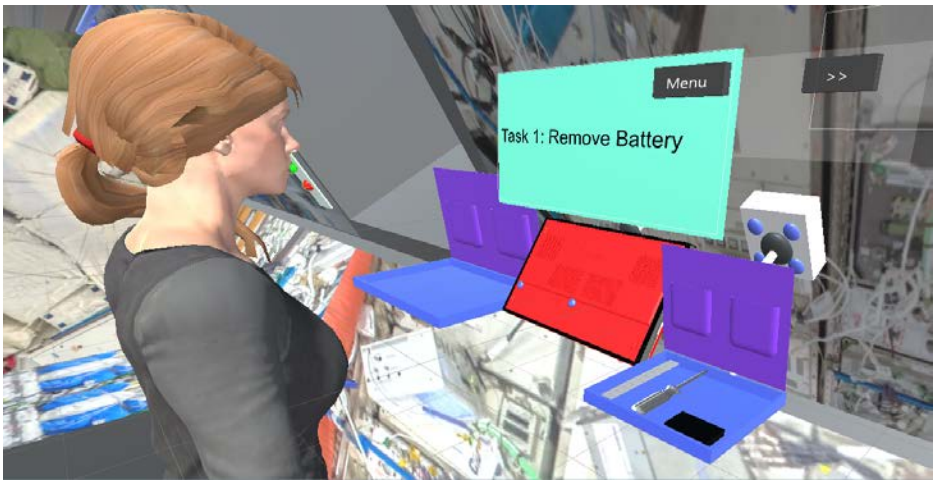


Fig 7.6. One of the options in the AR interface under “Act” menu options. The user sees the view shown in this figure where a human mannequin steps through the various tasks (one step at a time). While watching these tasks performed by the human mannequin, an astronaut or user can perform them in the real world.

8. TEAM 3 (ARTS)

The AR based environment for Payload Storage was developed and implemented in the HoloLens. The scenario consisted of three modules. Simulations and Animations module includes finding/scanning markers, bloom and Pinch Animations and loading/unloading Payloads. Marker

Recognition module included marker recognition which was fully functional and other Interacts with UI and displays panels properly. Help UI module included scripts written to Activate/Deactivate panels and Start/Stop Animations. The UI layout was solidified to a cleaner version with fully functional buttons and completed button designs facilitating easy navigation between menus as shown in figure 8.1 to figure 8.5.

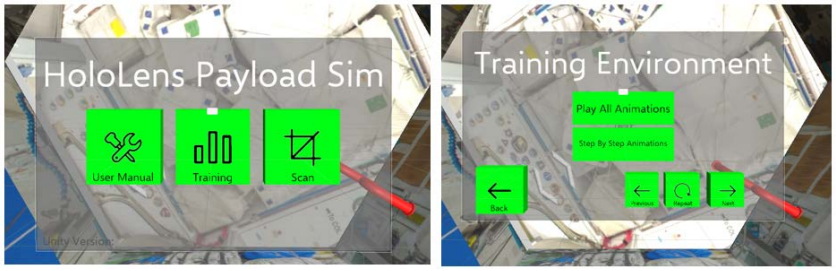


Fig 8.1 (left) Main Menu UI and Fig 8.2 (right) Training Environment UI (developed by Team 2 ARTS)

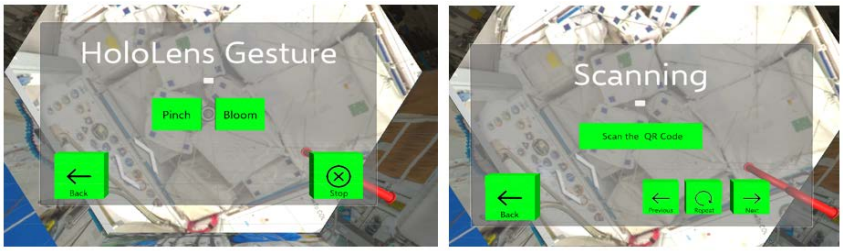


Fig 8.3. (left) Help UI to familiarize users with Pinch and Bloom gestures and Fig 8.4. (right) UI to scan the QR code (developed by team 2 ARTS)

