



SPOOF DETECTOR

TEAM MEMBERS

- Leah Casey
- Samuel Gross
- Logan Parker
- Ashley Vanaman
- Michaela Walker

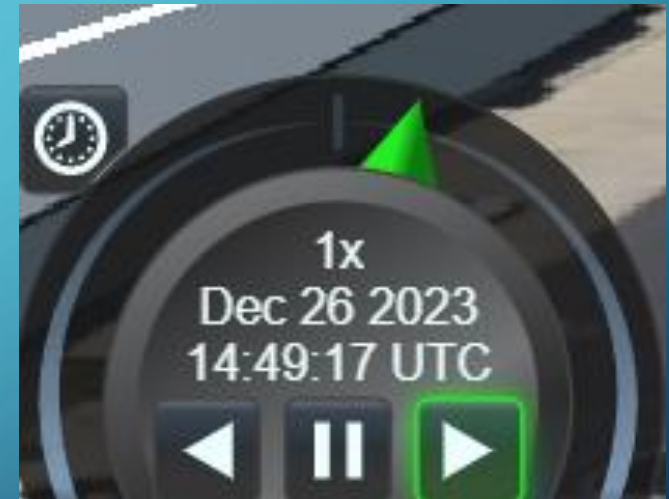
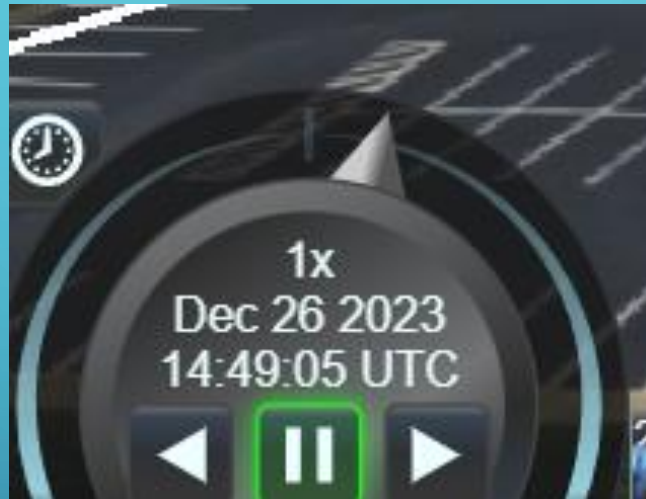
CLIENT INFO

- Pierce Aerospace
 - Aaron
 - Gary
 - Chris
 - Michael

The background is a deep blue gradient with intricate, lighter blue circuit-like patterns. These patterns include various line styles, circles, and dots, resembling a printed circuit board or a digital network. The word "FEATURES" is centered in a clean, white, sans-serif font.

FEATURES

FEATURE 1: TIME IMPLEMENTATION




FEATURE 2: DRONE MODEL



FEATURE 3: CONNECT TO CLIENT DATABASE TEST SERVER


✓ UTILITIES-MASTER

 A50Flights_11-23to12-23.csv

 CSVToFlights.py

 DroneFlightSim.py

 GetServerData.py

 PostData.py

 SquidSerialRead.py

FEATURE 4: REFRESH BUTTON

Clean

Refresh

HOW DOES EACH FEATURE CONNECT TO BUSINESS REQUIREMENTS?

- Feature 1 (Time implementation) connects to Business Requirement 2 by showing visual and text information base on drone flight time duration and by showing the specific locations of the drone(s).
- Feature 2 (Drone model) connects to Business Requirement 2 by showing visual information to end users by showing the flight path of the specific drone.
- Feature 3 (Connecting to test server) connects to Business Requirement 1 by developing a data curation framework for forensic users that can be used to identify potential spoofing attacks within cyber-physical systems.
- Feature 4 (Refresh button) connects to Business Requirement 1 and 2 by providing an updated drone flight path for a drone as well as providing an updated data from the test server framework.

MENTOR FEEDBACK

MID-ITERATION

- In a good spot for the project.
- Likes the goals and features set for this iteration.
- Liked the idea of us beginning meetings by relaying progress and issues in our tasks.

END OF ITERATION

- Glad we got the time feature implemented for a drone.
- Only complaint was that the drone flight path instantly started rather than starting on a pause.
- Thinks the implementation of the drone model is neat.
- Believes we made good progress this iteration.

CLIENT FEEDBACK

MID-ITERATION

- Relayed to us that the test server has been completed and showed us how to use it.
- Stated that they would like to see operator information in text data.
- Would like more drone information
- Would like to see a drone model rather than multiple dots
- Would like to see time implemented in drone flight.

END OF ITERATION

- Was excited to see that we were able to implement the test server as well as show a refresh of data.
- Was excited to see that the language change didn't set us far back.
- Loved the moving drone model through the flight path
- Excited that the time play-back, play-forward, and pause were able to be implemented
- Would like to see the height, longitude, latitude, and rating implemented back into cesium.

CLIENT USING SOFTWARE

- Used to software fairly easily while we walked him through the process of what needed to be running before getting to Cesium.
- Features that were seen included the drone model, new features that were tested included the time functions at the bottom of cesium as well as the refresh button.
- Messed with our software for 7 minutes
- Client was thrilled when seeing the new features added to Cesium and that interaction on Cesium has increased.
- Noticed that the height, longitude, latitude, and rating has disappeared from cesium when interacting with the drone

4TH ITERATION FEATURES

- Display more drone information such as operator location
- Allow three dimensionality to view potential collisions between drones
- Create a user sign on system
- Package a releasable to end-users: Unreal Engine product

WHAT DID EACH TEAM MEMBER DO AND WHAT THEY WORKED ON

- Ashley: Worked on creating Javascript tests for the script.js file in Java. After language switch, worked on converting and redoing the java tests into python. Assisted in helping trying to solve model drone appearance issue.
- Leah: Worked on refactoring of DataProcessor, updated tests (Java). After language switch, added time implementation feature that allows the user to view the drone's actual flight path (sped up, but can be paused, slowed down, played backwards, etc.). Worked with Michaela on getting a model drone to appear on the flight path instead of a dot.
- Logan: Worked on modularizing webpage JavaScript file. After language switch, finished the python class that periodically requests drone data from a test server and then calls the main DataProcessor method, cleans up the data, and saves it to a JSON file used by the web page. I also added the refresh button, which updates the data structure in the Javascript code that holds the drone data by once again referencing the saved json file. Refactored tests to pass and added more assertions.
- Michaela: Before java to python switch, worked on refactoring for readability of the script.js, worked on updating drone and data-processor test classes to test the time function. After the language switch, task included the implementation of the drone model (with troubleshooting issues from Leah and Ashley).
- Sam: Worked on implementing time conversion and date into the java code from the data sets. Then worked on implementing the language switch from Java to Python for the project.

INTERESTING SLIDE



Buddy — Leah's doggo.
He has his own GPT. This
is him as origami.



Vega — Logan's cat. Mean,
loves food.



Jedi- Michaela's dog
in his favorite spot
outside.