AHUD

Augmented Heads-Up Display

NASA SUITS Design Challenge 2019-2020

Chapman University

Team Luna

Team Members

Clarisse Bonang: Design & Graphic Artist
bonang@chapman.edu /Junior/ Computer Science

Dana Davidson: Design & Programmer
ddavidson@chapman.edu / Junior / Computer Science

Lisa Pham: Design & Graphic Artist & Programmer
lisaphcm@gmail.com / Sophomore / Computer Science

Michelle Yoon: Design & Project Manager
myoon@chapman.edu / Junior / Computer Science

Advisors

Christopher Boyd: Professor at the Fowler School of Engineering at Chapman University

Gotthard Janson: NASA Point of Contact & Senior Engineering Specialist

Contents

1 Project Timeline **3**
1.1 Phase 1 3
1.2 Phase 2 3

2 Abstract **4**

3 Acknowledgment **4**

4 Introduction **4**

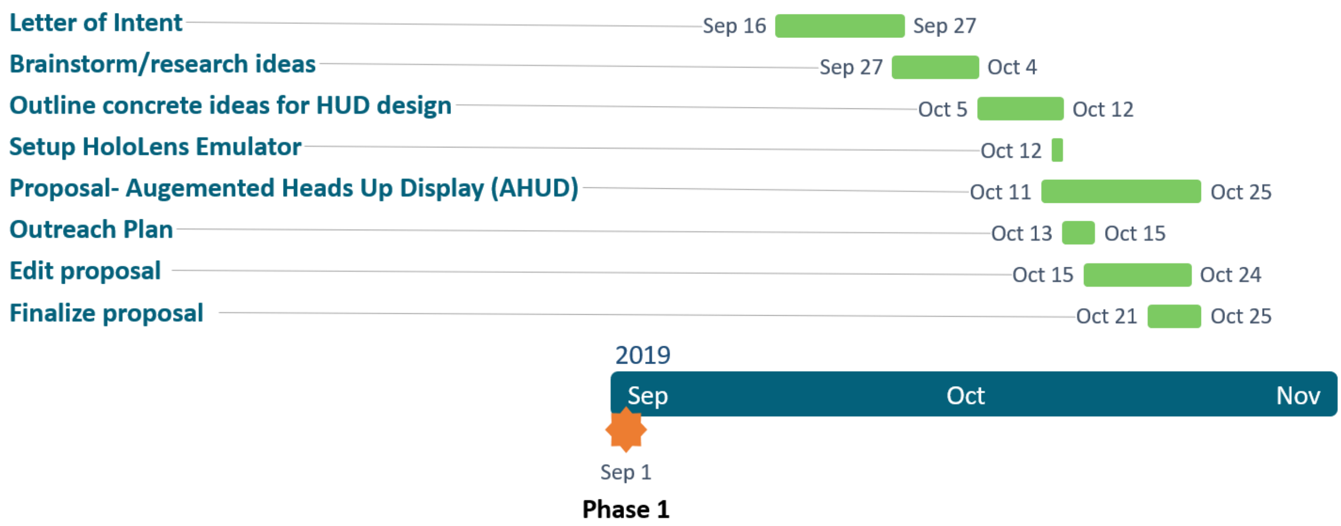
5 Technical Summary **5**
5.1 GUI Design 5
5.1.1 Starter Menu 5
5.1.2 Subtopic Menu 5
5.1.3 Instructional Slides 6
5.2 Software Framework 6
5.3 Telemetry Stream 8
5.4 Hardware 8
5.5 Software Testing 8
5.6 Software Study Design 9
5.7 Testing Measurements 10

6 Outreach **10**

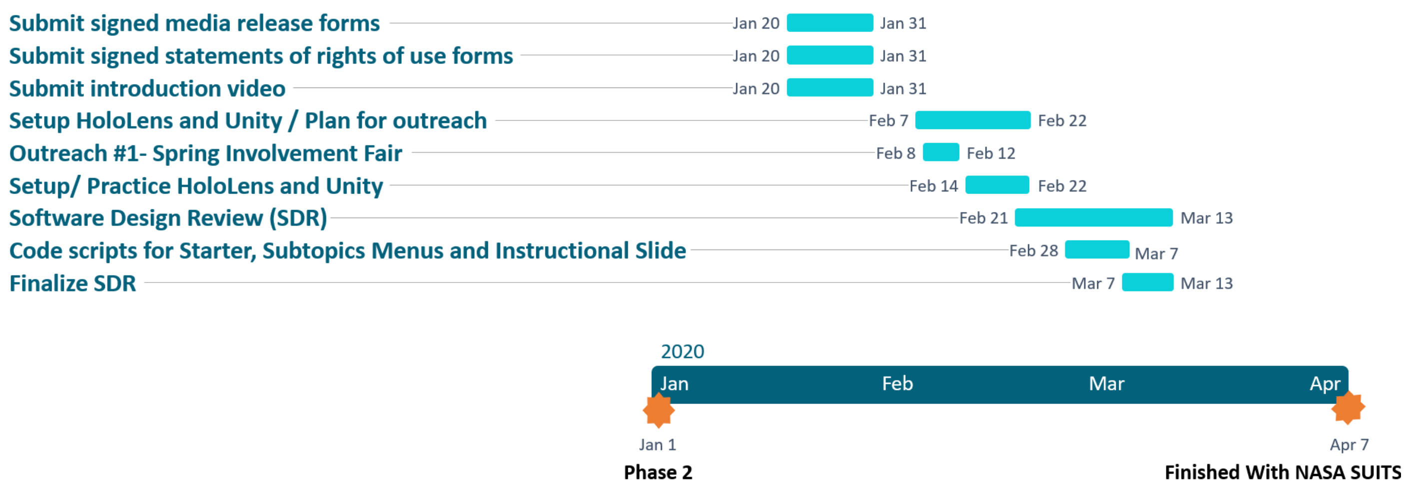
7 Summary **11**

1 Project Timeline

1.1 Phase 1



1.2 Phase 2



2 Abstract

The Augmented Heads-Up Display (AHUD) was designed for assistive use in Extravehicular Activities (EVAs). It uses Augmented Reality (AR) that aims to provide a visual information system to assist astronauts with tasks during their missions by presenting instructions and other vital information. AHUD takes inspiration from video games heads-up displays (HUDs) to create a simplistic and non-obtrusive layout on the Head Mounted Display (HMD). Within the user interface, it will display all types of EVA instructions and/or documentations to help astronauts during their tasks.

3 Acknowledgment

Team Luna would like to thank NASA SUITS Mentors, Gotthard Janson and Christopher Boyd for their expertise, input, and support of AHUD design for the NASA SUITS design challenge.

4 Introduction

Our main goal was to design a UX-framework that would help create an intuitive assistive application that displays all types of instructions and/or documentation to assist with EVA tasks. Uses can range from displaying detailed instructions for a rover tire change to displaying a geology chart as a rock sampling reference.

5 Technical Summary

5.1 GUI Design

5.1.1 Starter Menu

The AHUD's starter menu displays square application button in a grid like formation that is separated by topics. Each topic represents an EVA task that the user can choose from to display instructions and/or documentations relating to the task at hand. On the bottom of the starter menu, there is a home button which will take the user back to the starter menu

.

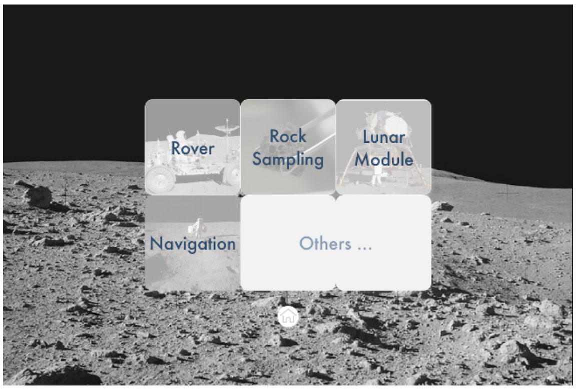


Figure 1: Starter Menu

5.1.2 Subtopic Menu

The AHUD's subtopic menu is followed after the starter menu. Each subtopic is divided up by a specific issue that might occur during the EVA task chosen the from the starter menu. In addition, on the right side of the subtopic menu, there is a scroller that the user can use to view other subtopics.

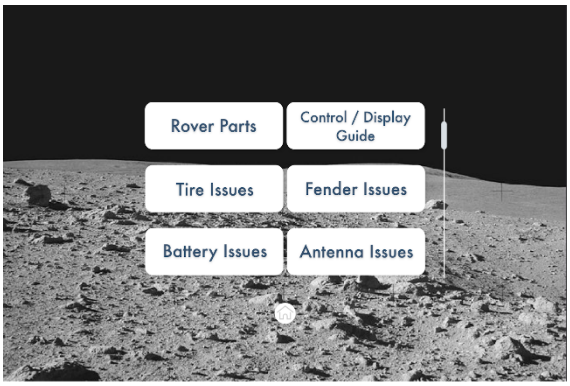


Figure 2: Subtopic Menu

5.1.3 Instructional Slides

The AHUD's instructional slides display both the pictures and its associated instructions in order. On the bottom of the starter menu, there is a home button which will take the user back to the starter menu.

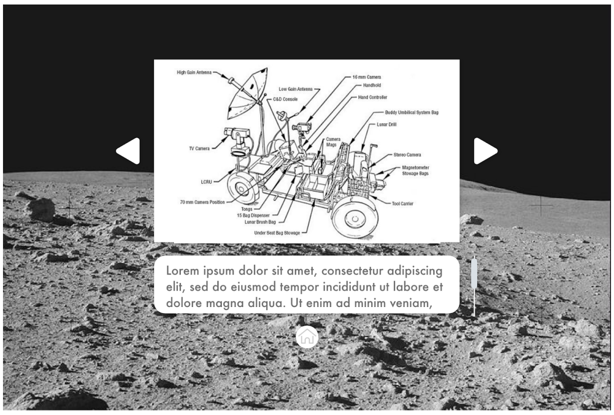


Figure 3: Instructional Slide

5.2 Software Framework

1. Frontend (Client-Side): Responsible for what the user can see and interacts with on the application.
   (a) GUI Design
      i. Starter Menu
      ii. Subtopic Menu
      iii. Instructional Slide
      iv. Home Button
      v. Navigational Buttons
2. Backend (Server-Side): Responsible for storing and organizing data and ensuring everything on the Frontend works.
   (a) Scripts
      i. SceneMenu
      ii. InstructionalMenu
      iii. SlideMenu
      iv. HomeButtons
      v. SlideNext

(b) JSON Package

   i. The data packages are received initially through a JSON package and then delegated to each of their respective scripts through a File I/O handler.

(c) SlideDeck Package

   i. Compressed file of pictures and respective instructions to be sent and allocated to slides

(d) Upload Prior/ Stream

   i. To be received by the FileIO Script, the SlideDeck can either be “uploaded prior” to the mission by manual entry, or through the “stream” if the mission is ongoing