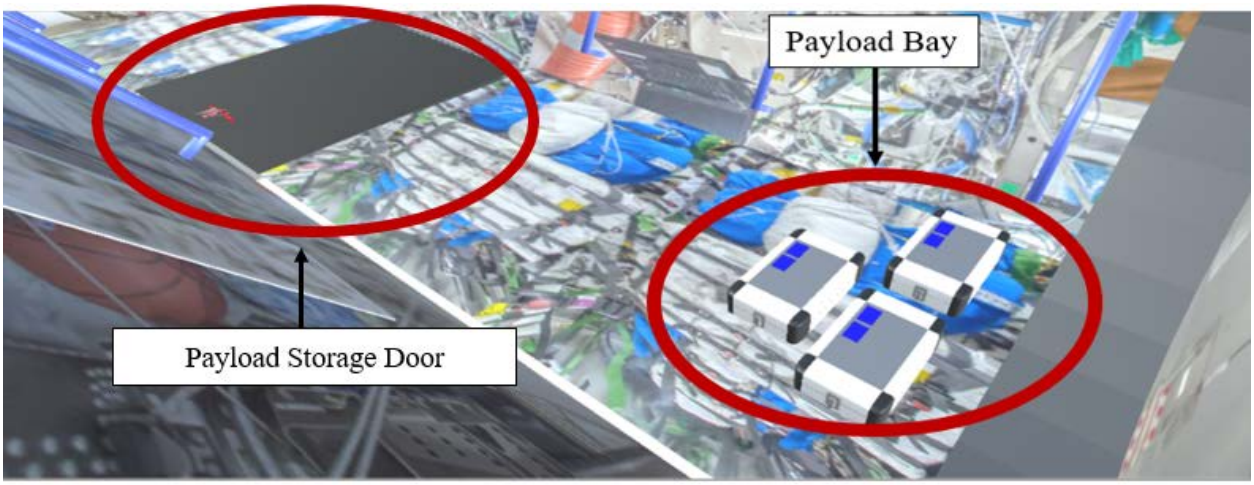


Fig 8.5. View of the AR based environment showing the location of payload and the storage (developed by Team 2 ARTS)

9. PHYSICAL VALIDATION ENVIRONMENT

Physical Validation Environments were created in the lab for both training scenarios. The following images show users following AR based instructions and interacting with the physical validation environment in figure 9.1 and figure 9.2.

Fig 9.1 A user following the AR based instructions and replacing the laptop component

Fig 9.2 A user following the AR based instructions and storing the payload

10. TESTING PLAN

Testing plan and test cases were built and performed to check the accuracy of the performance. The unit test case and functional tests were performed in the simulation environment. Testing plan was categorized into 3 levels as unit testing without humans in the loop, unit testing with human in loop and high level functional testing.

(i) Unit testing: without humans in the Loop

- On its own – check code (correctness criteria)
- Test to see if each of the interactions/code works (do inputs/outputs work?)
- Example (table) follows

(ii) Unit Testing: Human in Loop

- Test with humans in the interactions (human clicks on X, is selection read, is response correct? Correctness criteria)
- Example (table) follows

(iii) High Level: Functional

- Test to see if entire training simulation works (completeness criteria)

10.1. TEST CASES FOR UNIT TESTING AND FUNCTIONAL TESTING

The test cases were built based on the requirements, objective, procedure and the expected outcome of the test were documented. The unit test cases and the functional test cases are shown in figure 10.1 and figure 10.2 respectively.

