

INDEPENDENT PROJECT TITLE: IMPROVING FRONT-END AUTOMATION TESTING

AIMS & OBJECTIVES

1. Turn all manual testing cases and creating them into automatic tests.
2. Implement dashboards for all live feedbacks such as failed/successful after it has been run on CATS.

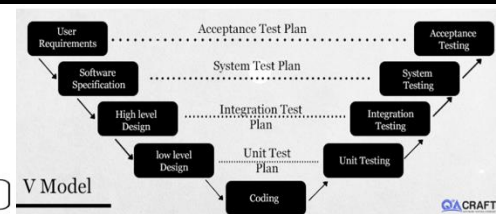
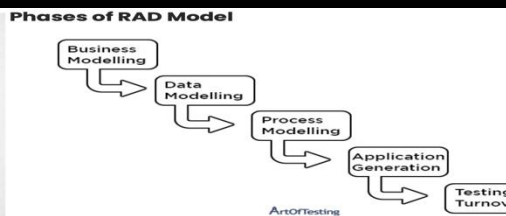
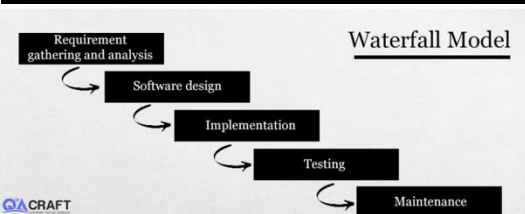
SOFTWARE TESTING

A process whereby validation and verification of a software product or an application is tested to ensure that it is doing what it is supposed to do.

MANUAL TESTING VS AUTOMATED TESTING

The process whereby test cases are executed manually without the aid of any automated tools	The process of using software tools and scripts to perform a diversified list of testing activities automatically without any human interaction
Time-consuming due to planning and execution of test cases for all core features	Not time-consuming to complete a feedback cycle, hence increases reliability
Still remains a fundamental part of testing – cannot be replaced entirely yet	High degree of repetition leading to more positive outcomes
Expensive when repeats are being done	Reduces the cost drastically when it comes to repetition.

SOFTWARE TESTING METHODOLOGIES



Github Jira Ide Technology Stack (Python & Selenium) Git Bash CATS

RESULTS & DISCUSSION

CODE FOR STANDALONE URL SCRIPT

```
13 class StandaloneUrlpage(object):
14     def __init__(self, driver):
15         self.driver = driver
16         self.url = "http://localhost:8080/"
17         self.confirm_detail_proceed_standalone()
18         self.switch_to_main_frame(self)
19         try:
20             self.driver.switch_to.default_content()
21             logging.info("switched to main frame")
22         except Exception:
23             logging.info("Frame not available. Proceeding without it")
24         self.switch_to_cross_origin_frame(self)
25         try:
26             self.driver.switch_to.frame(self.driver.find_element_by_xpath("//object[2]"))
27             logging.info("switched to frame")
28         except:
29             self.driver.switch_to.frame(self.driver.find_element_by_xpath("//object[1]"))
30             logging.info("switched to frame")
31         except Exception:
32             logging.info("Frame not available. Proceeding without it")
33         self.__init__(self, webdriver):
34             self.driver = webdriver
35             close_need_help_popup(self.driver)
36             def click_on_upfront_deposit_standalone_mobile(self):
37                 try:
38                     self.switch_to_main_frame()
39                     self.switch_to_cross_origin_frame()
40                     self.driver.find_element_by_xpath(self.upfront_deposit_cost_no)
41                     webdriverWait(self.driver, 30).until(
42                         lambda d: C.exists_of_element_located((d, XPATH, self.upfront_deposit_cost_no)))
43                     self.driver.find_element_by_xpath(self.upfront_deposit_cost_no).click()
44                 except Exception:
45                     logging.info("Not found upfront deposit no btn")
46             def click_proceed_btn_to_confirm_standalone(self):
47                 self.driver.find_element_by_xpath(self.confirm_detail_proceed_standalone)
48                 webdriverWait(self.driver, 30).until(
49                     lambda d: C.exists_of_element_located((d, XPATH, self.confirm_detail_proceed_standalone)))
50                 self.driver.find_element_by_xpath(self.confirm_detail_proceed_standalone).click()
```

LIST OF SUCCESSFUL AUTOMATION TESTS

Test Case ID	Test Case Name	Test Case Description	Test Case Status	Test Case Date
TC_001	Verify the user can login successfully	Verify the user can login successfully with valid credentials	Pass	2023-01-01
TC_002	Verify the user can register successfully	Verify the user can register successfully with valid details	Pass	2023-01-01
TC_003	Verify the user can reset their password	Verify the user can reset their password successfully	Pass	2023-01-01
TC_004	Verify the user can update their profile	Verify the user can update their profile successfully	Pass	2023-01-01
TC_005	Verify the user can delete their account	Verify the user can delete their account successfully	Pass	2023-01-01

- Working document have been significantly effective within the BT team.
- Creates an accurate framework and ensures testing is done precisely.
- Automation testing times has significantly decreased due to the creation of scripts
- Successful transition from manual to automated testing due to the implementation of standalone URL (see above)

TOOL REQUIREMENTS

Manual Tester	Automated Tester
Test Case → Prepared in Excel	Technology Stack → Selenium & Python
WireMock → For Stubs	WireMock → For Stubs
Test Execution → Capture in Word	CATS → Automation Scripts
Jira → Upload Test pack	Git Bash → Keeping up to date with code and pushing created codes
	Jira → All test management including reporting
	IDE → All of the scripts will be written in this application

TEST TIMES BEFORE & AFTER ADDING STANDALONE URL

Last 5	Current Status	Avg. Runtime	Browser
	SUCCESS : Delay step complete	15 mins 19 secs	Chrome

BEFORE

Last 5	Current Status	Avg. Runtime	Browser
	SUCCESS : Delay step complete	2 mins 29 secs	Chrome

AFTER

IMPROVEMENTS



Git Blame v10.5.0

Wade Anderson | 1,942,347 | ★★★★★ (43)

See git blame information in the status bar.

Reload Required Disable Uninstall

This extension is enabled globally.

- Implementation of 'Git Blame' would have avoided a cascade of confusion within the team
- Improved communication and understanding of each piece of code & functionality of codes
- Resourceful application that could have been added to Visual Studio Code

I. Sommerville, Software Engineering 10th Edition, Pearson, 2015.
P. Pocatilu, "Automated software testing process," *Economy Informatics*, vol. 1, pp. 97-99, 2002.