|  |  |
| --- | --- |
| **Project Name: Project 1: Voting System Team#08** | |
| **Test Stage: Unit \_x\_ System \_\_** | **Test Date: 3/20/2023** |
| **Test Case ID#: Party\_01** | **Name(s) of Testers: Claire Maurus** |
| **Test Description:** Using file PartyTest.cpp, which is using google test, to test the the Constructor setting voteTotal to 0 and that the setter setVoteTotal() and getter getVoteTotal() work.  Ran with all files in the same directory. |  |
| **Automated: yes\_\_x\_ no \_\_\_** |  |
| **Results: Pass \_\_x\_\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:** Having Data that is passed into the constructor, the following data will be used to construct 2 objects used during testing(Candidate comma delineated string, part string):  (1) "Candidate A, Candidate B, Candidate C", "Party A"  (2) "A, B, C, D, E, G", "Testing" | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
|  |  |  |  |  |  |
| 1 | Instantiate two objects of type Party using information (1) and (2) | Passing in (1) and (2) info to constructor makes two objects testing1 and testing2 | Nothing | Nothing | Just for creating the objects so no current results to see |
| 2 | Call getVoteTotal() on testing1 and testing 2 objects | Fucntion called on the two testing objects, expect back integer values | For testing1 : integer 0  For testing2 : integer 0 | For testing1 : integer 0  For testing2 : integer 0 | This was verified by Google testing, means not votes sored and was initialized correctly |
| 3 | Called setVoteTotal() on both tesing objects then getVoteTotal() | For testing1 set votes to 10 and for testing2 set votes to 700000 | getVoteTotal() return:  For testing1: 10  For testing2: 700000 | getVoteTotal() return:  For testing1: 10  For testing2: 700000 | This matched as expected |
| 4 | Repeated above step on same two objects with new numbers | For testing1 set votes to 9999 and for testing2 set votes to 8 | getVoteTotal() return:  For testing1: 9999  For testing2: 8 | getVoteTotal() return:  For testing1: 9999  For testing2: 8 | This matched as expected |

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

At the end of the test the votes were set to 9999 for Party object testing1 and 8 for Party Object testing2 so they were correctly set after initialization and correctly updated.