



Sprint Planning Document (Sprint 3)

Sprint Goal Backlog (Sprint 3)

April 1 – April 22, 2025

Trang Do, Donovan Kohler, Samuel Kwon, Raudel Armenta, Anthony Rutherford

High-level Project Overview

Project Mission:

- The focus of our project is to explore what is possible with the cutting-edge Apple Vision Pro. We aim to build an app that collects data from the Vision Pro and utilizes Hugging Face models for inference to provide a kind of distributed intelligence. This project will also set up initial infrastructure for future development with the Vision Pro and cloud computing.

Problems We Are Solving:

- People don't know exactly what the Vision Pro can be used for
- Developing on the Vision Pro is relatively new and a sandbox environment to play around with the headset would make further development easier

Project Overview (High-Level Features):

- **VisionOS app - Discover:**
 - **Photo Library:** Users can select an image from the Photos app on the Apple Vision Pro and display the image.
 - **Main Camera Access:** Users can capture an image using the main camera from the Apple Vision Pro and display the captured image.
 - **Connect to WebSocket:** Users can send the selected image to AWS Lambda function via WebSocket. Then, the app can receive the response from the function and display the result to users.
 - **Speech Recognition:** Users can talk to the headset and apply speech recognition to send a request to OpenAI. Then, the app can receive the response from OpenAI and display the result to users.
- **Cloud Computing**
 - **Cloud Service:** AWS, using EC2 instance
 - **Protocol:** WebSockets Secure, as required by project sponsor
 - **AI/ML:** Hugging Face models
 - **Flow:**
 - Client establishes WebSocket connection with AWS
 - Client sends message over connection
 - AWS receives message and uses Hugging Face models to infer and provide result
 - Result sent back to user over connection

Sprint 3 Planning

Sprint 3 Goals:

1. Add voice recognition and the ability to send requests to ChatGPT API
2. Make security updates based on recommendations from the cybersecurity team
3. Create a website for documentation regarding “How to install software”, “How to use each feature”, “How to modify/extend software”, and “FAQs” section
4. Write unit tests for the visionOS app and backend

Sprint 3 Deliverables:

- **Allow users to use voice recognition and send a request to OpenAI**
 - **Assigned:** Raudel Armenta
 - Finished OpenAI request for speech to text
 - Attempted to create view to display results from OpenAI to user
- **Display the Voice Recognition View based on Navigation Stack and Refactor the Code for Readability**
 - **Assigned:** Trang Do
 - Worked on the Voice Recognition View to display the result from OpenAI to user
 - Use Navigation Stack to stack the Voice Recognition View on top of another view
 - Refactored the code for readability
- **Set up the documentation website from Read the Docs Tutorial**
 - **Assigned:** Trang Do
 - Followed the Read the Docs tutorial to set up our documentation website
- **Work on website section for how to install software (visionOS)**
 - **Assigned:** Trang Do
 - Documented the steps to install Xcode and the visionOS app
 - Also listed the dependencies and how to install them
- **Work on website section for how to set up AWS**
 - **Assigned:** Sam Kwon
 - Wrote section of docs to explain to users how to set up the AWS EC2 instance and get the server up and running
 - Instructed user on how to use test file
- **Work on website section for how to use each feature**
 - **Assigned:** Donovan Kohler

- Listed all product features with step-by-step instructions on how to use each feature through the visionOS app
- **Work on website section for modification/extension**
 - **Assigned:** Raudel Armenta
 - Documented steps on possible extensions for front end
 - Documented possible modifications for front and back end
 - Talked about dependencies, backlog, style, and tests
- **Work on website section for FAQs**
 - **Assigned:** Anthony Rutherford
 - Filled in questions and answers for questions often asked by sponsor and other application testers
 - Tested multiple formats to try and make page easily readable
 - Gave detailed answers that would allow someone unfamiliar with project to follow along
- **Do research on unit testing in Xcode and write unit tests for the visionOS app**
 - **Assigned:** Trang Do
 - Used XCTest framework and wrote the unit testing for OpenAI and WebSocket Connection
- **Add screen recording for software installation and features to documentation website**
 - **Assigned:** Trang Do
 - Did many screen recordings for how to install the visionOS app, to start up EC2 instance, to start the server, to use all the features on the app, and to run test cases.
 - Uploaded those screen recordings to the website documentation and the project website
- **Make security updates based on recommendations from the cybersecurity team**
 - **Assigned:** All
 - Added various verification methods including type and size checking
 - Server is only running when app is in use
 - Only devices using UA IP can connect to EC2 instance
 - Began to implement WSS but cyber-team posed that it may be overkill and only lead to making the process too difficult to use
 - Made sure there were no important keys or information in code that could lead to security risk
- **Write unit tests for the backend**
 - **Assigned:** Donovan Kohler, Sam Kwon, Anthony Rutherford
 - Wrote unit tests for AWS backend server
 - Validate invalid and valid requests as well as WebSocket connection

- Added error messages for different failure scenarios