Testing Lightning Talk

## Small Equipment Checkout Locker Software

Ainara Machargo del Rio, Camille Cramer, Laura Mejia, Jon Gonzalez and Ben Johnson

sddec23-03

Unit Testing

For our project we will be testing individual components of our website. We will be using Python libraries such as unittest as a tool to help us test our code.

The team will be testing if data is being recorded and collected from our database, to ensure the data in the MySQL database is correct. We will test the backend connection to the solenoid, checking whether or not it is opening and closing as it should, and test the GUI for the website as well. The team will test individual components of the GUI, such as buttons, labels, and text fields.
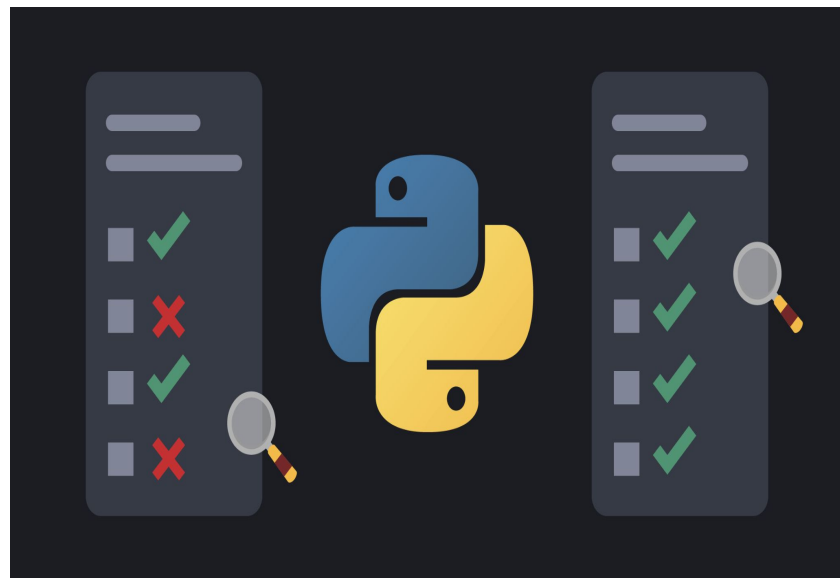
Interface Testing

For our project we will be developing a GUI for the website when browsed on a computer, and a separate GUI for the touch screen in front of the locker. These interfaces will be tested using the library PyAutoGUI, which will help us simulate user inputs. This will help us test components such as buttons and text on the GUI.

Integration Testing

- Server integration with hardware
  - Need to physically test the hardware
  - Must be done in person
  - Requires documentation for ETG employees to follow
- Front end integration with API
  - Postman for API testing
  - Ensures the connection between front end elements and the server are working as intended
- API integration with database
  - Verify the proper data is being recorded
  - PyTest suite

System Testing

- Our test plan is comprised of 3 main sections
- Each section must be passed before the application can be deployed
- The tests will cover all basic interactions with the application
- Unit Testing
  - Tests using mock data to ensure the application is working as expected
- Interface Testing
  - This is a focus of the test plan
  - User interactions are a priority
- Integration Testing
  - Connecting each component properly is critical to the project

Regression Testing

- We will add tests for each new component as they are created
- This will allow us to run tests on all project components at once when a new component is added
  - This will help us find issues with new components quickly
- There are several critical components that cannot break:
  - Lockers ability to open and close on demand
  - Users can use the reservation website
  - Users can use the touch screen
  - Users can use the admin website

Acceptance Testing

- Due to our projects timeline, we anticipate having a working prototype by the end of the semester
- After demonstrating the capabilities to our client, we will hand it over to them for testing
- It will be deployed in Coover Hall over the summer to gather feedback
- This feedback will be used to refine the design in the second semester

Security Testing

- We will conduct some basic security testing before deploying our project.
- This will help us look out for critical vulnerabilities.
- Static code analysis via GitLab.

Results

- Errors in the code will be caught by unit testing or by functionality based on the interface and integration.
- A successful test suite will ensure that the code is performing as expected and that the system is functioning as expected.

QUESTIONS?