|  |  |
| --- | --- |
| **Project Name: Project 1: Voting System Team#3** | |
| **Test Stage: Unit x\_\_ System \_\_** | **Test Date: 4/1/20** |
| **Test Case ID#: Plurality\_election\_record\_UT007** | **Name(s) of Testers: Colin Kluegel** |
| **Test Description:After several candidates are move to the winners list we need to check that we can move the remaining candidates to the losers list** | **Test file: plurality\_election\_record\_UT.cc**  **Method: TEST\_F(PluralityElectionRecordTests, MoveRemainingCandidatesToLosersList)** |
| **Automated: yes\_\_x\_ no \_\_\_** | **Indicate where are you storing the tests (what file) and the name of the method/functions being used.** |
| **Results: Pass \_\_\_x\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test: Preconditions for Test: 5 candidate objects and 5 ballot objects are created and put in candidate lists and ballot lists respectively. A new PluralityElectionRecord object is created with these lists** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
| 1 | Move first 2 candidates on non-elected list to winners list |  |  |  |  |
| 2 | Call MoveRemainingCandidatesToLosers  List to move the rest of the candidates to the losers list |  |  |  |  |
| 3 | Create losers\_lists by setting to what is returned by election\_record->GetLosersList() | Losers list |  |  |  |
| 4 | Check size of losers list |  | Size should be 3 | Size is 3, test passed |  |
| 5 | Create a candidate object and set it to the first candidate in the losers list | Loser candidate |  |  |  |
|  |  |  |  |  |  |
| 6 | Check that loser candidate is the correct candidate |  | Loser candidate should be candidate 3 | Loser candidate is candidate 3, test passed |  |
| 7 | Pop first candidate off of losers\_list, set next candidate on list to loser candidate |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| 8 | Check that loser candidate is the correct candidate |  | Loser candidate should be candidate 2 | Loser candidate is candidate 3, test passed |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| 9 | Pop first candidate off of losers\_list, set next candidate on list to loser candidate |  |  |  |  |
| 10 | Check that loser candidate is the correct candidate |  | Loser candidate should be candidate 1 | Loser is candidate 1, test passed |  |
|  |  |  |  |  |  |

**Post condition(s) for Test:**

First 2 candidates have been moved from the non-elected list to the winners list, the rest of the candidates are on the losers list

Project Name: The project #, name of your system, and the team#

Test Stage: Indicate whether it is a unit test or a system test.

Test Date: The date the test was performed.

Test Case ID#: A unique ID is required. Decide on a naming convention and use numbering. Example: Ballot\_Shuffle\_1

Name(s) of Testers: List the names of anyone involved in running this test case.

Test Description: Describe briefly the test objective.

Automated: Indicate if the test is completely automated or being checked manually. (If you have methods running the tests and checking results, select “yes”. If you are manually checking results, indicate manual by selecting the “no.”)

**Results:** Indicate if the test passed or failed.

**Step #:** You will be listing the test steps in order. This number is the step number in the process.

**Test Step Description:** Details of the test step.

**Test Data:** What the test data will be for this step. Be clear on what the input data will be. If using a specific file, be clear on the name.

**Expected Result:** What result are you expecting from the program component or system.

**Actual Result:** What result were returned based on the test.

**Post condition for Test:** What will be true after the test has been run? Has the state of the system changed in any way?

**Notes:** Comments and notesfor you and your team members.