

## Project Name: Project 1: Voting System

Team#11

**Test Stage:** Unit Testing

**Test Date:** 3/22/24

**Name(s) of Testers:** Arzab Bhattarai, Sam Bakri, Andy Wang, Wendell Relacion

**Test Case ID#:** UC\_001

**Test Description:** Testing from testing/Ballot\_Unittest.cpp. Testing TEST\_F(BallotTest, GetVotes). This checks to see if the correct amount of votes is actually returned from the getVotes() function.

Notes:

Comment every other TEST\_F other than TEST\_F(BallotTest, GetVotes). Additionally, when in the terminal cd into the testing folder.

Indicate where are you storing the tests (what file) and the name of the method/functions being used.

**Automated:** Yes

**Results:** Pass

### Preconditions for Test:

Ballot object is created and initialized.

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Run g++ -o my_tests Ballot_Unittest.cpp -lgtest -lgtest_main -pthread				
3	Run ./my_tests		Pass	Pass	
4					

### Post condition(s) for Test:

There is no error that occurs.

**Test Stage: Unit Testing**

**Test Date:** 3/22/24

**Test Case ID#:** UC\_002

**Name(s) of Testers:** Arzab Bhattarai, Sam Bakri, Andy Wang, Wendell Relacion

**Test Description:** Testing from testing/Ballot\_Unittest.cpp.  
Testing TEST\_F(BallotTest, SetVotes). This checks to see if the setVotes function is properly setting the vote for the ballot object

Notes:

Comment every other TEST\_F other than TEST\_F(BallotTest, SetVotes). Additionally, when in the terminal cd into the testing folder.

**Indicate where are you storing the tests (what file) and the name of the method/functions being used.**

**Automated:** Yes

**Results:** Pass

**Preconditions for Test:**

Ballot object is created and initialized.

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Run g++ -o my_tests Ballot_Unittest.cpp -lgtest -lgtest_main -pthread				
3	Run ./my_tests		Pass	Pass	
4					

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

There is no error that occurs.

**Test Stage: Unit Testing**

**Test Date:** 3/22/24

**Test Case ID#:** UC\_003

**Name(s) of Testers:** Arzab Bhattarai, Sam Bakri, Andy Wang, Wendell Relacion

**Test Description:** Testing from testing/Ballot\_Unittest.cpp.

Testing TEST\_F(BallotTest, IncrementVotes). This checks to see if the incrementVotes function is properly incrementing by 1 for a ballot object.

Notes:

Comment every other TEST\_F other than TEST\_F(BallotTest, IncrementVotes). Additionally, when in the terminal cd into the testing folder.

**Indicate where are you storing the tests (what file) and the name of the method/functions being used.**

**Automated:** Yes

**Results:** Pass

**Preconditions for Test:**

Ballot object is created and initialized.

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Run g++ -o my_tests Ballot_Unittest.cpp -lgtest -lgtest_main -pthread				
3	Run ./my_tests		Pass	Pass	
4					

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

There is no error that occurs.

**Test Stage: Unit Testing**

**Test Date:** 3/22/24

**Test Case ID#:** UC\_004

**Name(s) of Testers:** Arzab Bhattarai, Sam Bakri, Andy Wang, Wendell Relacion

**Test Description:** Testing from testing/Candidate\_Unittest.cpp. Testing TEST\_F(CandidateTest, GetCandidateName). This function checks to see if the getCandidateName() function returns the string name for the candidate.

Notes:

Comment every other TEST\_F other than TEST\_F(CandidateTest, GetCandidateName). Additionally, when in the terminal cd into the testing folder.

**Indicate where are you storing the tests (what file) and the name of the method/functions being used.**

**Automated:** Yes

**Results:** Pass

**Preconditions for Test:**

Candidate and party are initialized with test values.

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Run g++ -o my_tests Candidate_Unittest.cpp -lgtest -lgtest_main -pthread				
3	Run ./my_tests		Pass	Pass	
4					

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

There is no error that occurs.

**Test Stage: Unit Testing**

**Test Date:** 3/22/24

**Name(s) of Testers:** Arzab Bhattarai, Sam Bakri, Andy Wang, Wendell Relacion

**Test Case ID#:** UC\_005

**Test Description:** Testing from testing/Candidate\_Unittest.cpp.

Testing TEST\_F(CandidateTest, SetCandidateName). This function checks to see if the setCandidateName() function properly assigns a string name to a candidate object.

Notes:

Comment every other TEST\_F other than TEST\_F(CandidateTest, SetCandidateName). Additionally, when in the terminal cd into the testing folder.

**Indicate where are you storing the tests (what file) and the name of the method/functions being used.**

**Automated:** Yes

**Results:** Pass

**Preconditions for Test:**

Candidate and party are initialized with test values.

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Run g++ -o my_tests Candidate_Unittest.cpp -lgtest -lgtest_main -pthread				
3	Run ./my_tests		Pass	Pass	
4					

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

There is no error that occurs.

**Test Stage: Unit Testing**

**Test Date:** 3/22/24

**Name(s) of Testers:** Arzab Bhattarai, Sam Bakri, Andy Wang, Wendell Relacion

**Test Case ID#:** UC\_006

**Test Description:** Testing from testing/Candidate\_Unittest.cpp. Testing TEST\_F(CandidateTest, GetParty). This function checks to see if the getParty() function properly returns the party object that was assigned to that candidate

Notes:

Comment every other TEST\_F other than TEST\_F(CandidateTest,GetParty). Additionally, when in the terminal cd into the testing folder.

**Indicate where are you storing the tests (what file) and the name of the method/functions being used.**

**Automated:** Yes

**Results:** Pass

**Preconditions for Test:**

Candidate and party are initialized with test values.

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Run g++ -o my_tests Candidate_Unittest.cpp -lgtest -lgtest_main -pthread				
3	Run ./my_tests		Pass	Pass	
4					

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

There is no error that occurs.

**Test Stage: Unit Testing**

**Test Date:** 3/22/24

**Test Case ID#:** UC\_007

**Name(s) of Testers:** Arzab Bhattarai, Sam Bakri, Andy Wang, Wendell Relacion

**Test Description:** Testing from testing/Candidate\_Unittest.cpp. Testing TEST\_F(CandidateTest, SetParty). This function checks to see if the setParty() function is properly setting the party object.

Notes:

Comment every other TEST\_F other than TEST\_F(CandidateTest, SetParty). Additionally, when in the terminal cd into the testing folder.

**Indicate where are you storing the tests (what file) and the name of the method/functions being used.**

**Automated:** Yes

**Results:** Pass

**Preconditions for Test:**

Candidate and party are initialized with test values.

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Run g++ -o my_tests Candidate_Unittest.cpp -lgtest -lgtest_main -pthread				
3	Run ./my_tests		Pass	Pass	
4					

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

There is no error that occurs.

**Test Stage: Unit Testing**

**Test Date:** 3/22/24

**Test Case ID#:** UC\_008

**Name(s) of Testers:** Arzab Bhattarai, Sam Bakri, Andy Wang, Wendell Relacion

**Test Description:** Testing from testing/Candidate\_Unittest.cpp. Testing TEST\_F(CandidateTest, GetPartyName). This function checks to see if the getPartyName() function properly returns the string PartyName for the candidate object

Notes:

Comment every other TEST\_F other than TEST\_F(CandidateTest, getPartyName). Additionally, when in the terminal cd into the testing folder.

**Indicate where are you storing the tests (what file) and the name of the method/functions being used.**

**Automated:** Yes

**Results:** Pass

**Preconditions for Test:**

Candidate and party are initialized with test values.

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Run g++ -o my_tests Candidate_Unittest.cpp -lgtest -lgtest_main -pthread				
3	Run ./my_tests		Pass	Pass	
4					

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

There is no error that occurs.



**Test Stage: Unit Testing**

**Test Date:** 3/22/24

**Name(s) of Testers:** Arzab Bhattarai, Sam Bakri, Andy Wang, Wendell Relacion

**Test Case ID#:** UC\_009

**Test Description:** Testing from testing/Candidate\_Unittest.cpp. Testing TEST\_F(CandidateTest, SetPartyName). This function checks to see if the SetPartyName() function properly sets the party name of the candidate object by taking in the string name.

Notes:

Comment every other TEST\_F other than TEST\_F(CandidateTest, SetPartyName). Additionally, when in the terminal cd into the testing folder.

**Indicate where are you storing the tests (what file) and the name of the method/functions being used.**

**Automated:** Yes

**Results:** Pass

**Preconditions for Test:**

Candidate and party are initialized with test values.

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Run g++ -o my_tests Candidate_Unittest.cpp -lgtest -lgtest_main -pthread				
3	Run ./my_tests		Pass	Pass	
4					

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

There is no error that occurs.

**Test Stage:** Unit Testing

**Test Date:** 3/22/24

**Name(s) of Testers:** Arzab Bhattarai, Sam Bakri, Andy Wang, Wendell Relacion

**Test Case ID#:** UC\_010

**Test Description:** Testing from testing/CandidateOPL\_Unittesting.cpp. Testing TEST\_F(CandidateTest, GetRankingCandidates). This function checks to see if the GetRankingCandidates() function properly gets the rankings of the candidateOPL object by returning integer rankingCandidates.

Notes:

Comment every other TEST\_F other than TEST\_F(CandidateTest, GetRankingCandidates). Additionally, when in the terminal cd into the testing folder.

**Indicate where are you storing the tests (what file) and the name of the method/functions being used.**

**Automated:** Yes

**Results:** Pass

**Preconditions for Test:**

Ballot and candidate objects are initialized.

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Run g++ -o my_tests CandidateOPL_Unittesting.cpp -lgtest -lgtest_main -pthread				
3	Run ./my_tests		Pass	Pass	
4					

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

There is no error that occurs.

**Test Stage: Unit Testing**

**Test Date:** 3/22/24

**Name(s) of Testers:** Arzab Bhattarai, Sam Bakri, Andy Wang, Wendell Relacion

**Test Case ID#:** UC\_011

**Test Description:** Testing from testing/CandidateOPL\_Unittesting.cpp. Testing TEST\_F(CandidateTest, SetRankingCandidates). This function checks to see if the SetRankingCandidates(int rankingCandidates) function properly sets the rankings of the candidateOPL objects.

Notes:

Comment every other TEST\_F other than TEST\_F(CandidateTest, SetRankingCandidates). Additionally, when in the terminal cd into the testing folder.

**Indicate where are you storing the tests (what file) and the name of the method/functions being used.**

**Automated:** Yes

**Results:** Pass

**Preconditions for Test:**

Ballot and candidate objects are initialized.

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Run g++ -o my_tests CandidateOPL_Unittesting.cpp -lgtest -lgtest_main -pthread				
3	Run ./my_tests		Pass	Pass	
4					

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

There is no error that occurs.

**Test Stage:** Unit Testing

**Test Date:** 3/22/24

**Name(s) of Testers:** Arzab Bhattarai, Sam Bakri, Andy Wang, Wendell Relacion

**Test Case ID#:** UC\_012

**Test Description:** Testing from testing/CandidateOPL\_Unittesting.cpp. Testing TEST\_F(CandidateTest, GetBallot). This function checks to see if the GetBallot() function properly gets the ballot associated with the candidate.

Notes:

Comment every other TEST\_F other than TEST\_F(CandidateTest, GetBallot). Additionally, when in the terminal cd into the testing folder.

**Indicate where are you storing the tests (what file) and the name of the method/functions being used.**

**Automated:** Yes

**Results:** Pass

**Preconditions for Test:**

Ballot and candidate objects are initialized.

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Run g++ -o my_tests CandidateOPL_Unittesting.cpp -lgtest -lgtest_main -pthread				
3	Run ./my_tests		Pass	Pass	
4					

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

There is no error that occurs.

**Test Stage: Unit Testing**

**Test Date:** 3/22/24

**Name(s) of Testers:** Arzab Bhattarai, Sam Bakri, Andy Wang, Wendell Relacion

**Test Case ID#:** UC\_013

**Test Description:** Testing from testing/CandidateOPL\_Unittesting.cpp. Testing TEST\_F(CandidateTest, SetBallot). This function checks to see if the SetBallot() function properly sets the ballot associated with the candidate.

Notes:

Comment every other TEST\_F other than TEST\_F(CandidateTest, SetBallot). Additionally, when in the terminal cd into the testing folder.

**Indicate where are you storing the tests (what file) and the name of the method/functions being used.**

**Automated:** Yes

**Results:** Pass

**Preconditions for Test:**

Ballot and candidate objects are initialized.

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Run g++ -o my_tests CandidateOPL_Unittesting.cpp -lgtest -lgtest_main -pthread				
3	Run ./my_tests		Pass	Pass	
4					

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

There is no error that occurs.

**Test Stage: Unit Testing**

**Test Date:** 3/22/24

**Test Case ID#:** UC\_014

**Name(s) of Testers:** Arzab Bhattarai, Sam Bakri, Andy Wang, Wendell Relacion

**Test Description:** Testing from testing/Party\_Unittest.cpp. Testing TEST\_F(PartyTest, GetPartyName). This function checks to see if the GetPartyName() function properly gets the name of the party object.

Notes:

Comment every other TEST\_F other than TEST\_F(PartyTest, GetPartyname). Additionally, when in the terminal cd into the testing folder.

**Indicate where are you storing the tests (what file) and the name of the method/functions being used.**

**Automated:** Yes

**Results:** Pass

**Preconditions for Test:**

Party object is properly initialized

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Run g++ -o my_tests Party_Unittest.cpp -lgtest -lgtest_main -pthread				
3	Run ./my_tests		Pass	Pass	
4					

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

There is no error that occurs.

**Test Stage: Unit Testing**

**Test Date:** 3/22/24

**Test Case ID#: UC\_015**

**Name(s) of Testers:** Arzab Bhattarai, Sam Bakri, Andy Wang, Wendell Relacion

**Test Description:** Testing from testing/Party\_Unittest.cpp.  
Testing TEST\_F(PartyTest, SetPartyName). This function checks to see if the SetPartyName() function properly sets the name of the party object.

Notes:

Comment every other TEST\_F other than TEST\_F(PartyTest, SetPartyname). Additionally, when in the terminal cd into the testing folder.

**Indicate where are you storing the tests (what file) and the name of the method/functions being used.**

**Automated: Yes**

**Results: Pass**

**Preconditions for Test:**

Party object is properly initialized

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Run g++ -o my_tests Party_Unittest.cpp -lgtest -lgtest_main -pthread				
3	Run ./my_tests		Pass	Pass	
4					

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

There is no error that occurs.

**Test Stage: Unit Testing**

**Test Date:** 3/22/24

**Test Case ID#:** UC\_016

**Name(s) of Testers:** Arzab Bhattarai, Sam Bakri, Andy Wang, Wendell Relacion

**Test Description:** Testing from testing/Party\_Unittest.cpp.  
Testing TEST\_F(PartyTest, GetCandidates). This function checks to see if the GetCandidates() function properly gets the candidate associated with the party object.

Notes:

Comment every other TEST\_F other than TEST\_F(PartyTest, GetCandidates). Additionally, when in the terminal cd into the testing folder.

**Indicate where are you storing the tests (what file) and the name of the method/functions being used.**

**Automated:** Yes

**Results:** Pass

**Preconditions for Test:**

Party object is properly initialized

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Run g++ -o my_tests Party_Unittest.cpp -lgtest -lgtest_main -pthread				
3	Run ./my_tests		Pass	Pass	
4					

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

There is no error that occurs.



**Test Stage: Unit Testing**

**Test Date:** 3/22/24

**Name(s) of Testers:** Arzab Bhattarai, Sam Bakri, Andy Wang, Wendell Relacion

**Test Case ID#:** UC\_017

**Test Description:** Testing from testing/Party\_Unittest.cpp. Testing TEST\_F(PartyTest, BallotFunctions). This function checks to see if the BallotFunctions() function properly performs the setVotes(), setBallots(), getBallots(), and also getVotes() functions.

Notes:

Comment every other TEST\_F other than TEST\_F(PartyTest, BallotFunctions). Additionally, when in the terminal cd into the testing folder.

**Indicate where are you storing the tests (what file) and the name of the method/functions being used.**

**Automated:** Yes

**Results:** Pass

**Preconditions for Test:**

Party object is properly initialized

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Run g++ -o my_tests Party_Unittest.cpp -lgtest -lgtest_main -pthread				
3	Run ./my_tests		Pass	Pass	
4					

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

There is no error that occurs.

**Test Stage: Unit Testing**

**Test Date:** 3/22/24

**Test Case ID#:** UC\_018

**Name(s) of Testers:** Arzab Bhattarai, Sam Bakri, Andy Wang, Wendell Relacion

**Test Description:** Testing from testing/Party\_Unittest.cpp.  
Testing TEST\_F(PartyTest, RemainderFunctions). This function checks to see if the BallotFunctions() function properly performs the getRemainder() and setRemainder() functions.

Notes:

Comment every other TEST\_F other than TEST\_F(PartyTest, RemainderFunctions). Additionally, when in the terminal cd into the testing folder.

**Indicate where are you storing the tests (what file) and the name of the method/functions being used.**

**Automated:** Yes

**Results:** Pass

**Preconditions for Test:**

Party object is properly initialized

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Run g++ -o my_tests Party_Unittest.cpp -lgtest -lgtest_main -pthread				
3	Run ./my_tests		Pass	Pass	
4					

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

There is no error that occurs.

**Test Stage: Unit Testing**

**Test Date:** 3/22/24

**Test Case ID#:** UC\_019

**Name(s) of Testers:** Arzab Bhattarai, Sam Bakri, Andy Wang, Wendell Relacion

**Test Description:** Testing from testing/Party\_Unittest.cpp. Testing TEST\_F(PartyTest, AllocatedSeatsFunctions). This function checks to see if the BallotFunctions() function properly performs the getAllocatedSeats() and setAllocatedSeats() functions.

Notes:

Comment every other TEST\_F other than TEST\_F(PartyTest, AllocatedSeatsFunctions). Additionally, when in the terminal cd into the testing folder.

**Indicate where are you storing the tests (what file) and the name of the method/functions being used.**

**Automated:** Yes

**Results:** Pass

**Preconditions for Test:**

Party object is properly initialized

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Run g++ -o my_tests Party_Unittest.cpp -lgtest -lgtest_main -pthread				
3	Run ./my_tests		Pass	Pass	
4					

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

There is no error that occurs.

**Test Stage:** Unit Testing

**Test Date:** 3/22/24

**Name(s) of Testers:** Arzab Bhattarai, Sam Bakri, Andy Wang, Wendell Relacion

**Test Case ID#:** UC\_020

**Test Description:** Testing from testing/OPLParty\_Unittest.cpp. Testing TEST\_F(OPLPartyTest, CandidateVoteFunctions). This function checks to see if the CandidateVoteFunctions() function properly performs the getCandidateVote() and setCandidateVote() functions.

Notes:

Comment every other TEST\_F other than TEST\_F(OPLPartyTest, CandidateVoteFunctions). Additionally, when in the terminal cd into the testing folder.

**Indicate where are you storing the tests (what file) and the name of the method/functions being used.**

**Automated:** Yes

**Results:** Pass

**Preconditions for Test:**

OPLParty and CandidateOPL objects are properly initialized

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Run g++ -o my_tests OPLParty_Unittest.cpp -lgtest -lgtest_main -pthread				
3	Run ./my_tests		Pass	Pass	
4					

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

There is no error that occurs.

**Test Stage: Unit Testing**

**Test Date:** 3/22/24

**Name(s) of Testers:** Arzab Bhattarai, Sam Bakri, Andy Wang, Wendell Relacion

**Test Case ID#:** UC\_021

**Test Description:** Testing from testing/OPLParty\_Unittest.cpp. Testing TEST\_F(OPLPartyTest, SetCandidateRankings). This function checks to see if the SetCandidateRankings() function properly sets the rankings of the candidate objects.

Notes:

Comment every other TEST\_F other than TEST\_F(OPLPartyTest, SetCandidateRankings). Additionally, when in the terminal cd into the testing folder.

**Indicate where are you storing the tests (what file) and the name of the method/functions being used.**

**Automated:** Yes

**Results:** Pass

**Preconditions for Test:**

OPLParty and CandidateOPL objects are properly initialized

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Run g++ -o my_tests OPLParty_Unittest.cpp -lgtest -lgtest_main -pthread				
3	Run ./my_tests		Pass	Pass	
4					

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

There is no error that occurs.

**Test Stage: Unit Testing**

**Test Date:** 3/22/24

**Name(s) of Testers:** Arzab Bhattarai, Sam Bakri, Andy Wang, Wendell Relacion

**Test Case ID#:** UC\_022

**Test Description:** Testing from testing/OPLParty\_Unittest.cpp. Testing TEST\_F(OPLPartyTest, GetRankings). This function checks to see if the GetRankings() function properly gets the rankings for the candidate objects.

Notes:

Comment every other TEST\_F other than TEST\_F(OPLPartyTest, GetRankings). Additionally, when in the terminal cd into the testing folder.

**Indicate where are you storing the tests (what file) and the name of the method/functions being used.**

**Automated:** Yes

**Results:** Pass

**Preconditions for Test:**

OPLParty and CandidateOPL objects are properly initialized

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Run g++ -o my_tests OPLParty_Unittest.cpp -lgtest -lgtest_main -pthread				
3	Run ./my_tests		Pass	Pass	
4					

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

There is no error that occurs.

**Test Stage: Unit Testing**

**Test Date:** 3/22/24

**Name(s) of Testers:** Arzab Bhattarai, Sam Bakri, Andy Wang, Wendell Relacion

**Test Case ID#:** UC\_023

**Test Description:** Testing from testing/OPLParty\_Unittest.cpp. Testing TEST\_F(OPLPartyTest, AddOPLCandidate). This function checks to see if the AddOPLCandidate() function properly adds a candidate object to the candidates vector.

Notes:

Comment every other TEST\_F other than TEST\_F(OPLPartyTest, AddOPLCandidate). Additionally, when in the terminal cd into the testing folder.

**Indicate where are you storing the tests (what file) and the name of the method/functions being used.**

**Automated:** Yes

**Results:** Pass

**Preconditions for Test:**

OPLParty and CandidateOPL objects are properly initialized

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Run g++ -o my_tests OPLParty_Unittest.cpp -lgtest -lgtest_main -pthread				
3	Run ./my_tests		Pass	Pass	
4					

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

There is no error that occurs.

**Test Stage: Unit Testing**

**Test Date:** 3/22/24

**Name(s) of Testers:** Arzab Bhattarai, Sam Bakri, Andy Wang, Wendell Relacion

**Test Case ID#:** UC\_024

**Test Description:** Testing from testing/Seat\_Unittest.cpp.

Testing TEST\_F(SeatTest, GetSeats). This function checks to see if the GetSeats() function correctly gets the seats object.

Notes:

Comment every other TEST\_F other than TEST\_F(SeatTest, GetSeats). Additionally, when in the terminal cd into the testing folder.

**Indicate where are you storing the tests (what file) and the name of the method/functions being used.**

**Automated:** Yes

**Results:** Pass

**Preconditions for Test:**

Seat object is properly initialized

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Run g++ -o my_tests Seat_Unittest.cpp -lgtest -lgtest_main -pthread				
3	Run ./my_tests		Pass	Pass	
4					

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

There is no error that occurs.



**Test Stage: Unit Testing**

**Test Date:** 3/22/24

**Test Case ID#:** UC\_025

**Name(s) of Testers:** Arzab Bhattarai, Sam Bakri, Andy Wang, Wendell Relacion

**Test Description:** Testing from testing/Seat\_Unittest.cpp.

Testing TEST\_F(SeatTest, SetSeats). This function checks to see if the SetSeats() function correctly sets the seats object.

Notes:

Comment every other TEST\_F other than TEST\_F(SeatTest, SetSeats). Additionally, when in the terminal cd into the testing folder.

**Indicate where are you storing the tests (what file) and the name of the method/functions being used.**

**Automated:** Yes

**Results:** Pass

**Preconditions for Test:**

Seat object is properly initialized

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Run g++ -o my_tests Seat_Unittest.cpp -lgtest -lgtest_main -pthread				
3	Run ./my_tests		Pass	Pass	
4					

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

There is no error that occurs.

**Test Stage: Unit Testing**

**Test Date:** 3/22/24

**Name(s) of Testers:** Arzab Bhattarai, Sam Bakri, Andy Wang, Wendell Relacion

**Test Case ID#:** UC\_026

**Test Description:** Testing from testing/Seat\_Unittest.cpp.

Testing TEST\_F(SeatTest, IncSeats). This function checks to see if the IncSeats() function correctly increments the seat variable in the seat object.

Notes:

Comment every other TEST\_F other than TEST\_F(SeatTest, IncSeats). Additionally, when in the terminal cd into the testing folder.

**Indicate where are you storing the tests (what file) and the name of the method/functions being used.**

**Automated:** Yes

**Results:** Pass

**Preconditions for Test:**

Seat object is properly initialized

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Run g++ -o my_tests Seat_Unittest.cpp -lgtest -lgtest_main -pthread				
3	Run ./my_tests		Pass	Pass	
4					

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

There is no error that occurs.

**Test Stage: System Testing**

**Test Date:** 3/22/24

**Name(s) of Testers:** Arzab Bhattarai, Sam Bakri, Andy Wang, Wendell Relacion

**Test Case ID#: ST\_01**

**Test Description:** Testing from testing/systems\_test. Testing TEST\_F(ElectionSystemTest, RunOPLSystemTest).

This function checks to see if the run function properly works for the case that the Ballot File is an election of type OPL.

Notes:

Comment every other TEST\_F other than

TEST\_F(ElectionSystemTest, RunOPLSystemTest).

Additionally, when in the terminal cd into the testing folder.

**Indicate where are you storing the tests (what file) and the name of the method/functions being used.**

**Automated: Yes**

**Results: Pass**

**Preconditions for Test:**

filename string is given a value of the csv file

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Run g++ -o my_tests Election_systemTest.cpp -lgtest -lgtest_main -pthread				
3	Run ./my_tests		Pass	Pass	
4					

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

There is no error that occurs.

**Test Stage: System Testing**

**Test Date:** 3/22/24

**Name(s) of Testers:** Arzab Bhattarai, Sam Bakri, Andy Wang, Wendell Relacion

**Test Case ID#:** ST\_002

**Test Description:** Testing from testing/systems\_test. Testing TEST\_F(ElectionSystemTest, RunCPLSystemTest).

This function checks to see if the run function properly works for the case that the Ballot File is an election of type CPL.

Notes:

Comment every other TEST\_F other than

TEST\_F(ElectionSystemTest, RunCPLSystemTest).

Additionally, when in the terminal cd into the testing folder.

**Indicate where are you storing the tests (what file) and the name of the method/functions being used.**

**Automated:** Yes

**Results:** Pass

**Preconditions for Test:**

filename string is given a value of the csv file

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Run g++ -o my_tests Election_systemTest.cpp -lgtest -lgtest_main -pthread				
3	Run ./my_tests		Pass	Pass	
4					

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

There is no error that occurs.