Test team ID3 goals

Feature	Complete
Multi Level Logging	No
Find Dre/ Jest Issues	No
Add to test list	Yes
Add to test matrix	Yes
Gherkin for Data Display	Yes
Gherkin for UI Interaction	Yes
Gherkin for PCA backend	Yes
Gherkin for data exclusion	N/A
Improved test plan	Yes
Review gherkins created	Yes
Create Plan for smoke test	Yes
New feature test after code freeze	Yes

Test team ID4 current planned features

Finish fix for Drei + Jest issues
Add Gherkins for features implemented
Complete MLL
Create Smoke test framework
Implement first set of smoke tests using Tesseract for image - text recognition
Start to look at different ways we can stress test the application

Test team learning / knowledge sharing

Information	Length	Attendance
How to write a gherkin	1 hour	Mitchell/ Quinn on discord
How to write a gherkin	1 hour	Mitchell/ Joe on discord

Smoke Test

During this ID, we reached out to 2 communities. First playwright, and second threeJS. By reaching out I was trying to determine if there were any premade libraries or frameworks to run the automated testing on WebXR

1. Playwright responded by saying there was a prior commit where it looks like someone shut off the support for WebXR. So currently it would not work to any scale but if we filed a ticked with them and they found that there was more drive for it, they would start to create some amount of support for it. – This would take much too long for our purposes
2. threeJS did not respond, but their community had someone respond to me saying that they don't think there is anything out there that would support WebXR automated testing out of the box.

New Approach

The new approach is a hybrid approach to smoke testing. Instead of someone running the playwright / Jest (undecided which is better needs to be tested) one of the testers will manually walk through the app recording pictures of different scenarios. From these pictures we will then pull different data from them, allowing us to find words, colours, and other things in an ideal world. We can then run the tests to verify different results.

This will come in different versions, and we should have some number of tests working over the next ID. To start we will verify that in a picture it has the colours of the Axis, which should verify all individual accesses are on the page, and when we label an Axis, we can then use tesseract to pull strings from the image and then verify that the required text is on the display.