Chapter 3

Task: Design and build an automated testing framework that you will use to implement your test plan per the **Project Specifications** document.

Experience

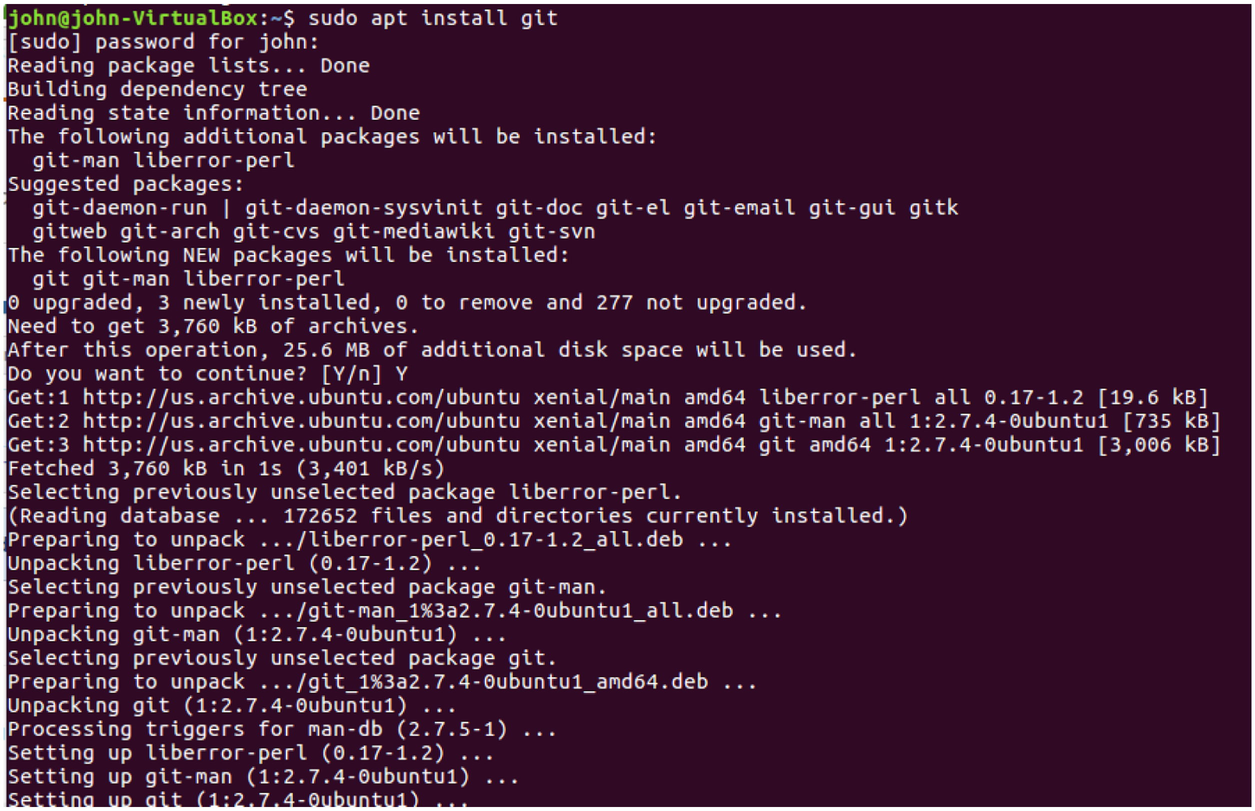
Our experience thus far has been a rough one. Our third teammate dropped the course leaving us short on manpower. We wrote our automated testing framework in Python. The goal is to make the framework know as little as possible and keep every piece of information localized to each individual test case. When we originally demonstrated our framework this was not the case. We Could not get our individual test cases to run from the commandline so we developed a driver after the fact that allow us to run individual methods from the command line. This is where we ran into many problems. An inability to easily and consistently import modules from within Amara's codebase was our primary cause of frustration because we could not access many of the classes and functions that it used. Because of this we were not able to run our original tests from our test plan because we were unable to create the necessary objects or call the appropriate functions. We were able to import from the deploy file of Amara, which contained a handful of classes and methods that we could test, but were still very limited by our inability to import modules from outside that file .

Architectural Framework

The driver of the program, *runAllTests.py*, works by fetching a list of classes located in a one of the text files in the *testCases* folder, initializing an instance of each one using Python's *eval*. Each object represents a class defined in the *scripts* folder that inherits from the abstract class *Test*. All *Test* objects define a test name for themselves that is used to look in the *testCases* folder and return all text files with filenames that include said test name. Each of these text files describes the requirement the method must meet, the component the function is located in, and the name of the function being tested. Additionally, the text files give a set of arguments and an expected result from the function's execution that the actual result is compared against. The *Test* class provides methods for retrieving that information, and automatically calls them upon initialization. For each test case file, a different entry in the *Test* object's *paramList*, *expectedList*, and *requirements* is created. The testing script then runs the function being tested with each of the parameters given in *paramList*, and compares the outcome to the corresponding entry in *expectedList*. The results are passed to an HTML formatted string that's thrown back to *runAllTests.py*, which then adds the formatted string to a larger collection string for each test. This HTML string is then written to *testOutput.html*, which is then opened at the end of the *runAllTests.py* file.

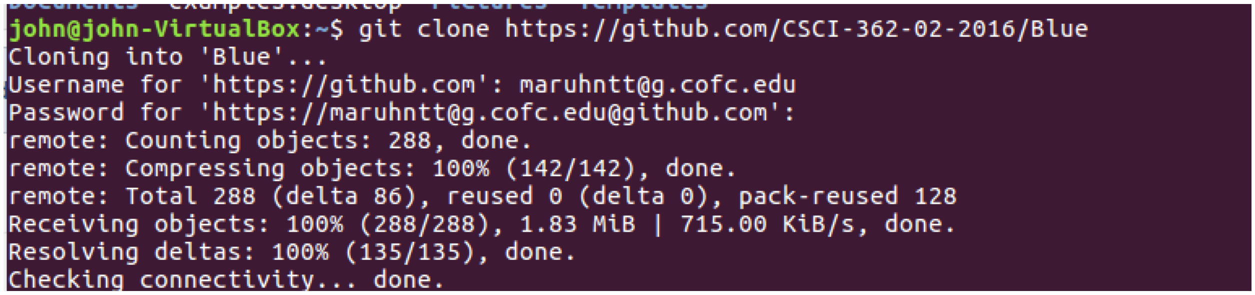
How-To:

Step 1: Install Git to your machine:



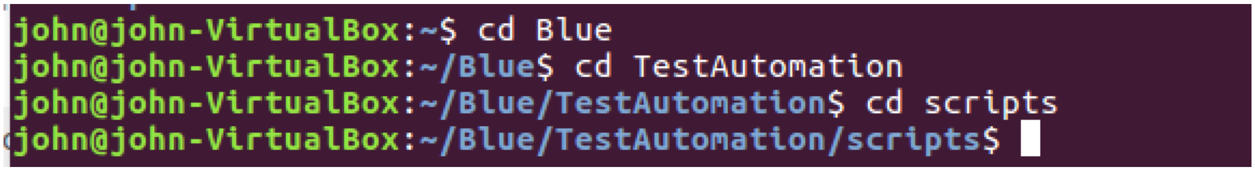
```
john@john-VirtualBox:~$ sudo apt install git
[sudo] password for john:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  git-man liberror-perl
Suggested packages:
  git-daemon-run | git-daemon-sysvinit git-doc git-el git-email git-gui gitk
  gitweb git-arch git-cvs git-mediawiki git-svn
The following NEW packages will be installed:
  git git-man liberror-perl
0 upgraded, 3 newly installed, 0 to remove and 277 not upgraded.
Need to get 3,760 kB of archives.
After this operation, 25.6 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://us.archive.ubuntu.com/ubuntu xenial/main amd64 liberror-perl all 0.17-1.2 [19.6 kB]
Get:2 http://us.archive.ubuntu.com/ubuntu xenial/main amd64 git-man all 1:2.7.4-0ubuntu1 [735 kB]
Get:3 http://us.archive.ubuntu.com/ubuntu xenial/main amd64 git amd64 1:2.7.4-0ubuntu1 [3,006 kB]
Fetched 3,760 kB in 1s (3,401 kB/s)
Selecting previously unselected package liberror-perl.
(Reading database ... 172652 files and directories currently installed.)
Preparing to unpack .../liberror-perl_0.17-1.2_all.deb ...
Unpacking liberror-perl (0.17-1.2) ...
Selecting previously unselected package git-man.
Preparing to unpack .../git-man_1%3a2.7.4-0ubuntu1_all.deb ...
Unpacking git-man (1:2.7.4-0ubuntu1) ...
Selecting previously unselected package git.
Preparing to unpack .../git_1%3a2.7.4-0ubuntu1_amd64.deb ...
Unpacking git (1:2.7.4-0ubuntu1) ...
Processing triggers for man-db (2.7.5-1) ...
Setting up liberror-perl (0.17-1.2) ...
Setting up git-man (1:2.7.4-0ubuntu1) ...
Setting up git (1:2.7.4-0ubuntu1) ...
```

Step 2: Clone our repo using the url https://github.com/CSCI-362-02-2016/Blue



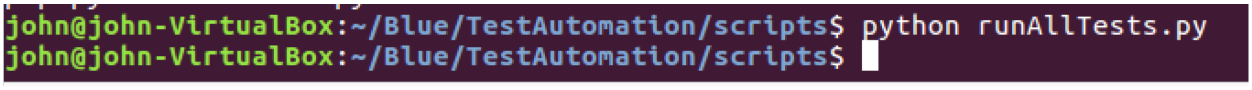
```
john@john-VirtualBox:~$ git clone https://github.com/CSCI-362-02-2016/Blue
Cloning into 'Blue'...
Username for 'https://github.com': maruhntt@g.cofc.edu
Password for 'https://maruhntt@g.cofc.edu@github.com':
remote: Counting objects: 288, done.
remote: Compressing objects: 100% (142/142), done.
remote: Total 288 (delta 86), reused 0 (delta 0), pack-reused 128
Receiving objects: 100% (288/288), 1.83 MiB | 715.00 KiB/s, done.
Resolving deltas: 100% (135/135), done.
Checking connectivity... done.
```

Step 3: Navigate to the scripts directory



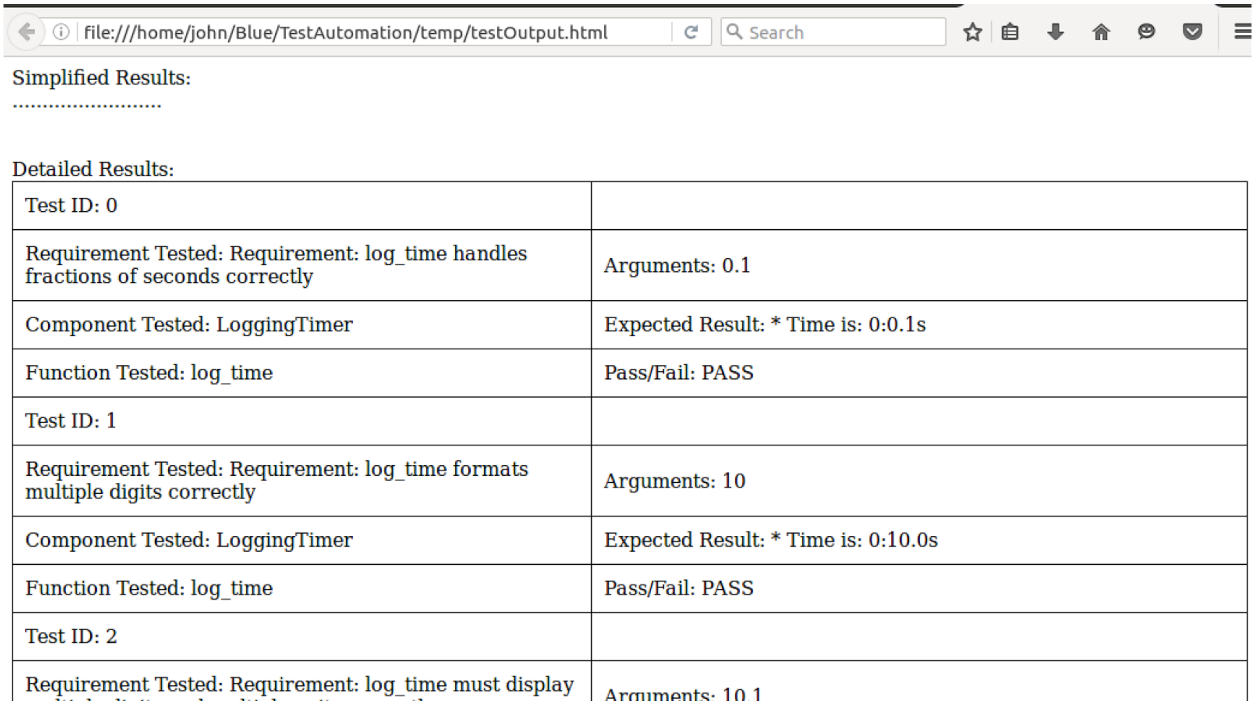
```
john@john-VirtualBox:~$ cd Blue
john@john-VirtualBox:~/Blue$ cd TestAutomation
john@john-VirtualBox:~/Blue/TestAutomation$ cd scripts
john@john-VirtualBox:~/Blue/TestAutomation/scripts$ 
```

Step 4: Run the file runAllTests.py with Python (will take a bit due to sleep functions, >25 sec)



```
john@john-VirtualBox:~/Blue/TestAutomation/scripts$ python runAllTests.py
john@john-VirtualBox:~/Blue/TestAutomation/scripts$ 
```

Step 5: An HTML page of the test results should automatically open at the end



Simplified Results:
....

Detailed Results:

Test ID: 0	
Requirement Tested: Requirement: log_time handles fractions of seconds correctly	Arguments: 0.1
Component Tested: LoggingTimer	Expected Result: * Time is: 0:0.1s
Function Tested: log_time	Pass/Fail: PASS
Test ID: 1	
Requirement Tested: Requirement: log_time formats multiple digits correctly	Arguments: 10
Component Tested: LoggingTimer	Expected Result: * Time is: 0:10.0s
Function Tested: log_time	Pass/Fail: PASS
Test ID: 2	
Requirement Tested: Requirement: log_time must display	Arguments: 10.1