S No	Objective (Based on Requirements)	Test Plan	Procedure	Expected Result
1	The simulator needs to **provide appropriate responses** for every user action [O1-6]	Test Plan for **Correctness of Code** [O2-6]	1. Wear the Headset 2. Look towards the number* code 3. A "start Simulation" button shows up 4. Click on the button using gesture 	1. **Number* Code Recognized** 2. **Gesture Recognized** 3. Training simulation begins
2	The simulator needs to **replicate the each physical laptop repair step** [O1-5]	Test Plan for **Completeness of Training Step** (Will be performed for each step individually) [O2-5]	1. Wear the Headset 2. Look towards the number* code 3. A "start Simulation" button shows up 4. Click on the button using gesture	1. **Button Press Recognized** 3. Appropriate step begins
3	The simulator needs to support the camera-based interface **to understand the trainee's gestures and provide appropriate response** [O1-3]	Test Plan for **Hand Interaction Capabilities** (HoloLens) [O2-3]	1. Wear the HoloLens Headset 2. Position your hand in front of the headset and make appropriate gestures (Test for all the gestures) 3. Test for satisfactory recognition of following gestures. 	1. **Gesture Recognized** 2. Appropriate step begins
4	The simulator needs **to support the camera-based interface to scan the number* Code** [O1-4]	Test Plan for **number* Code Support Requirement** [O2-4]	1. Wear the Headset 2. Stand near the object to be recognized	1.**Number* Code Recognized** 2.Appropriate step begins
5	The simulator needs to **support and assist users through text, voice and Avatar based instructions** [O1-2]	Test Plan for **User Support and Assistance** [O2-2]	1. Wear the Headset 2. Click on the Text, voice or Avatar based instructions buttons using the gesture 	**Appropriate Text, Voice or Avatar based instructions should assist** the user during the training process

Fig 10.1 Test Cases for Unit Testing

S.No	Objective (Based on Requirements)	Test Plan	Procedure		Expected Result
			Step	Description	
1	The simulator needs to satisfy all the functional requirements to repair the laptop using the AR environment	Test Plan for Integrated AR environment	1	Astronaut will wear the HoloLens	Successful
2			2	The Appropriate number* code is scanned to trigger a training sequence a. Processor Replacement Activity b. Payload	A Screen displaying the following options appear: 1. Overview 2. Training Module 3. Help Facility
3			3	Click on the following options using the gesture 1. Overview 2. Training Module 3. Help Facility	Appropriate response from each of the option is displayed
4			4	Click on the "Training Module" option using the gesture to initate the training.	Simulation of Battery removal starts automatically with voice/avatar/text based commands (Training Step - 1)
5			5	Click on the Next instruction button using the Gesture	Simulation of back panel screw removal starts automatically with voice/ avatar/text based commands (Training Step - 2)
6			6	Click on the Next instruction button using the Gesture	Simulation of back panel removal plays with voice/avatar/text based commands (Training Step - 3)
7			7	Click on the Next instruction button using the Gesture	Simulation of RAM upgrade plays with voice/avatar/text based commands (Training Step - 4)
8			8	Click on the Next instruction button using the Gesture	Simulation of placing back panel plays with voice/avatar/text based commands (Training Step - 5)
9			9	Click on the Next instruction button using the Gesture	Simulation of back panel screwing starts automatically with voice/avatar/text based commands (Training Step - 6)
10			10	Click on the Next instruction button using the Gesture	Simulation of placing the battery back starts automatically with voice/avatar/text based commands (Training Step - 7)
11			11	Click on the Next instruction button using the Gesture	Simulation is Finished successfully 16
12			12	Close the simulation	The simulation is closed successfully

Fig 10.2 Test Cases for Functional Testing

10.2 Testing Feedback

The testing was performed by each of the team on other team code and the outcome of the testing and the feedback were documented.

- Team RIDE tested Team ARTS code
- Team ARTS tested Team PHOBOS code
- Team PHOBOS tested Team ARTS code

Test plan and feedback of all the teams:

10.2.1 Team RIDE:

Summary of Test Plan

- First Phase: ensure that all necessary hardware is connected and working properly

- Begin testing with the reviewing the simulations and how they perform
- Check User Interface functionality and usability, this includes testing controls that will enable user input
- After all of the above has been tested several complete runs of the training are completed to find rare bugs and glitches

Test Results and Actions

- Team PHOBOS tested Team RIDE AR simulator
- Resulted in several bug fixes
- Help menu and VR/AR based training modules are functioning properly
- Simulations are in correct sequence
- Initial feedback: Voice instructions appear to be clear and concise
- Controllers were functioning as intended

