Software Architecture Document:

360 Project: Election Software Application

Brandi Durham, Austin Hollis, Justin Willis

Table Of Contents:

Vision

Our vision for this project is to make a secure, easy to use e-election software application with the following features:

- Easy to use and interact with
- Keeps information and votes secure
- Allows for immediate voting tally at any time
- Unbiased candidate display
- Easy to maintain
- Quick and easy to understand instructions

We understand each stakeholder has unique requirements and specification for the election software, and we hope to meet each of these. While there may be some speculation around election software, we hope that by outlining exactly how we plan to implement the software we can show confidence in our product. In each of the use cases below, we have outlined what we think each stakeholder will be interested in. These may change after meeting with the client.

Use Cases

Listed below are our four use cases. These guide our functional and nonfunctional requirements and explains how our application handles the requirements.

Use Case 1: User Votes

Scope: Election Software application

Level: user goal

Primary Actor: User

Stakeholders and Interests:

- **User**: Want accurate results, the person they votes for to receive a vote, fast and easy-to-use interface, privacy of personal information and who they votes for, quick turnaround on results.
- **Candidates**: Want accurate results, equal representation (alphabetical order, no bold, same fonts), fast and easy to use interface process, quick turnaround on results.
- **Government/Business**: Want inexpensive in terms of resources and management, easy to maintain, accurate results, quick turnaround on results.
- **Election Official**: Want smooth setup, easy to maintain, proper working application, easy to follow instructions, quick process for each user.

Preconditions: Election official has application up and running, user is registered before current day.

Success Guarantee: Vote is counted, accurate vote, information is secure.

Main Success Scenario (or Basic Flow):

1. User arrives and logs in at an available computer
2. Vote *repeat for number of categories
3. System shows review page of choices

4. User clicks confirm button on review pages
5. System logs choices in database, or adds to queue to add to database
6. “Thank you” page appears indicating user can leave and vote has been processed
7. User leaves

Extensions (or Alternative Flows):

1. User is not registered
   - 1.1. Send user to registration page
   - 1.2. Resume voting process
2. Wrong login credentials
   - 2.1. Tell user information is incorrect and have them try again (set number of attempts)
   - 2.2. If user cannot login, screen locks and waits for election official
3. Trying to vote before time
   - 3.1. Tells user that it’s too early
   - 3.2. Election Official would reset screen to home
4. Application freezes
   - 4.1. Databases store after each vote
   - 4.2. When the users re-logs in, jumps to unanswered category

Special Requirements:

- Computer with keyboard and mouse
- Processing time for vote to register in database is with 5 mins.

Technology and Data Variations List:

- Election Official override by keyboard code
- Vote by clicking
- Log in by entering name and social security number

Frequency of Occurrence: Could be nearly continuous.

Open Issues:

- When is voting open?

Use Case 2: User Registers.

Scope: Election Software application

Level: user goal

Primary Actor: User

Stakeholders and Interests:

- **User**: Fast and easy to use interface, privacy of personal informations, clear registration information, secure database.
- **Candidates**: Want voters to have a fast and easy registration process, voters given clear registration information so they can vote.
- **Government/Business**: Want Inexpensive in terms of resources and management , easy to maintain, secure database.
- **Election Official**: Want smooth setup, easy to maintain, proper working application, easy to follow instructions, quick process for each user.

Preconditions: Election Offical has application up and running, and user is not registered to vote already.

Success Guarantee: User is able to register to vote, vote is counted, accurate vote, information is secure.

Main Success Scenario (or Basic Flow) :

1. User arrives
2. User chooses to register at an available computer
3. User inputs personal identification (Ex: first name, last name, DOB, SSN, etc.)
4. User clicks submit to confirm registration
5. Application tells user they are registered
6. User is notified that they are not allowed to vote on the same day as registration but is told when they can vote along with when voting ends
7. User returns to vote on appropriate day

Extensions (or Alternative Flows):

1. Incorrect Registration Information
    - 1.1. User may be asked to re-enter information is not correct
2. Application freezes
    - 2.1. Application doesn't store registration data until user clicks submit
    - 2.2. If application freezes while user is registering then data is not saved and user is prompted to enter information again

Special Requirements:

- Computer with keyboard and mouse
- Processing time for vote to register in database is within 5 mins.

