

NAVSEA - Product Requirements Document

About the Team

Team Name: Sea++

Company: Navy Pt. Mugu

Team Members:

- Lyuda Panina (Lead)
- Vivian Ross (Scribe)
- Emily O'Mahony
- Kyle Kam
- Thao Phan

Edited by: Clay Greunke

Introduction

What is the problem?

Navy ships contain multiple systems, ranging from weapons to communication, that need to be maintained. These systems are up to forty years old, and they need maintenance in the form of repairs, upgrades, or retrofits. The existing model for undertaking maintenance involves sending a trained expert to a ship in order to work with the system. This is resource intensive and inefficient.

How is the problem being addressed today?

Existing AR applications exist to walk users through systems maintenance through two different methods procedural guidance (delivered compiled instructions) and ad-hoc instructions in the form of a "remote expert". For the latter, these applications work by scanning the environment to get a physical understanding of the space and create a map/representation of the room the user is in. The map created can be used by an expert guiding a novice through a procedure by placing overlays, models, and animations of what needs to be interacted with. The expert will provide verbal feedback during the process to the novice.

An example of an ad-hoc "remote expert" implementation:

- Four Eyes Lab - A research application that uses video conferencing and remote scene exploration in order to allow spatial annotations to be viewed in augmented reality

The other form of maintenance method is through procedural guidance. This application has its benefits in that it is self-contained. Authoring step by step procedures with 3D models that will be overlaid on the physical object being worked on in AR.

An example of traditional AR procedural guidance implementation:

- PTC and their Vuforia Studio line to author step-by-step AR guidance.

However, up until now, current AR implementations for procedural guidance have had no objective way to measure whether someone has actually followed the procedure. That Navy has created a process called qualitative quality assurance for mixed reality

(<https://patents.justia.com/patent/11138805>) that promises to do just that, by having an application that contains the evaluation benefit of ad-hoc instructions with the stand-alone possibility of procedural guidance (giving guidance through a model of the system). What is missing is what would a kalman filter look like that can take those independent sensor streams and come back with a confidence level (that can be presented to the user) of the task being performed correctly.

Objective, Implementation, and Assumptions

What is our objective and who will benefit from it?

Our goal is to develop an application to allow an untrained user to repair or maintain some system with the help of a CAD model of the system within augmented reality. This would save on resources by eliminating the need for an expert to be transported to the ship for repairs. This application will be an augmented reality based application that will be deployed on a HoloLens 2.

To do this, we will provide a user with instructions, then the application will take in sensor streams and evaluate through a kalman filter of how much (a confidence value) the system believes the process has been done correctly. That feedback will be presented back to the user. This will raise the end-user's confidence that the procedure is being done correctly.

How will we implement our solution?

We will create a physical proof-of-concept model, consisting of a board with switches, buttons, dials, and a lever to simulate machinery that needs to be repaired. We will model the board using Autodesk Fusion 360 in order to create a 3D file that can be imported into Unity. We will then create a Unity Prefab of the model to use in our application. We will program instructions for the user to follow, such as pressing buttons and flipping switches.

In order to verify that a user is following the instructions before programmatically proceeding to the next, we will add hand, gesture, and eye tracking. We will verify that a user has moved their hand to approximately the correct location and has made the correct gesture (such as pointing their finger to press a button). We will also use HoloLens eye tracking to verify that a user is looking at the component of the machine that they should be manipulating. With these checks and providing feedback to the user, we aim to increase confidence that a user is correctly following the instructions.

What are the necessary assumptions to be made?