Feedback and test findings/comments after testing

10.2.2 Team ARTS:

- Team RIDE tested Team ARTS program: *
  - Note they were provided the non-immersive version due to their lack of access of the HoloLens
- Hardware Testing - Team Ride did not have access to a HoloLens, so all testing was performed only on the non-immersive version of the interfaces (on Unity) and not involving the HoloLens platform
- Animation/Scene Testing: All components of the animation/simulation were clear and worked correctly
- UI Testing: everything worked as it should, and no bugs or errors were found.
- Camera Dimensions: All dimensions enabled user to see necessary details in the simulation
- Animations /Controls Testing: no obvious tweaking after testing all animations and simulations, which responded to various button options. Controls were not able to be tested through HoloLens; however, all buttons and control options tested using the Key Board did not yield any bugs
- QR Code Testing: Unable to test due to it being included in the HoloLens portion of the test files

10.2.3 Team PHOBOS:

- Number Code Recognition: Number Code Recognition works as intended. Suggestion: A different Icon for the Start Animation is recommended.
- UI Testing: Sub-module Testing in Menu
  - Help menu

- All submodules were presented accordingly.
  Suggestion: Instead of creating different scenes with different UI panels, create all the UI panels in the same scene and enable and disable them as required. This would decrease the overall project file size.

- Watch menu
  - The oral cues were not being executed (some problem here; please fix this)
  - Animations appeared well designed and carefully planned.
  - Problem: Regarding the automated help recording to be watched prior to the actual AR based task accomplishment, the video seemed too blurry at the beginning; recommend increasing the resolution to 1080 pixels.
- Act menu
  - All Tasks were in correct order.
  - Problem: Found a certain problem: something is wrong when the next button is selected; the simulation/animation was not smooth and it appeared jagged as if it was starting and stopping (possible bug, please fix this).

The Teams made modifications according to the feedback provided (listed above).

11. RISK MANAGEMENT

In developing an AR/VR solution, risks were identified that could hinder project deadlines and milestones. In order to give a visual representation, we chose to use a Risk Matrix. Each item in the Risk analysis has a description, state, category, impact, probability, consequence, priority and a mitigation plan. In order to generate the Risk Matrix some calculations were made from following columns using available data and/or best guess.

- Impact: Is a weighted average of the items impact on cost, scheduling and technical.
- Probability: Is the likelihood the risk will occur at any given time.
- Consequence: Is how the item will impact the project over time.

The calculations resulted in a priority, which takes the items probability and consequences and averages them to give a result.

S.No	Risks		Normalized Priority	Mitigation Plan
1	HoloLens has occasional problems with battery and recharging		1.00	● Purchase a new HoloLens (next version) ● Budget limitations may preclude this
2	Finger gesture recognition failure in HoloLens		0.04	●Properly training the users in operating the HoloLens ●Testing the application using gestures in multiple angles
3	Risks in employing such systems in space	Vive platform requires trackers mounted on walls/stands to track the position of headset/controllers. Cluttered	0.72	Choose the location of trackers such that it is clearly visible in the Gateway

		environment of Gateway may hinder the tracking		
		Use of AR system (HoloLens/Vive) in the Gateway places additional constraints on power. Battery power may be consumed leading inability to use the AR system	0.75	Keep the instructions in the AR environment concise and clear such that the tasks are completed before the battery power is consumed
4	Analog model (validation environment) is not a 100 % replication of the gateway model		0.31	● Replicate essential design elements ● Obtain additional research material from NASA on the gateway structure ● Obtain feedback from NASA engineers
5	Discomfort in using HoloLens/VIVE for extended periods of time.		0.80	● Minimize eye strain (reduce flashes, minimize sharp motion etc.) ● Usage of required HCI principles
6	Objects may not be recognized during simulation [wear and tear of the QR codes, code failure, camera malfunction]		0.60	● Rigorous testing (coming up with foolproof mechanisms that would handle all the undefined cases) ● Safe handling of cameras ● Protective layer on the Code
7	Lack of software support (HoloLens, Vive)		0.81	● Perform necessary research (on software toolkits, libraries, plugins) beforehand ● Tight communication between design, build and test teams ● Loose coupling among the modules to allow changes without affecting multiple areas
8	project behind schedule		0.55	● Effective project planning and management ● Project Scheduler coordinates various activities based on project schedule, implementing Scrum and regular meetings with stakeholders

Risk Matrix:

A risk matrix is used during risk assessment to define the level of risk. A risk matrix has a horizontal axis which indicates the consequence, impact over time, and a vertical axis indicating the probability as shown in figure 11.1. Inside the matrix there are color coded boxes. These colors

represent the priority of the item(s) contained. Green being low priority, Yellow being moderate priority and Red being high priority.

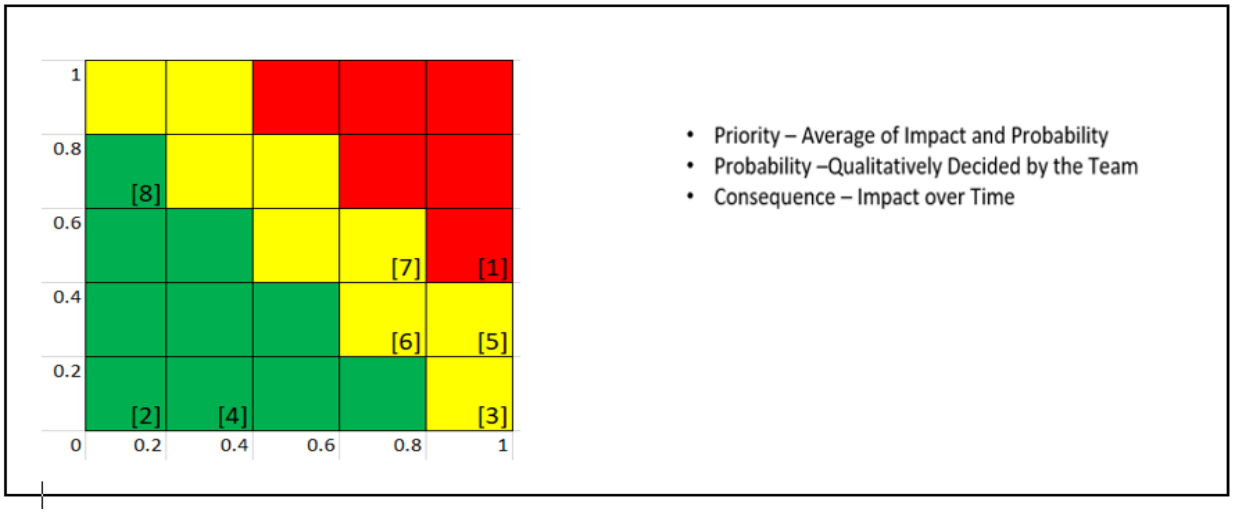


Fig 11.1 Risk Matrix

Risk Identified during the project included:

- HoloLens had occasional problems with battery and recharging.
- Finger gesture recognition failure in HoloLens.
- Risks in employing AR/VR systems in space.
- Use of AR system (HoloLens/Vive) in the Gateway places additional constraints on power. Battery power may be consumed leading inability to use the AR system.
- Vive platform requires trackers mounted on walls/stands to track the position of headset/controllers. Cluttered environment of Gateway may hinder the tracking.
- Astronauts may not be used to wearing a headset for long periods of time; it may take some time to get used to wearing it and doing their usual tasks.
- Use of AR system (HoloLens/Vive) in the Gateway places additional constraints on power. Battery power may be consumed leading inability to use the AR system
- Analog model (validation environment) is not a 100 % replication of the gateway model
- Discomfort in using HoloLens/VIVE for extended periods of time.
- Objects may not be recognized during simulation (wear and tear of the QR codes, code failure, camera malfunction)
- Lack of software support (HoloLens, Vive)
- Project behind schedule

12. VALIDATION AND ASSESSMENT

A Study was conducted to compare the HoloLens and Vive based AR environments. Six users interacted with the training simulators and Cyber Physical Analysis of the training simulator was made. Some criteria of the assessment were based on NASA TLX test.

- Mental, physical and temporal demand
- Performance
- Effort
- Frustration

The following tables shows the user assessment based on Modified NASA TLX Survey:

1. Assessment for different simulators on the same platform

Criteria (Rating between 1- 10)	HoloLens Laptop		HoloLens Storage	
	Mean	Variance	Mean	Variance
1. Clarity of the instructions in the AR platform.	7.5	2.8	7.3	3.7
2. The degree of usefulness of the 3D avatar-based interactions.	8.6	0.7	8	1.5
3. The degree of usefulness of the voice-based interactions.	7.8	0.2	8.6	1.2
4. The degree of usefulness of the text-based interactions.	8.3	1.2	8	1.5
5. The ease of completing tasks while wearing the headset.	7.6	0.7	8	1.0
6. The ease of navigating through the UIs	8	0.5	7.8	0.7
7. The overall usefulness of the UIs.	8.9	0.5	9	0.5
8. Effort: How hard did you have to focus in order to complete the training? (1:Low, 10:High)	4.4	5.6	3.3	3.2
9. Frustration: How frustrated were you when trying to complete the training? (1:Low, 10:High)	2.1	1.4	2.0	1.7
10. Mental Demand: How mentally demanding was the training? (1:Low, 10:High)	4.3	11.1	3.5	8.5
11. Physical Demand: How physically demanding was the training? (1:Low, 10:High)	4.5	6.75	3.6	8
12. Temporal Demand: How hurried or rushed were you during the training? (1:Low, 10:High)	1.5	0.75	1.3	0.8
13. Which is more responsive in a reliable way when interacting with the AR based UIs? Finger gestures or controllers.	9.4	1.36	9.4	1.36
14. Which is more intuitive/natural when interacting with the AR based UIs? Finger gestures or controllers.	6.8	0.24	6.8	0.24

2. Assessment of same simulator on different platforms

Criteria (Rating between 1- 10)	HoloLens Laptop		Vive Laptop	
	Mean	Variance	Mean	Variance
1. Clarity of the instructions in the AR platform.	7.5	2.8	7	2.8
2. The degree of usefulness of the 3D avatar-based interactions.	8.6	0.7	8.2	2.56
3. The degree of usefulness of the voice-based interactions.	7.8	0.2	7.6	1.04
4. The degree of usefulness of the text-based interactions.	8.3	1.2	7.4	0.24
5. The ease of completing tasks while wearing the headset.	7.6	0.7	8	2
6. The ease of navigating through the UIs	8	0.5	7.8	2.16
7. The overall usefulness of the UIs.	8.9	0.5	8.6	0.24
8. Effort: How hard did you have to focus in order to complete the training? (1:Low, 10:High)	4.4	5.6	4	8
9. Frustration: How frustrated were you when trying to complete the training? (1:Low, 10:High)	2.1	1.4	2	1.6
10. Mental Demand: How mentally demanding was the training? (1:Low, 10:High)	4.3	11.1	3.4	4.24
11. Physical Demand: How physically demanding was the training? (1:Low, 10:High)	4.5	6.75	4	6.4
12. Temporal Demand: How hurried or rushed were you during the training? (1:Low, 10:High)	1.5	0.75	1.6	0.64
13. Which is more responsive in a reliable way when interacting with the AR based UIs? Finger gestures or controllers.	9.4	1.36	9.4	0.64
14. Which is more intuitive/natural when interacting with the AR based UIs? Finger gestures or controllers.	6.8	0.24	6.8	0.16

Based on the modified NASA TLX test, 6 users were asked to provide feedback and the main outcomes reported are:

- The 3 simulators (2 running on the HoloLens and 1 on the Vive) appeared comparable in most of the categories
- One of the areas in which there was a clear difference had to do with reliability of the interfaces: for example, can finger gestures be recognized by the simulator system? The HoloLens based finger gestures were not as reliably responsive in terms of recognizing the commands when compared to the controller based commands in the Vive. However, the users preferred the intuitive manner in which the finger based gestures supported the interactions (when they worked).

