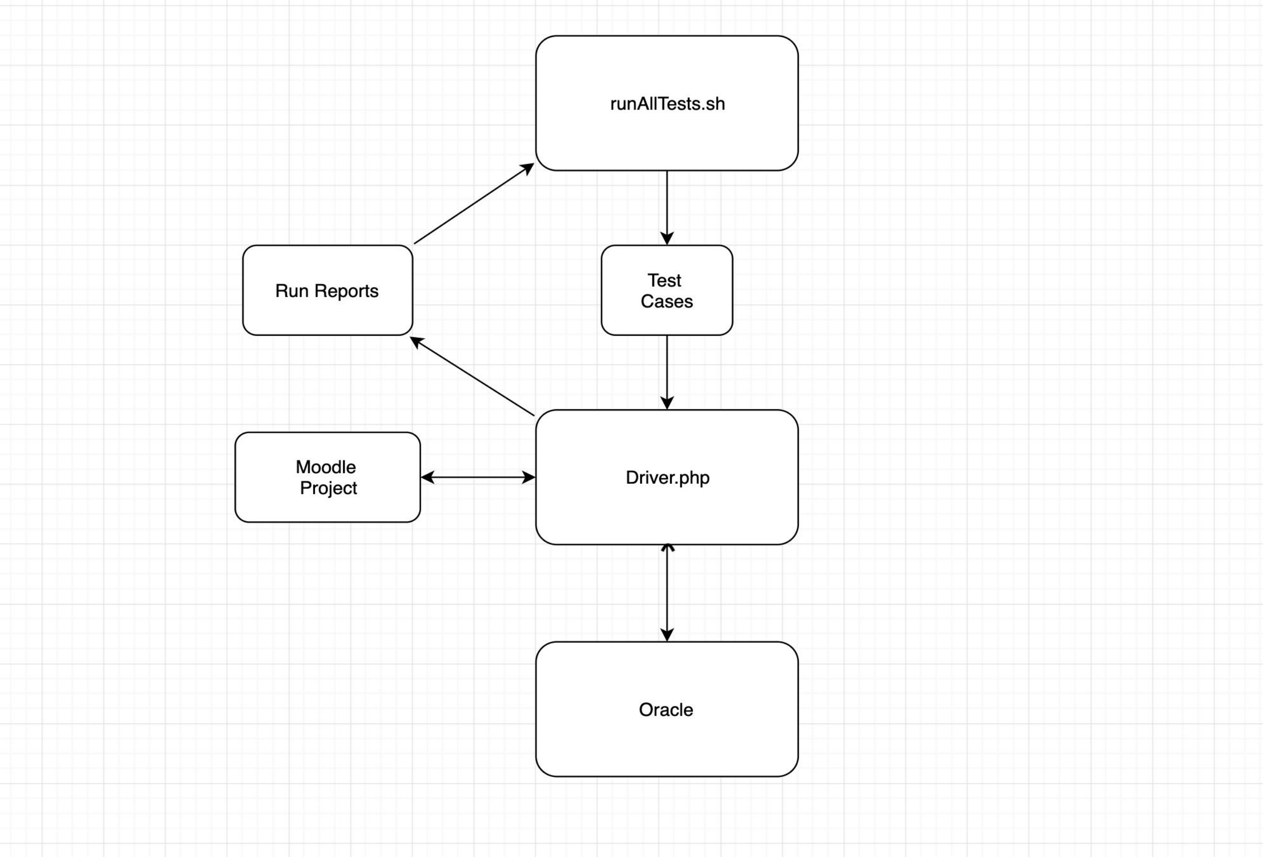
Automated Testing of OSS Moodle

Chandler Long, Alex Laughlin, Andrew Nesbett

Project Description

For our project we will be testing various methods from the open source software Moodle. We will be testing five methods with five different test inputs. These results will be recorded and output to an HTML file. Once the original methods are tested, we will inject faults into the moodle code to see if our test cases are written well enough to detect the faulty code. These output of these tests will be written to an HTML file to view the results.

Project Architecture

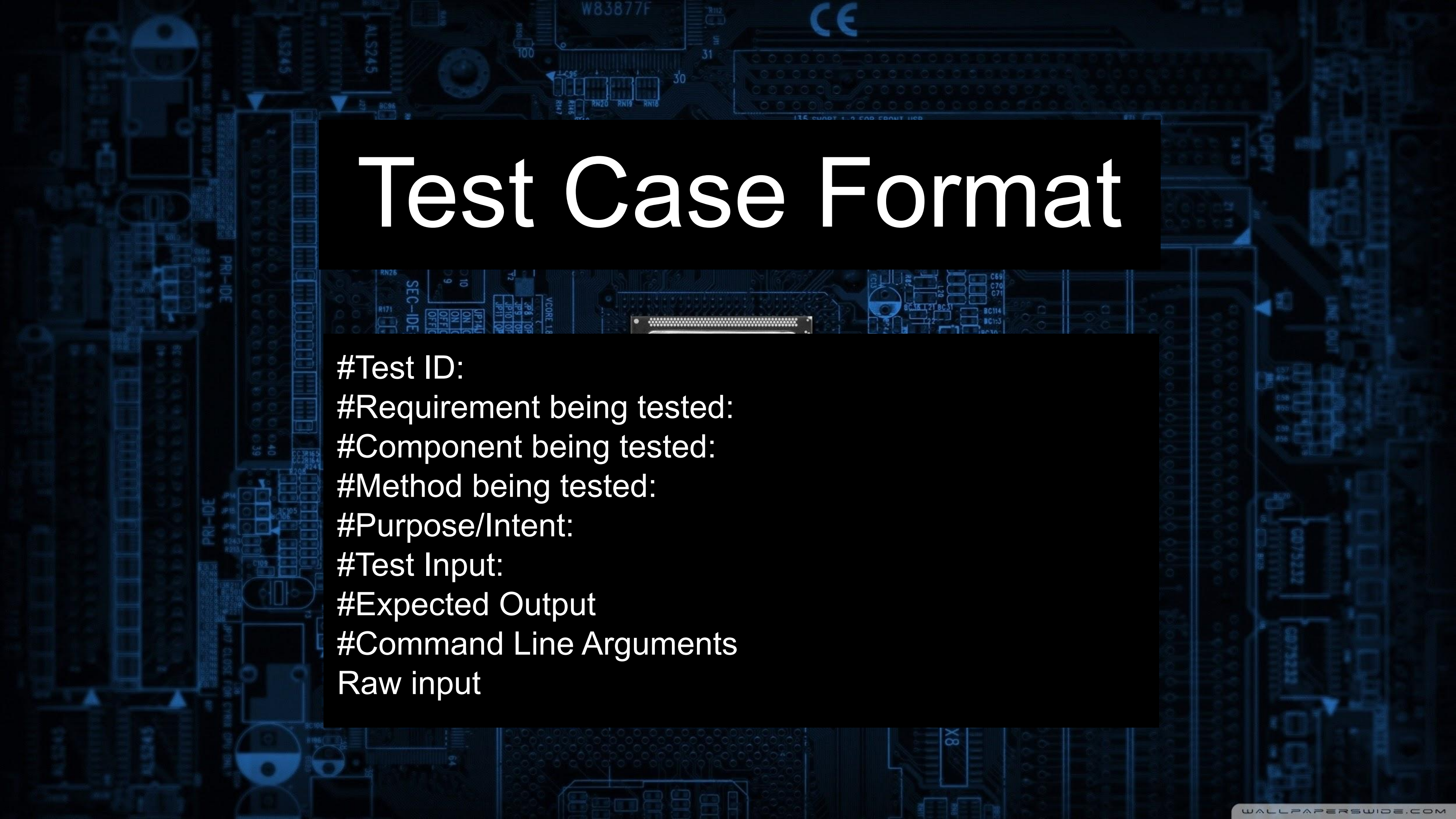


Methodology

- Call a bash that loops through all of our test cases files
- The necessary data is aggregated and assigned to variables
- The expected output is then written to the oracle
- The bash script navigates to the directory in moodle to run the method
- The method being tested is run with the test cases input
- The output of the method is written to a text file
- The expected output, actual output, and the result of comparing the two are written to and saved in an HTML file

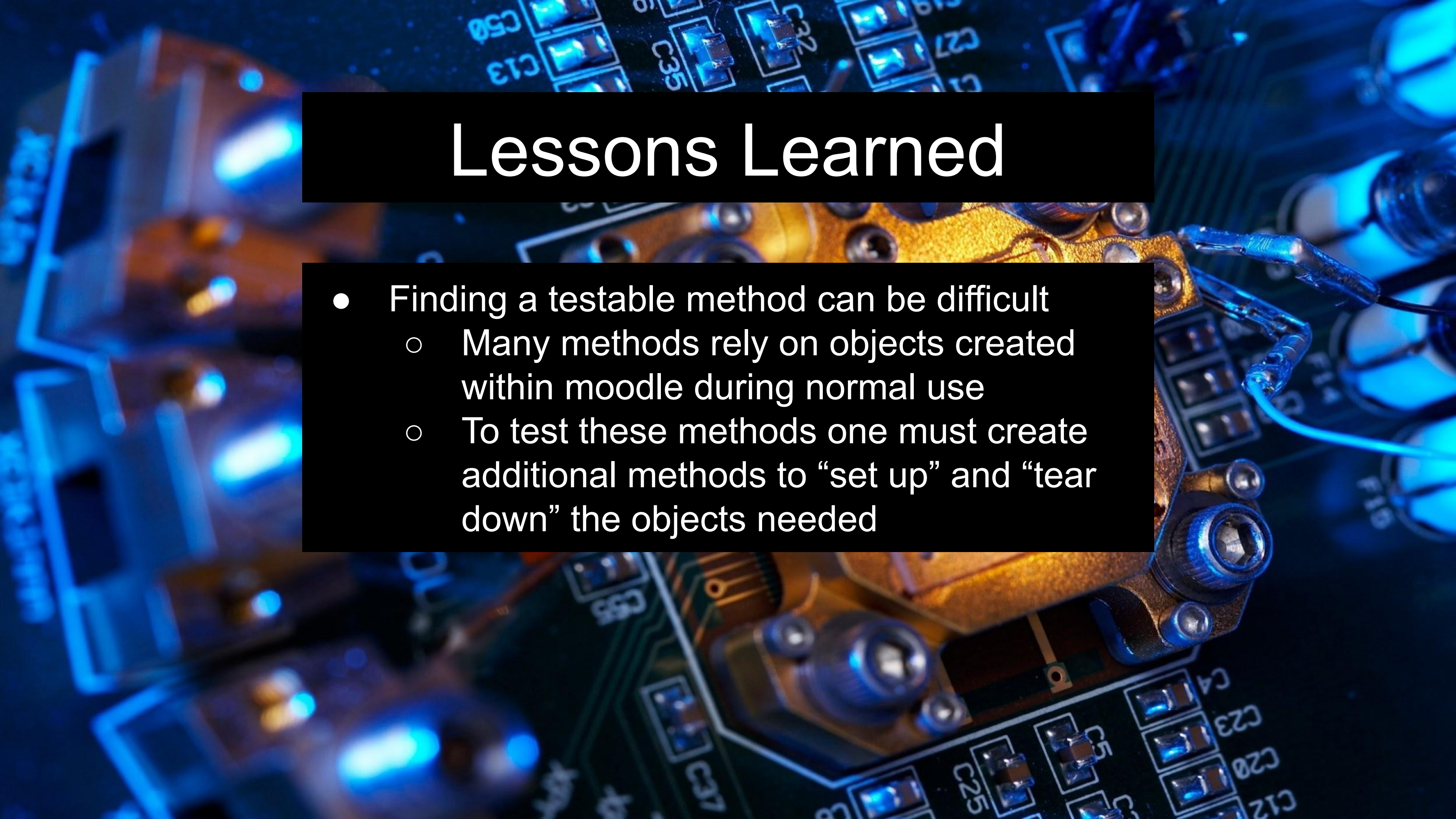
Test Case Format

#Test ID:
#Requirement being tested:
#Component being tested:
#Method being tested:
#Purpose/Intent:
#Test Input:
#Expected Output
#Command Line Arguments
Raw input



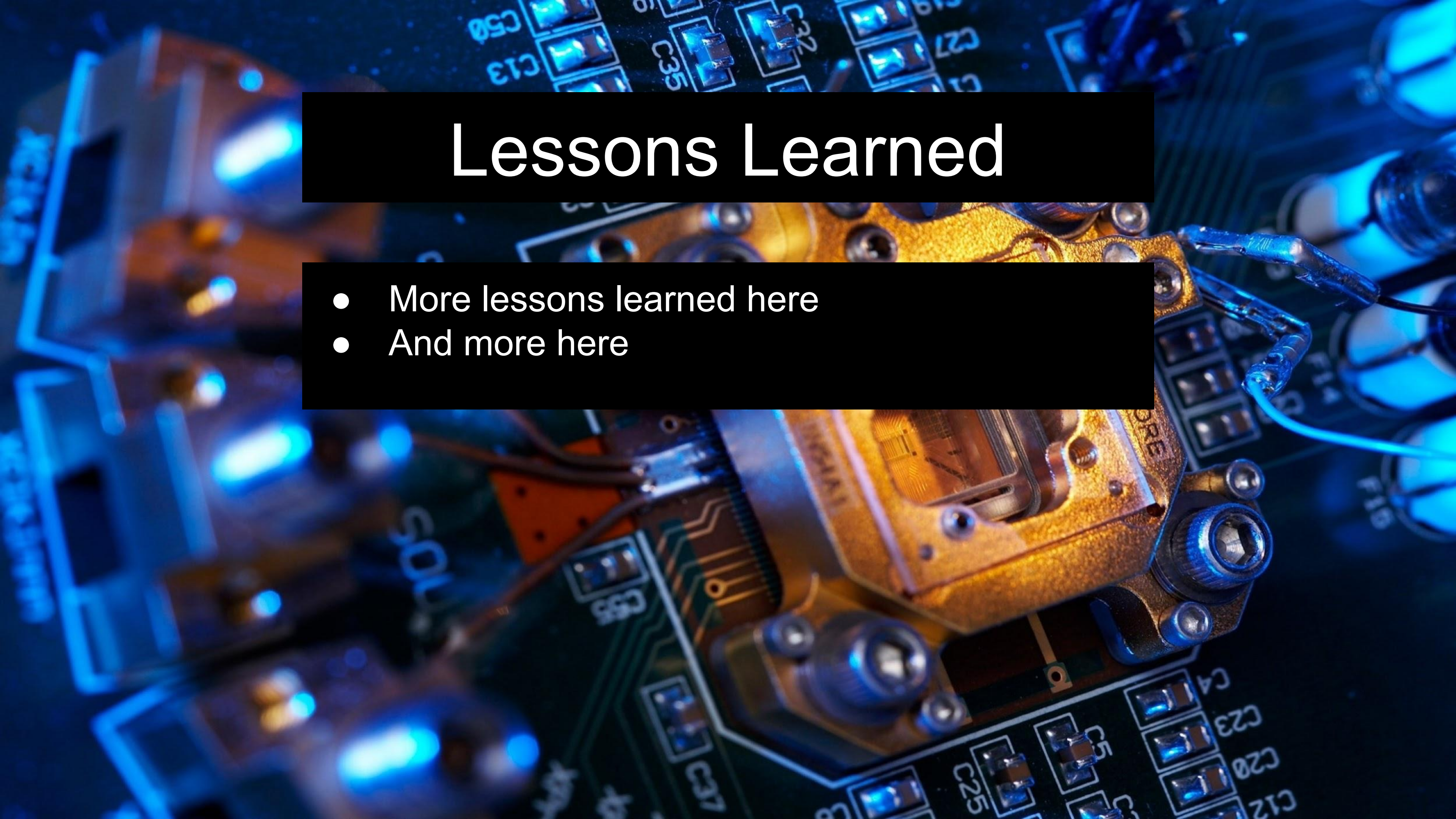
Test Case Example

#Test ID: 001
#Requirement being tested: Tokenize accepts a string as a parameter. The method returns an array of all the words passed to it as individual elements excluding one letter words and repeated words
#Component being tested: moodle/grade/grading/lib/tokenize
#Method being tested: tokenize()
#To verify that the tokenize method works properly
#Test: "You are fish"
#Oracle: ["You”, “are”, “fish"]
#Command line arguments: php phpdriver
"You are fish"



Lessons Learned

- Finding a testable method can be difficult
  - Many methods rely on objects created within moodle during normal use
  - To test these methods one must create additional methods to “set up” and “tear down” the objects needed



Lessons Learned

- More lessons learned here
- And more here

Fault Injection Results

- Fault injection results go here
- And here
- More here
- Even more still
- Also here
- Maybe even here