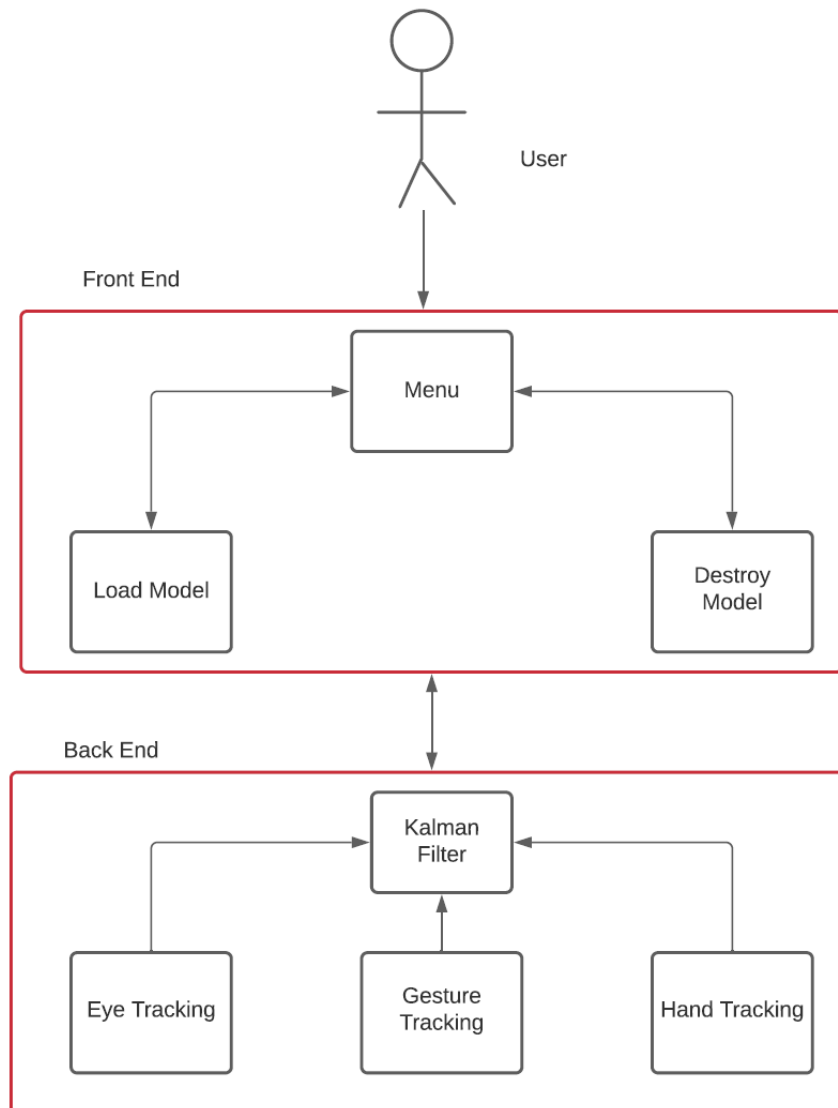
Our application will assume that the person on the ship wearing the HoloLens has a clear vision of the broken machine: adequate lighting and no obstructions in the way on the ship. In addition, we will assume that the user has a CAD model of the system available. Although this may not be true now, the hope is that in future decades, as this technology becomes more widespread, work will be done to create the necessary CAD models to make applications like this one applicable to any system. Finally, we will assume that the user has the ability to line up the CAD model precisely with the actual, physical machine.

System Architecture Overview

The diagram below shows the data flow of our interactive application. The author will be able to open a menu and open a CAD model in space and create interactive instructions for the end-user to follow. The instructions include pressing buttons and flipping switches along with expected values for touching/gestures and eye tracking expected times for the operation. That information will then be compiled and sent to an end user with a HoloLens.

The end-user will put on the HoloLens and start the application. The user will then be able to use the menu to begin the interactive instructions. The application will be tracking the user's eyes and hands and will not proceed with the next instruction until the user has performed the necessary action (decided on by a confidence level generated by a kalman filter). Upon completion of the instructions, the instructions will confirm that the repair is complete and display a confirmation button. When the user presses the button, the instructions and the CAD model will disappear, and the user will be free to begin again or exit the application.

System Architecture Model:



User Stories

As a HoloLens user, I can rotate my left palm towards my face to pull up a menu.

[GitHub issue](#)

- Scenario 1: The user turns their left palm towards their face, if they are looking at their left palm, the menu with available CAD models appears
- Scenario 2: The user's left palm is not shown or they are not looking at it so the menu is not shown

As a HoloLens user, if the menu is currently active, I can choose from a variety of available CAD models to load

- Scenario 1: The user chooses the model and it renders on the HoloLens successfully
- Scenario 2: The user chooses the model but if the file is corrupted, an error notification needs to be displayed

As a HoloLens user, I can only have one CAD model active at a time

- Scenario 1: The user renders a new CAD model when there are none currently active, the new CAD model should start rendering
- Scenario 2: The user renders a new CAD model when there is another currently active, this will destroy the current model and start rendering the new model

As a HoloLens user, if the menu is currently being displayed, I can turn my left palm away from my face to remove the menu.

[GitHub issue](#)

- Scenario 1: The user turns their left palm away and the menu disappears

As a HoloLens user, I can resize the CAD model to fit the real world object in front of me

[GitHub issue](#)

- Scenario 1: The user is able to resize the CAD model successfully and the dimensions of the CAD model is dynamically updated
- Scenario 2: The user is not able to resize the CAD model and the dimensions of the CAD model stay as they were

As a HoloLens user, I can grab the CAD model and move it in space to align it with the physical model.

[GitHub issue](#)

- Scenario 1: The user is able to move the CAD model through space to align it with the physical model
- Scenario 2: The model is not able to move through space and stays stationary.