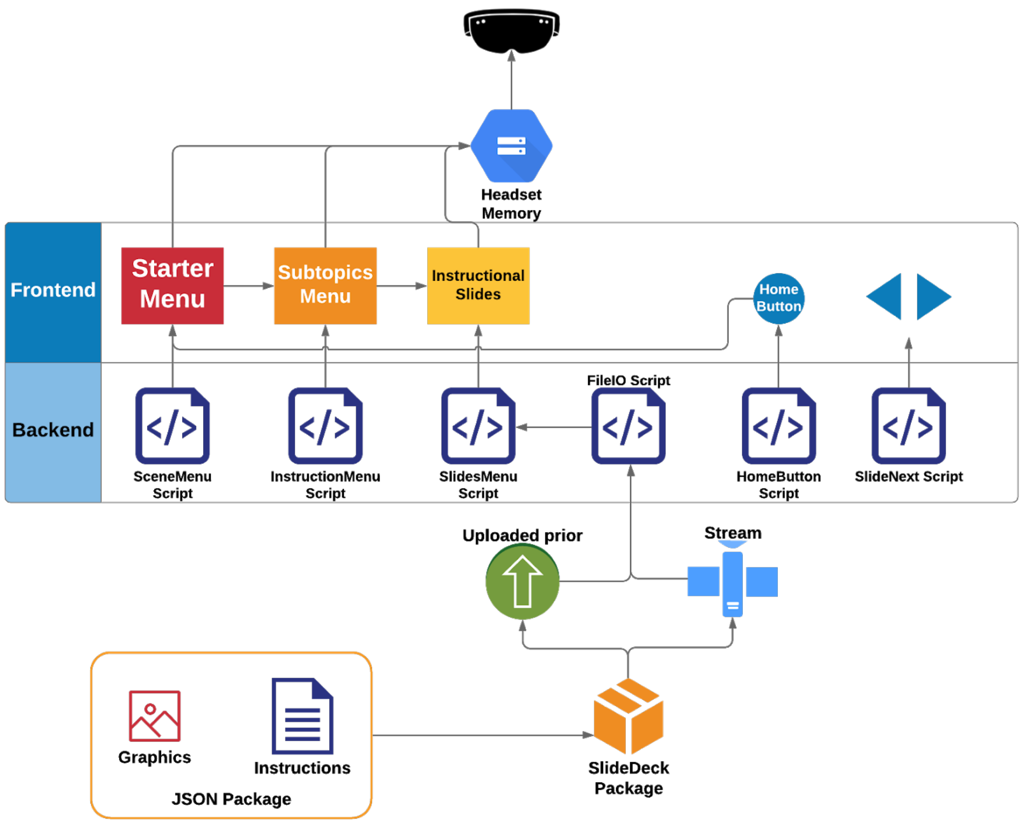


Figure 4: Software Framework

5.3 Telemetry Stream

1. Implementation and Handling
   (a) The telemetry stream will be assumed to be in JSON format. Data will be read at regular intervals for evaluation of both current values and variations from the past X values, where X varies by data point.
      i. Variations of specific ranges and over Y evaluations will lead to warnings (more on the Reliability section).
   (b) Standard JSON libraries associated with .NET development such as the built-in libraries and Newtonsoft library will be used for implementation. A custom Unity plugin is being developed for handling JSON data by LUNA.
2. Reliabilty
   (a) Failures, abnormalities, warnings, and other issues will generate warning messages to the astronaut and, where applicable, mission control.
   (b) The telemetry system will rely upon a ping system that will gather data every 3-5 seconds. Consecutive failed pings will generate escalating warnings to the astronaut and mission control.
3. Warnings and Anomalies
   (a) Telemetry from on-board systems will be monitored for values that fall outside of their respective "healthy" ranges, whether technical, medical, etc.
      i. Consecutive and gradually escalating results outside of the standard range will result in warnings.
   (b) Warnings will be conveyed to the astronaut via their display as well as relayed back to mission control.
   (c) Additional abnormalities will be monitored, but will report to the astronaut display first for real-time evaluation.

5.4 Hardware

For testing and development, the HoloLens (1st gen) was used, with approval for 2nd gen equipment when it is available for general purchase.

5.5 Software Testing

1. Software testing will be composed of two parts, functional testing and nonfunctional testing.
2. Functional Testing
   (a) Unit Testing
      i. Validate that each unit/ components of the software code performs as expected.

(b) Integration Testing

i. Test combinations of components together to see if those components interact properly.

(c) System Testing

i. Test behavior of different components interacting with one another.

(d) Acceptance Testing

i. Determined whether the software system has met the requirement specifications.

3. Non-Functional Testing

(a) Performance Testing

i. Check if system responds quickly enough so there is no delay in astronauts performing tasks. If any system latency is shown, all EVA activities might not be completed.

(b) Reliability Testing

i. To prevent system crashing or failing intermittently during the EVA task.

(c) Usability Testing

i. If the system is difficult to use, it could prevent an astronaut from accomplishing all EVA activities. To mitigate any usability difficulties, utilize the observation record tickets to evaluation.

5.6 Software Study Design

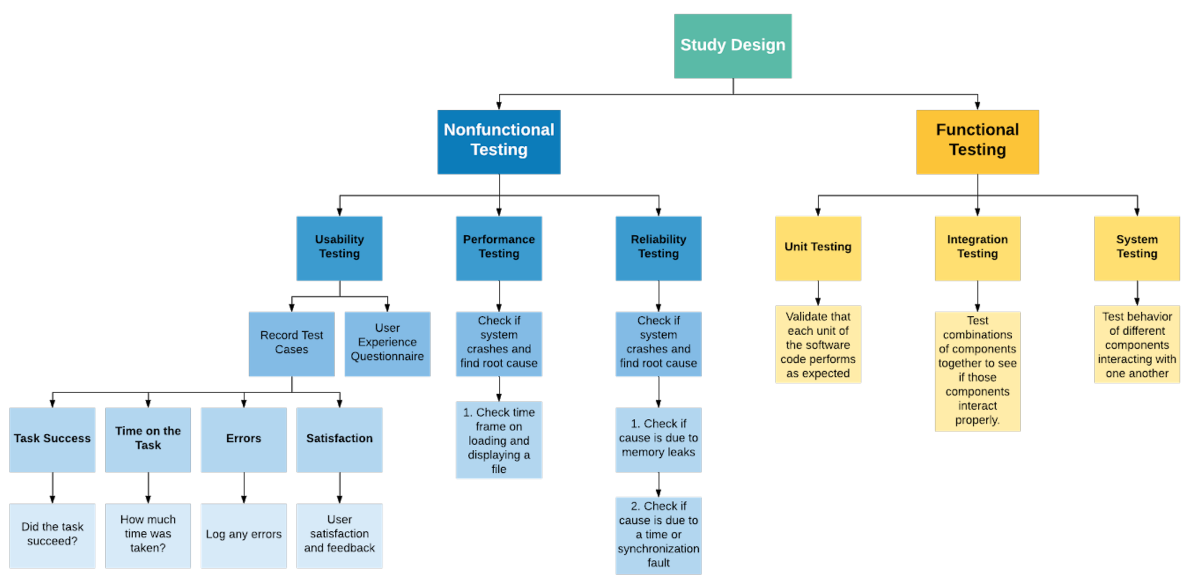


Figure 5: Software Study Design

5.7 Testing Measurements

1. To prevent any system latency in all EVA activities and better software performance, testing will include measurements of the following variables below to identify areas of improvement. Testing measurements will be categorized into two groups: Quantitative and Qualitative.

2. Quantitative Examples

   (a) Time: Measure the time frame it takes to open the application

   (b) Measure the time frame it takes to open a file from either the rover or rock sampling category

   (c) Measure the time of stream for displaying diagrams and photos

   (d) Measure the time on the task: the length of time it takes the participant to complete to the task

   (e) Errors: record count of errors each participant make in their task

3. Qualitative Examples

   (a) Task success: Check how accurately user operates intended functions of software

   (b) Visuals: Check if desired GUI is displayed

   (c) Functionality: Record functions that perform and don't perform desired execution

   (d) Satisfaction: record user's satisfaction- participants overall feeling about the application during and after their test. Given a questionnaire.

6 Outreach

Following the projected timeline in Luna's initial proposal of outreach, Luna applied as an official Chapman club in December and in January, Luna was approved under the name Chapman STEAM RD. The club's purpose was to promote various team-based and individual STEAM (Science, Technology, Engineering, Art, Math) challenges and competitions hosted by organizations in a variety of industries throughout the year so members would strengthen and hone their skills in STEAM-related fields, experience cross-departmental teamwork and enhance career opportunities. Luna also reached out and started working with other clubs in order to reach a wide audience.

At this time, team members also created an Instagram account for Luna, posting promotional content about the project, the team, their progress, and outreach events. In February, Luna's main outreach event was participating in the Spring Involvement Fair at Chapman University. Team members talked with peers and faculty about NASA SUITS, their experience, what the team had accomplished so far and their plans moving forward. Luna also put together a small flyer of upcoming local STEM competitions and challenges similar to SUITS for students. Finally, a list of interested students and their contact information was collected for future outreach.

In March, Luna was invited by the Chapman STEMtor club to participate in Discover STEM Day, where the community would come to explore the possibilities of STEM fields at many interactive stations led by professors and students. The planned booth included an AR experience with HoloLens and 3D printing and modeling demos, with small tokens for visitors to take as souvenirs. Unfortunately, out of an abundance of caution because of the Coronavirus pandemic, this event was canceled per Chapman University's recommendation.

With the escalating number of Coronavirus cases, all Luna's future in-person outreach events were considered canceled or indefinitely postponed.

Figure 6: Outreach- Spring Involvement Fair

7 Summary

The Augmented Heads-Up Display (AHUD) was designed to aid astronauts during their Extravehicular Activities (EVAs). It uses Augmented Reality (AR) that aims to provide a visual information system to assist astronauts with tasks during their missions by presenting instructions and other vital information. AHUD is a UX-centric framework that would help create an intuitive assistive application that displays all types of instructions and/or documentation to assist with EVA tasks. Uses can range from displaying detailed instructions for a rover tire change to displaying a geology chart as a rock sampling reference. The project was developed in Unity, a cross-platform game engine and utilized the Hololens to create a virtual and interactive program.

In the AHUD's GUI framework, there is the starter menu, subtopic menu, and the instructional slides. The starter menu consists of a grid formation, split up by separate topics. The subtopics menu follows after the starter menu and is composed of topics related to the original broad topic. Each subtopic represents a particular problem that might be encountered and consists of instructional information and/or documentation. The instructional slides display both the graphics and its associated instructions in order.

Within the telemetry stream, in the implementation and handling component, data will be read at regular intervals for evaluation of both current values and variations from the past X values, where X varies by data point. Variations of specific ranges and over Y evaluations will lead to warnings. In the reliability component, failures, abnormalities, warnings, and other issues will generate warning messages to the astronaut and, where applicable, mission control. The telemetry system will rely upon a ping system that will gather data every 3-5 seconds. Consecutive failed pings will generate escalating warnings to the astronaut and mission control. In the warning and anomalies component, telemetry from on-board systems will be monitored for values that fall outside of their respective “healthy” ranges, whether technical, medical, etc. Consecutive and gradually escalating results outside of the standard range will result in warnings.

For software testing, there are functional and non-functional testing to ensure that AHUD meets industry standards of requirements. The functional testing includes unit testing, integration testing, system testing, and acceptance testing. The non-functional testing includes reliability testing, performance testing and usability testing.