- In terms of clarity of instructions, usefulness of Avatar and text instructions and navigation, users rated the Laptop repair interactions slightly higher than the 2 AR environments
- In terms of voice based instructions and overall usefulness, users rated the Payload storage interactions (on the HoloLens) slightly higher than the two AR environments
- Both HoloLens based payload storage AR environment and Vive based Laptop repair AR environment were equally good in terms of ease of completing the tasks
- The level of effort and frustration were low for the three environments; however, they were slightly higher for HoloLens based Laptop repair AR environment

Example Feedback:

1. Which is more responsive in a reliable way when interacting with the sim environment? Rate this 1 to 10: 10 is high (1 is low)

HoloLens: 5

Vive: 8

I chose the answers because first off for those that haven't used the HoloLens they won't be able to grasp how to properly perform the gesture. 8 on the Vive because it's literally an input via the controller only reason that it's not responding would be the code

2. Which is more intuitive/natural when interacting with the sim environment? Rate this 1 to 10: 10 is high (1 is low)

HoloLens: 10

Vive: 6

I chose the answers first off 10 for the HoloLens because that's a natural motion and it's easy to pick up on and doesn't require another device like the Vive controller. I picked 6 on the Vive because once we have used the HoloLens I personally wouldn't want to go back to the clunkiness of the controller. It just doesn't feel natural but for a newbie it's the perfect tool to interact with the sim environment.

Comparison of the Two Platforms Vive and HoloLens– Feedback from the Students:

1. Portability and Physical Setup
    - HoloLens was more portable compared to Vive (no trackers/antennas which the Vive requires for operation)
    - HoloLens is light in weight and is untethered
    - HoloLens requires minimal setup for running a completed simulation
    - The Vive is tethered
    - Vive requires the trackers to be initialized and setup for Vive before running a simulation

2. Power
    - HoloLens is battery operated and Vive is tethered (power is through cables)
    - Note: There was a power related battery issue with the HoloLens in the lab
    - HoloLens is an older product (not sure if that was the reason why power problems were encountered)
    - It would not ‘turn on’ sometimes even it was fully charged
    - Sometimes it would turn on after charging it continuously for a day or two
3. Complexity of Simulation
    - HoloLens is powered by an inbuilt processor (the executables have to be run on this processor, which resides or is stored within the 3D headset)
    - The Vive headset is linked to an external computer, where the simulation executable resides. This allows the platform to handle more complex simulations.
4. Interaction with real world
    - When using HoloLens, users can view the real world and the virtual world (this can be termed as Mixed Reality)
    - In the Vive, a set of cameras display the real world onto the 3D headset
    - The HoloLens provides a better real world view when compared to the Vive
5. Responsiveness and Intuitiveness of Interfaces in the two platforms to support interactions
    - HoloLens uses a gesture based interface (gestures made by user’s fingers)
    - Some students had difficulty with having their gesture’s recognized
    - With practice, this problem was overcome
    - The Vive’s interactions are accomplished by a gaming type of controller; this is very reliable and easy to use
    - However, using the finger based gestures appears to be more intuitive
6. Reading QR or numerical codes for cyber-physical interactions
    - HoloLens platform supported recognition of QR and other codes with the help of an external Plugin
    - We were not able to find a plug-in to support such QR or numerical code recognition for the Vive
7. Development of the Simulation environment including coding

- Both the platforms were comparable in terms of programming and development support (they both supported creation of 3D simulation and user interfaces based on Unity 3D engine)

8. Debugging and Testing

- For the HoloLens: The simulation executable had to be uploaded from the computer to the HoloLens headset, which made debugging and testing cumbersome

  (Note: There is a HoloLens emulator available. However, it is not user friendly and does not support all the features of HoloLens)

- For the Vive: as the headset is connected to the host computer, it was easier to complete debugging and testing activities

13. DISSEMINATION

- AIAA conference [Virtual Conference] – June 15- 19, 2020
- IEEE Syscon conference [Virtual Conference] – Aug 24 – Sept 20, 2020

(Note: One poster has been presented on WIP at Sigma Xi Conference - Nov 16,2019 at Madison)

Hochberg, Lynzi; Zimmer, N., Skinner, T., Murakonda, N., Gupta, A., Cecil, J., X-HAB: Exploring AR/VR Simulation-based Design Approaches to Study Design Alternatives for the Gateway and Lunar Mission, Poster Paper Presentation, 2019 Sigma Xi Student Research Conference, November 14-17, 2019, Madison, Wisconsin

Cecil, J., Krishnamurthy, R., Kennison, S., Gupta, A., Murakonda, N., Lopez, A., Crosby, N., Zimmer, N., Venkatramanan, K., Bhakta, M., Skinner, T., Olivero, S., Walston, H., Miftah, M., Design of Next Generation User Interfaces based on Augmented Reality technologies to train astronauts on the Gateway in support of the Moon Mission, 2020 AIAA AVIATION Forum (virtual conference), June 15-19.