As a HoloLens user, I can move the CAD model to align it with the physical model using a hand-held controller.

[GitHub issue](#)

- Scenario 1: The user is able to make fine adjustments to the position of the CAD model using a hand-held controller.
- Scenario 2: The model does not move and stays stationary.

As a HoloLens user, given that a CAD model is currently active, I can choose to start instructions

[GitHub issue](#)

- Scenario 1: Once the CAD model is active and fully rendered, the user is able to start the set of instructions by clicking the start button. Once that is successful, the start button will turn into a stop button
- Scenario 2: If the start button is not pressed, no instructions should be displayed

As a HoloLens user, given that I have started the instruction steps, I can see the instructions in a text box over the CAD model

[GitHub issue](#)

- Scenario 1: Once the user chooses to start the set of instructions, a text box appears over the CAD model to direct them

As a HoloLens user, I can hear the instructions corresponding to the text model read to me.

[GitHub issue](#)

- Scenario 1: The instructions are read to the user by a voice bot as they appear

As a HoloLens user, given that I have started instructions and is not yet finished, I can choose to stop instructions and close the CAD model

[GitHub issue](#)

- Scenario 1: Once the user clicks the stop button, the CAD model and the instructions will destroy themselves, leaving a blank slate for the user

As a HoloLens user, my eye movement can be tracked and translated through 3D space to determine where/what my eyes are looking at.

[GitHub issue](#)

- Scenario 1: There is a dot which will be drawn to indicate where we think the user is looking
- Scenario 2: If there is an error in coordinate translation, the dot should disappear

As a HoloLens user, my eye can be tracked to verify directional steps

[GitHub issue](#)

- Scenario 1: The user completes the previous steps while looking at the correct part, if all conditions are met, they can move on to the next step
- Scenario 2: The user completes the previous steps but is not looking at the correct part, thus they are not verified and cannot continue to the next step

As a HoloLens user, my hand position can be tracked to verify I am completing tasks in the correct position on the machine.

[GitHub issue](#)

- Scenario 1: The user moves their hand to the correct position on the machine, which allows them to move on to the next step.
- Scenario 2: The user fails to move their hand to the correct position, which results in the instructions halting and waiting for completion.

As a HoloLens user, my hand gestures can be tracked to verify I am performing the correct task on the machine.

[GitHub issue](#)

- Scenario 1: The user manipulates their hand in the correct way in order to perform a task (ex. Curling their fingers but extending the index finger to press a button), which allows them to proceed to the next step

- Scenario 2: The user fails to manipulate their hand in the correct way in order to perform a task, which results in the instructions halting and waiting for completion

As a HoloLens user, given that I have completed an instructional step and it is successfully verified, I can choose to move on to the next step

[GitHub issue](#)

- Scenario 1: If the user finishes a task successfully, they can click a button to move onto the next step
- Scenario 2: The user does not click the button to move onto the next step, the next step will be loaded automatically after a timeout

As a HoloLens user, I can navigate to previous steps by clicking a button

[GitHub issue](#)

- Scenario 1: The user can choose to go backwards to an earlier step by clicking a button
- Scenario 2: The steps should only move backwards if the user clicks the button

As a HoloLens user, my progress through the procedural instructions can be tracked in order to verify that I have completed the instructions.

[Github Issue](#)

- Scenario 1: The user is notified that they have completed the task upon completing all of the instructions.
- Scenario 2: If the user has not completed the task, they are not told that they have completed all of the instructions.

As a HoloLens user, I can click a confirmation button in order to acknowledge that I have completed the repair or maintenance task.

[Github Issue](#)

- Scenario 1: The user sees a button appear underneath the text box indicating that they are finished, they can then click the confirmation button to end the task
- Scenario 2: The user should only see this button once every instruction has been completed and successfully verified

As a HoloLens user, once everything is finished and successfully verified, I can confirm to make it go away or it will automatically go away by itself after some time out.

[Github Issue](#)

- Scenario 1: The user clicks the confirmation button and the CAD model will destroy itself, leaving a blank slate for the user
- Scenario 2: The user does not click the confirmation button, the CAD model will close itself automatically after a timeout interval

The instructions will highlight the part that the step is focused on with an animation or an arrow.

[Github Issue](#)

- Scenario 1: The area of interest will be animated to show the user what action to perform.

- Scenario 2: If the area is not animated, an arrow will be used to highlight the area of interest.

Technologies Employed

- Unity (C#)
- Mixed Reality Toolkit
- Autodesk Fusion 360 (for demo CAD models)
- Microsoft HoloLens Companion App

Sequential Model:

