Team 14
Cassandra Chanthamontry (chant077)
Jake Nippert (nippe014)
Meghann Silagan (silag001)
Christine Tsai (tsaix223)

Use Cases

1. Provides file for the program to run

Name	Provides file for the program to run
ID	UC_001
Description	The user will provide a file for the program in order for it to run and give an output.
Actors	Tester, electoral officer, software engineer
Organizational Benefits	This increase efficiency as it allows the actors to be able to get an output quickly by providing a file instead of manually entering the information into the program in order to save the user time.
Frequency of Use[1]	They must provide a file during every run of the program.
Triggers	The user provides the file name as a command line argument.
Preconditions	The program must be compiled. The file provided must be a csv file. The program and the file are in the same directory.
Postconditions	The program will run with the correct information from the provided file.
Main Course	*1. The user receives the file containing the vote information as a csv file.* *2. The actor provides the file name as a command line argument* 3. The program parse through the csv file and provides an output and audit file.
Alternate Courses	AC1 The user did not provide a file name while running the program 1. The program prompts the user for the filename 2. Return to main course 3.
Exceptions	*EX1 The program cannot find file* *1. Notifies the user that the file is not found.* *EX2The file has an unexpected MIME type* *1. Notifies the user that the file is of wrong MIME type*

2. View Election Results

Name	View Election Results
ID	UC_002
Description	After the program is run, the user is able to view the audit file containing the election information at the time (e.g. Type of Voting, Number of Candidates, Candidates, Number of Ballots, etc).
Actors	Election Official, Tester, Software Developer *Is the audit file made public?* *Do other parties access or view the audit file?*
Organizational Benefits	Viewing the audit file will allow replication of the election as it shows how the election progressed thus automating the process of gathering election information. This optimizes the time needed in the case of a replication of the election is required.
Frequency of Use[1]	This is done after running the program for a given election. The audit file may be viewed multiple times if necessary as it is not overwritten because the filename will contain a timestamp. *Is there an extent to which the file can be accessed (ie is it deleted after a certain amount of time? Is the file allowed to be viewed multiple times?)*
Triggers	Clicking the file in the file explorer. Opening the filename in the command line. *Does the audit file need to display/open automatically upon completion of running the program? Is it triggered by anything other than just clicking the file in the file explorer? How is the user opening/viewing the file? Can a user just open it in a text editor? What file format is the file in?*
Preconditions	The audit file is produced by the program. *What application/program is the user using to view the audit file?*
Postconditions	The user has viewed the audit file. The audit file has been opened/visited.
Main Course	1. The user runs the program and the audit file is generated 2. *What if a user wants to access an audit file already generated? Is this even allowed? (see AC)* 3. *Does the audit file open automatically?* 4. *Does the program require the user to open the file via the file explorer, command line, both?* 5. *Is the audit file viewed in specific applications or does the*

	user designate which program to view the audit file in?
Alternate Courses	*If a previously generated audit file can be viewed, how does the user open the audit file? Is this through the command line? System? File explorer? Other application?* *Can a party who isn't a direct user of the program still access the audit file? If so, how do they access the file? Log in to local machine? Shared folder or drive?*
Exceptions	The file is already opened.

3. Instant Runoff Voting

Name	Instant Runoff Voting
ID	UC_003
Description	The user selects the Instant Runoff (IRV) Voting Method to calculate the winning candidate from the ballot file
Actors	Tester, electoral officer, software engineer
Organizational Benefits	This increase efficiency as it allows the user to be able to get an output quickly by providing a file instead of calculating the winner by hand using the Instant Runoff Voting algorithm.
Frequency of Use[1]	This program will be run multiple times during the year at normal elections and special elections times. Depending on the actor the program may be run at least once to multiple times to check the accuracy of the output results.
Triggers	*In the file where will it indicate what type of voting system algorithm should be executed?*
Preconditions	The ballot file must be a CSV file and must be in the same directory as the program.
Postconditions	The ballot file will be parsed to get the information needed to use the IRV algorithm. This will display the winner on the console as well as provide an audit file.
Main Course	1. The system parses through the CSV file 2. The system counts the number one preferences. The candidate who received the most number one preferences/majority is declared the winner 3. An audit file will be produced

Alternate Courses	AC1 No candidate receives majority 1. The candidate who receives the fewest votes is eliminated 2. The ballots of supporters of this candidate are then transferred to whichever of the remaining candidates they marked as their number 2 choices 3. Go to MC 2. AC2 Fewest amount of votes Candidates are tied 1. Use a random coin flip to decide which candidate is eliminated. 2. Go to AC1
Exceptions	EX1: User does not provide ballot file 1. The system prompts for the ballot file EX2: User choose an algorithm that is not IRV or OPL 1. The system prompts for either IRV or OPL to be chosen.

4. Open Party List Voting

Name	Open Party List Voting
ID	UC_004
Description	The user selects the Open Party List (OPL) Voting Method to calculate the winning candidate from the ballot file
Actors	Tester, electoral officer, software engineer
Organizational Benefits	This increase efficiency as it allows the user to be able to get an output quickly by providing a file instead of calculating the winner by hand using the Open Party List Voting algorithm.
Frequency of Use[1]	This program will be run multiple times during the year at normal elections and special elections times. Depending on the actor the program may be run at least once to multiple times to check the accuracy of the output results.
Triggers	*In the file where will it indicate what type of voting system algorithm should be executed?*
Preconditions	The ballot file must be a CSV file and must be in the same directory as the program.
Postconditions	The ballot file will be parsed to get the information needed to use the OPL algorithm. This will display the winner on the console as well as provide an audit file.
Main Course	1. The system parses through the CSV file

	2. System calculates a quota by taking the total number of votes and dividing it by the number of seats 3. System calculates how many seats a party get by dividing its total amount of votes by the quota 4. The party with the largest remainder after the first allocation of seats is completed are allocated the remaining seats 5. An audit file will be produced
Alternate Courses	*AC1: No Remainder* *1. System calculates quota and allocates all seats.*
Exceptions	EX1: User does not provide ballot file 2. The system prompts for the ballot file EX2: User choose an algorithm that is not IRV or OPL 2. The system prompts for either IRV or OPL to be chosen.

5. Produce Audit File for Instant Runoff Voting

Name	Produce Audit File for Instant Runoff Voting
ID	UC_005
Description	Once the Voting System program finishes calculating the correct election results, the audit file containing instant runoff voting information is created and output to the folder containing the program.
Actors	Porting system (system which runs file out to operating system)
Organizational Benefits	The audit file will provide the user with more detailed information about the results from the instant runoff voting election.
Frequency of Use[1]	One audit file is produced every time the program is run for an instant runoff voting election.
Triggers	When the program is run and the election results are complete, the audit file is created and placed in the correct directory.
Preconditions	The program must finish all the calculations that will go into the audit file. The directory containing the program must be available for the audit file to be placed into.
Postconditions	The audit file will contain organized information gathered from the election results. The audit file will be placed in the program's directory and will be available for viewing but not editing.
Main Course	1. The system runs and finishes calculations. 2. An audit file is created.

	3. Relevant instant runoff voting information is placed in the audit file. [Information about when a candidate is removed, and how their votes are tallied is indicated so that a completely traceable election can be drawn] 4. The audit file is placed in the same directory as the program. 5. User opens and views audit file.
Alternate Courses	AC1 The user ends the program before the system finishes calculations. 1. The program closes. 2. The user can return to main course 1 after running the program again.
Exceptions	EX1 The directory containing the program cannot be found. 1. The program notifies the user of the error. 2. The program closes.

6. Produce Audit File for Open Party List Voting

Name	Produce Audit File for Open Party List Voting
ID	UC_006
Description	Once the Voting System program finishes calculating the correct election results, the audit file containing open party list voting information is created and output to the folder containing the program.
Actors	Porting system (system which runs file out to operating system)
Organizational Benefits	The audit file will provide the user with more detailed information about the results from the open party list voting election.
Frequency of Use[1]	One audit file is produced every time the program is run for an open party list voting election.
Triggers	When the program is run and the election results are complete, the audit file is created and placed in the correct directory.
Preconditions	The program must finish all the calculations that will go into the audit file. The directory containing the program must be available for the audit file to be placed into.
Postconditions	The audit file will contain organized information gathered from the election results. The audit file will be placed in the program's directory and will be available for viewing but not editing.
Main Course	1. The system runs and finishes calculations.

	2. An audit file is created. 3. Relevant open party list voting information is placed in the audit file. [Information about the first round of seat allocations and the final round using remainders from the first round are noted so it is possible for the election to be traced] 4. The audit file is placed in the same directory as the program. 5. User opens and views audit file.
Alternate Courses	AC1 The user ends the program before the system finishes calculations. 3. The program closes. 4. The user can return to main course 1 after running the program again.
Exceptions	EX1 The directory containing the program cannot be found. 3. The program notifies the user of the error. 4. The program closes.

7. Parse File IRV

Name	Parse File IRV
ID	UC_007
Description	Once the data from the user specified file name is opened and read, information is parsed in an agreed format for instant runoff voting.
Actors	System
Organizational Benefits	The user does not have to manually enter the information in, in regards to the type of voting system (instant runoff), number of candidates, the candidates, the number of ballots in the file, and the ballots themselves. This saves the User significant time.
Frequency of Use[1]	Parsing the file should be completed once per election run.
Triggers	When the file name is provided to the system.
Preconditions	The file name is provided by the User and the file has all of the requisite lines.
Postconditions	All data objects have been created and the proper algorithm for our system is selected to be run on the information provided.
Main Course	1. Open file specified by user 2. Read in first line to determine voting system

	3. Read in second line to determine the number of candidates 4. Read in the third line for name of candidates 5. Read in the fourth line for the number of ballots 6. Read in the corresponding number of ballots
Alternate Courses	See UC_008 for OPLV selected
Exceptions	There should be no errors in the information of the file. EX1 If an error occurs while reading in the file, the file will be closed and reopened to attempt to read in the information and parse again.

8. Parse File OPLV

Name	Parse File OPLV
ID	UC_008
Description	Once the data from the user specified file name is opened and read, information is parsed in an agreed format for open party list voting.
Actors	System
Organizational Benefits	The user does not have to manually enter the information in, in regards to the type of voting system (open party list voting), number of candidates, the candidates and the party they are in, the number of seats available, the number of ballots, and the ballots themselves. This saves the User significant time.
Frequency of Use[1]	Parsing the file should be completed once per election run.
Triggers	When the file name is provided to the system
Preconditions	The file name is provided by the User and the file has all of the requisite lines.
Postconditions	All data objects have been created and the proper algorithm for our system is selected to be run on the information provided.
Main Course	1. Open file specified by the user 2. Read in first line to determine voting system 3. Read in second line to determine the number of candidates 4. Read in the third line for name of candidates and their respective party

	5. Read in the fourth line for the number of seats 6. Read in the fifth line for the number of ballots 7. Read in the corresponding number of ballots
Alternate Courses	See UC_007 for IRV selected
Exceptions	There should be no errors in the information of the file. EX1 If an error occurs while reading in the file, the file will be closed and reopened to attempt to read in the information and parse again.