Cecil, J., Krishnamurthy, R., Sweet-Darter, M., Gupta, A., Role of Immersive Simulation and Cyber Technology based approaches in supporting Learning and Curriculum Innovation. 14th Annual IEEE International Systems Conference (virtual conference moved from April to Fall 2020).

- Scaling down simulator content for special outreach involving autistic students
  i. Interacting with Dr. Mary Sweet-Darter (psychologist/autism therapist)
  ii. Based on X-Hab AR environments, Virtual Learning Environments created for middle and high school autistic students
  iii. Science and engineering Concepts: Assembly, assembly plan, sequence, obstacles

14. FORMATIVE ASSESSMENT

Formative Assessment conducted by Dr. Cecil indicated:

1. Some students were behind in software skills especially AR model building skills using unity and CAD modeling using sketchup.
   - How was this addressed:
     - Additional help provided
     - One on one interactions by lab assistants
2. Tulsa students needed access to immersive AR platforms.
   - How was this addressed:
     - Obtained permission and new lab space for project was obtained
3. Students were enthusiastic about Moon Mission context.
4. Some teams were having problems with finding common time.
   - How was this addressed:
     - Instructor met with students to help find common time for project work
     - Weekly updates and discussions became requirement
     - Virtual meetings were encouraged
5. Covid-19 virus outbreak placed a limit in how many students could be in lab at same time.
   - How was this addressed:
     - Some students were given permission to take the Vive system home
     - Other students came to the lab different times to ensure physical distancing was practiced
6. HoloLens platform was giving problems.
   - How was this addressed:
     - Kept project managers informed
     - Adopted short term fix, which was to charge the HoloLens continuously

15. SUMMATIVE ASSESSMENT

Apart from the formative assessment, summative assessment was also conducted which indicated.

a. Students' performance in homework and project has improved over past years.
b. Students' engagement (obtained through feedback/ survey) improved over previous years.
c. Students' model building skills, system engineering and software engineering principles improved through involvement with X-Hab project.
d. Students are enthusiastic about disseminating results at 2020 AIAA virtual conference (June 2020).
e. Greater awareness and interest among students towards space systems and NASA related careers.

16. CONCLUSION

An overview of the X-Hab project activities including the requirements and design process resulting in the creation of Augmented Reality based User Interfaces was provided in this report. The design diagrams were based on the Unified Modeling Language (UML). Systems engineering

methods were adopted based on the IDEF-0 modeling language. Descriptions of the AR based user interfaces for the payload transport and laptop assembly tasks created by each team has been provided. The report also includes comparison of the platforms, the testing activities and results of assessment activities. Finally, a summary of the dissemination activities was also included.

ACKNOWLEDGEMENT

Special thanks to Dr. Lauren Underwood (NASA Stennis) and Dr. Jeremy Frank (NASA Ames) for their involvement in this project through feedback and encouragement of student teams. OSU project team would also like to thank the NASA X-Hab program for funding this project. Some of the undergraduate lab assistants were funded by NSF as part of a Research Experiences for Undergraduates (REU) site project.

17. REFERENCES

[1]. library.vuforia.com/content/vuforia-library/en/articles/Solution/Working-with-the-HoloLens-sample-in-Unity.html

[2]. developer.vive.com/resources/knowledgebase/intro-vive-srworks-sdk/

[3]. Tang, Y. M., Au, K. M., & Leung, Y. (2018). Comprehending products with mixed reality: Geometric relationships and creativity. *International Journal of Engineering Business Management*, *10*, 1847979018809599.

[4]. https://circuitstream.com/blog/htc-vive-tutorial-interactables/

[5]. Hart, S. G. (2006, October). NASA-task load index (NASA-TLX); 20 years later. In *Proceedings of the human factors and ergonomics society annual meeting* (Vol. 50, No. 9, pp. 904-908). Sage CA: Los Angeles, CA: Sage publications.

[6]. https://nasa3d.arc.nasa.gov/models