

Testing Plan

SDDec23-04

Unit Testing

- Use heavily for backend
 - Needs high reliability
 - Easy to design tests for
- Add test for basic frontend and camera ML functions

Interface Testing

- Primarily needed for the API
 - Ensure data provided to API is interpreted correctly by server
 - Ensure data provided by API is interpreted correctly by user software
- Also needed for camera and machine learning interface
 - Ensure that the machine learning algorithm correctly interprets format of images

Integration Testing

- Three primary critical integration paths exist in our design
 - Integrating YOLOv5 output data from LabStats
 - Test by mixing data from the two sources and ensuring they are compatible
 - Integrating data into database and ensuring data combines correctly
 - Test using unit tests with different sets of camera info and lab data
 - Integrating website into Canvas
 - Ensure we properly interact with Canvas API

System Testing

- Will largely be done by providing verifying real data, and ensuring system interprets and processes inputs correctly
 - Check real lab capacity, ensure website is displaying correct data, and updates quickly when data changes
 - Check lab schedules on classes.iastate.edu and ensure lab schedules are displayed correctly
- This testing will mainly be done by human verification. Team members will look at data and ensure correctness.

Regression Testing

- Part of our automated pipeline
 - Run all unit tests
 - Run any other tests that do not need human interfacing
- Perform relevant manual tests after a major change

Acceptance Testing

- Because our requirements align closely with our system testing goals, acceptance testing looks similar to system testing
 - Collect data on how often lab occupancy refreshes
 - Ensure real lab counts align with what is displayed on website
- In order to involve users in our testing process, we could administer a survey on user experience with CLARE

Security Testing

- Main security concern is output from cameras
 - We will perform penetration testing to ensure it is not easy to access
 - We will utilize static analysis tools to ensure code is up to security standards