Technology and Data Variations List:

- Election official override by keyboard code

Frequency of Occurrence: Could be nearly continuous.

Open Issues:

- When is voting open?
- Incorrect information?

Use Case 3: User Checks Registration Status.

Scope: Election Software application

Level: user goal

Primary Actor: User

Stakeholders and Interests:

- **User**: Fast and easy to use interface, privacy of personal informations, clear registration information, secure database.
- **Candidates**: Want voters to have a fast and easy registration process, voters given clear registration information so they can vote.
- **Government/Business**: Want Inexpensive in terms of resources and management , easy to maintain, secure database.
- **Election Official**: Want smooth setup, easy to maintain, proper working application, easy to follow instructions, quick process for each user.

Preconditions: Election official has application up and running.
Success Guarantee: User is able to check registration status, information is secure.
Main Success Scenario (or Basic Flow) :

1. User arrives
2. User chooses to check registration status at an available computer
3. User inputs personal identification (Ex: first name, last name, DOB, SSN, etc.)
4. User clicks submit
5. Application tells user if they are registered or not
6. User is notified that they are not allowed to vote on the same day as registration but is told when they can vote along with when voting ends
7. User returns to vote on appropriate day

Extensions (or Alternative Flows):

1. Incorrect Registration Information
   1.1. User may be asked to re-enter information is not correct
2. Application freezes
   2.1. Application doesn't store registration data until user clicks submit
   2.2. If application freezes while user is registering then data is not saved and user is prompted to enter information again

Special Requirements:

- Computer with keyboard and mouse
- Processing time for vote to register in database is within 5 mins.

Technology and Data Variations List:

- Election official override by keyboard code

Frequency of Occurrence: Could be nearly continuous.
Open Issues:

- When is voting open?
- Incorrect information?

Use Case 4: Election Official Tallies Votes
Scope: Election Software application
Level: admin goal
Primary Actor: Election Official
Stakeholders and Interests:

- **User**: Fast and easy to use interface, privacy of personal informations, clear registration information, secure database.
- **Candidates**: Want voters to have a fast and easy registration process, voters given clear registration information so they can vote.
- **Government/Business**: Want Inexpensive in terms of resources and management , easy to maintain, secure database.
- **Election Official**: Want smooth setup, easy to maintain, proper working application, easy to follow instructions, quick process for each user.

Preconditions: Election official has application up and running.
Success Guarantee: Election official is able to login and see an unofficial tally of the votes.

Main Success Scenario (or Basic Flow) :

1. Election Official arrives
2. Election Official chooses to get the unofficial tally of votes at an available computer
3. Election Official inputs identification (Ex: first name, last name, password, etc.)
4. Official clicks submit
5. Application tells official the unofficial tally

Extensions (or Alternative Flows):

1. Incorrect Registration Information
   - 1.1. User may be asked to re-enter information is not correct
2. Application freezes
   - 2.1. Application doesn't store registration data until user clicks submit
   - 2.2. If application freezes while user is registering then data is not saved and user is prompted to enter information again

Special Requirements:

- Computer with keyboard and mouse
- Processing time for vote to register in database is within 5 mins.

Technology and Data Variations List:

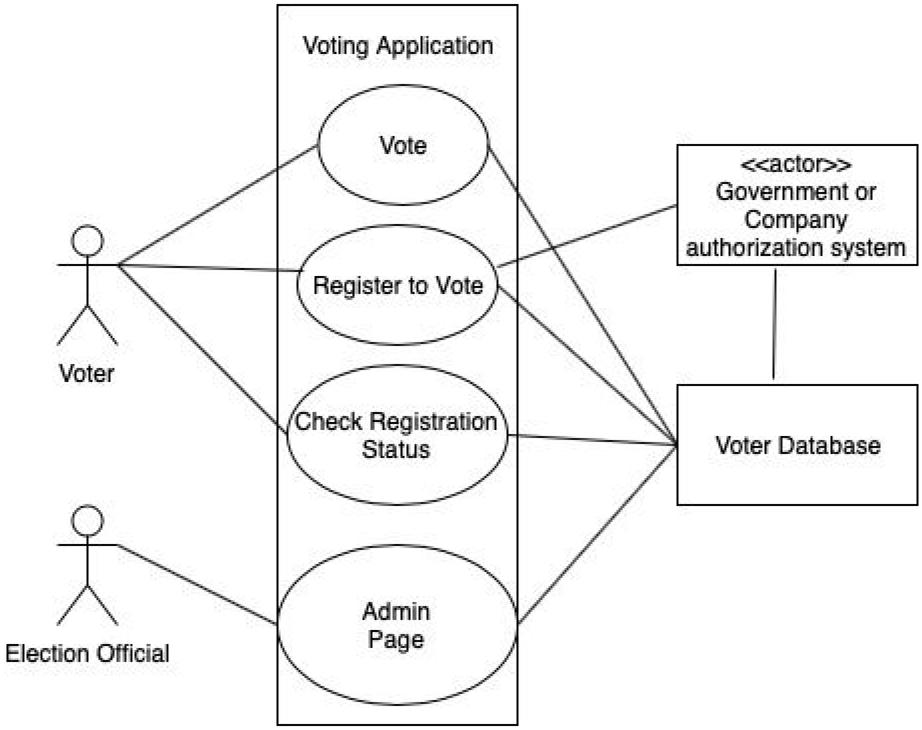
- Election official override by keyboard code

Frequency of Occurrence: Could be nearly continuous.

Open Issues:

- When is voting open?
- Incorrect information?

Use Cases UML diagram:



Our Use Case UML diagram shows the functionality of our application and what each type of user can do.

Supplementary Specifications:

Our supplementary specification outline our functional and nonfunctional requirement.

Functionality: The user should be able to cast their vote as mentioned in the use cases.
Logging and Error Handling: The software will handle all errors and will require an election official in some cases to resume.
Pluggable Rules: The system will be customizable to do certain features bases on the use cases.
Security: The user will have to identify themselves in order to vote, identification methods to be decided on later. There will be no chance to over-vote and voter information as well as choices will be kept secure.
Usability: The screen will have font that is easy to read for user. The interface will be easy to use and understand. There will be no colors associated with color blindness. User will be asked to confirm choices many times, and have the option to change vote.
Reliability: There should be a way for the user to start over or continue if there is a problem with the software and a reset is required.
Performance: The user should not have to wait longer than 1 min for their vote to be processed and counted in the database.

Supportability: The system could be configured for any type of voting needs and with any number of categories or canadetes.
Implementation Constraints: The software is implemented using Java and JavaFX.
Purchased Components: This would be the computer the application is run on and possibly the database being used.
Free Open Source Components: Will be using the free Java software, Java 8 and JavaFX along with Scene Builder to build the software application.
Interfaces: Computer with keyboard and mouse.
Application-Specific Domain Rules:
-User must register to vote
-User cannot vote on the same day that you register
-User can only vote during specified voting periods
Legal Issues: All voting rules in the state must be followed.
Information in Domains of Interest: Each state has their own rules for voter registration. Each users has to have identifiable information in order to register to vote.

Glossary:

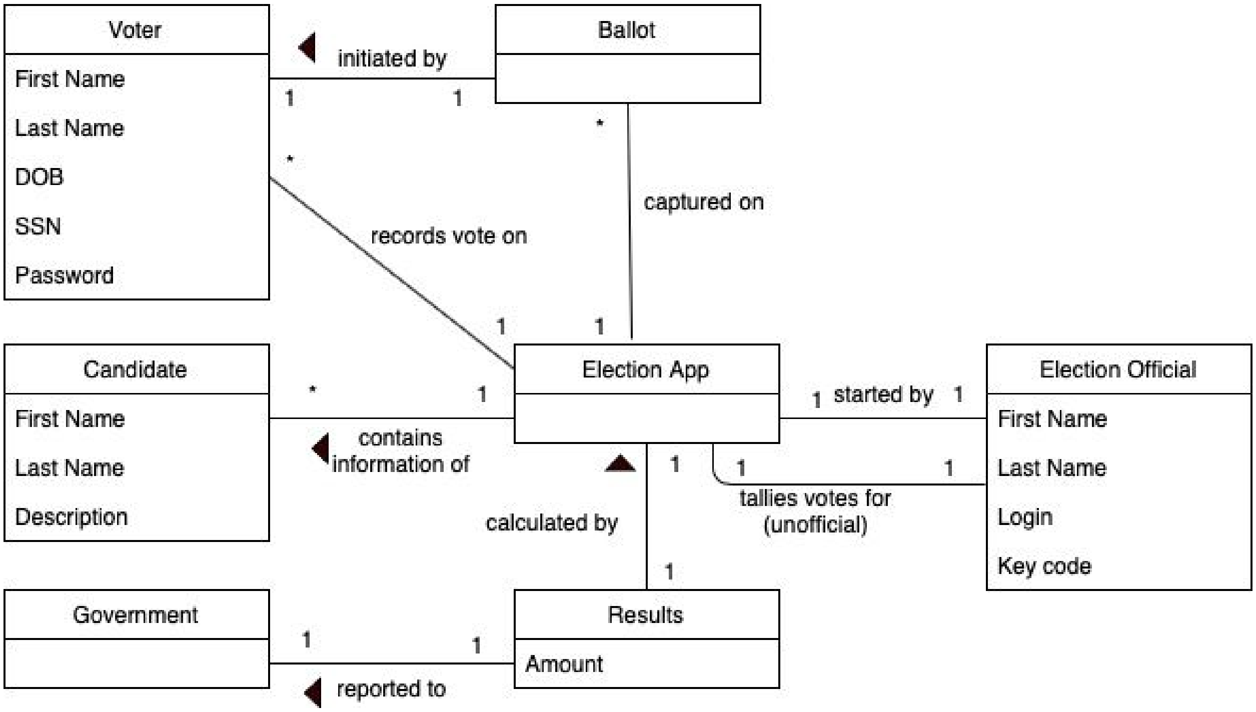
Ballot - the piece of paper used to record someone's vote.
Candidate - a person who applies for a job or is nominated for election.
Category - The classification of the candidates, tells the user who or what they are voting for.
Election - a formal and organized choice by vote of a person for a political office or other position.
Election Official - the person that will be overseeing the voting procedures.
Policy - a course or principle of action adopted or proposed by a government, party, business, or individual
Vote - a formal indication of a choice between two or more candidates or courses of action, expressed typically through a ballot or a show of hands or by voice.
Voter - a person who votes or has the right to vote at an election

Risk List:

Time -There could be a latency between submitting the vote and processing it in the database. This could cause user wait times.
Hardware - There needs to be a decent amount of computers available so the wait time between users voting is not unacceptable.
Labor - The application should be easy for the election official to set up and should be able to override the system if it freezes or can't connect to database.
Conversions - The users information, especially their social security numbers, should be kept in the database in plain text. There should be some amount of effective hashing done before information is stored.

Legal - The users information and voting choices should be kept private. There would be a legal issue if other people could access the database.

Domain Model:

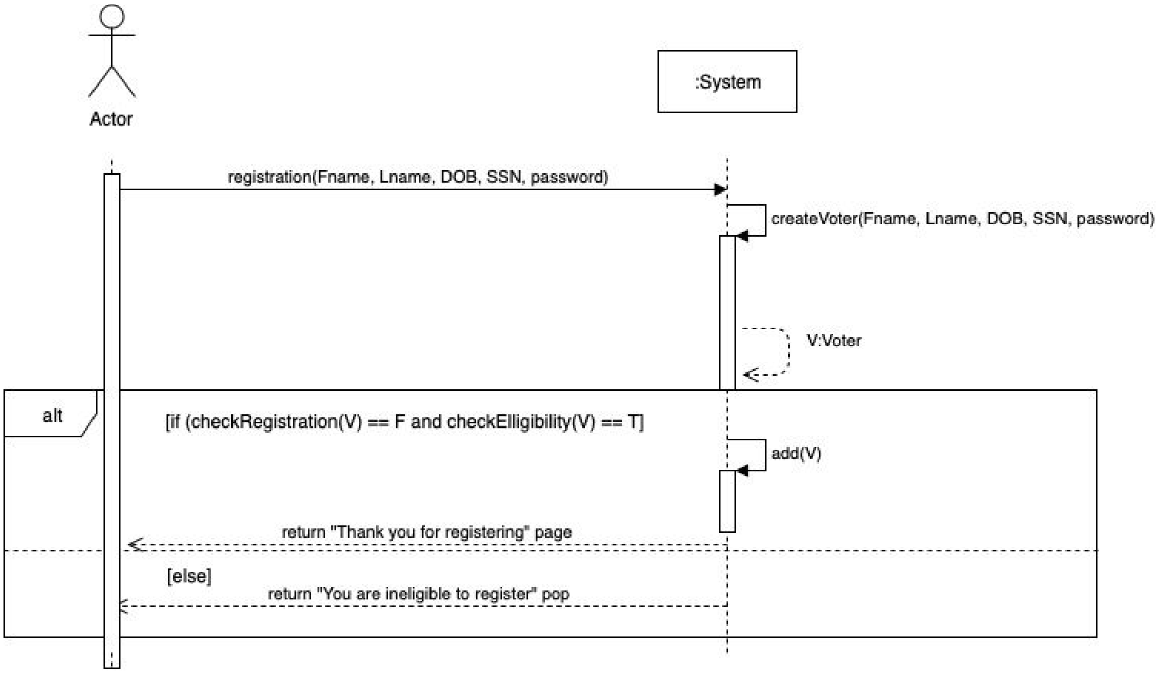


For our domain model, we utilized our use cases to determine the different components needed. Some of the components came easier because they were actors in our use cases, the voter, election app and election official. The other components came from looking at examples are running through our main success scenarios from our use cases. Thinking logically we knew the voter would cast their ballot and the ballot would go to the election app, this is how we made the relationship lines between each of these. The candidate, government and results were less obvious components, each of these might not be necessary but they provide more information about the flow of information. The computer component is represented because our application will require a mouse and keyboard. The results component represents the final count of votes and the flow of them to the government or company after they are calculated by our application.

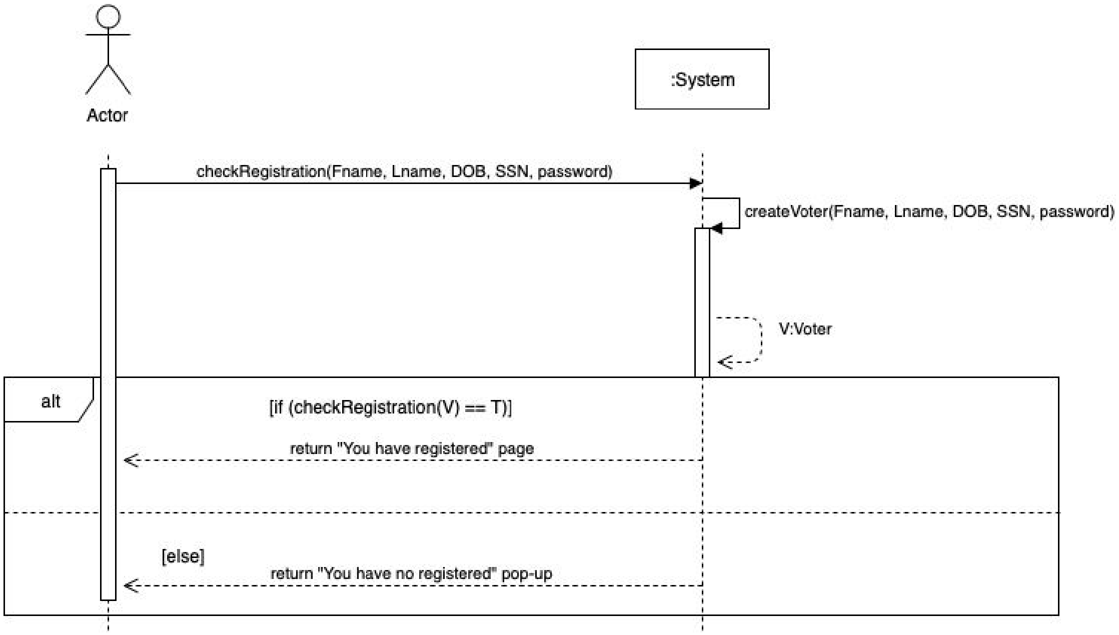
Our System Sequence Diagrams, shown below, outlines the flow of our application, and potential methods used.

System Sequence Diagrams:

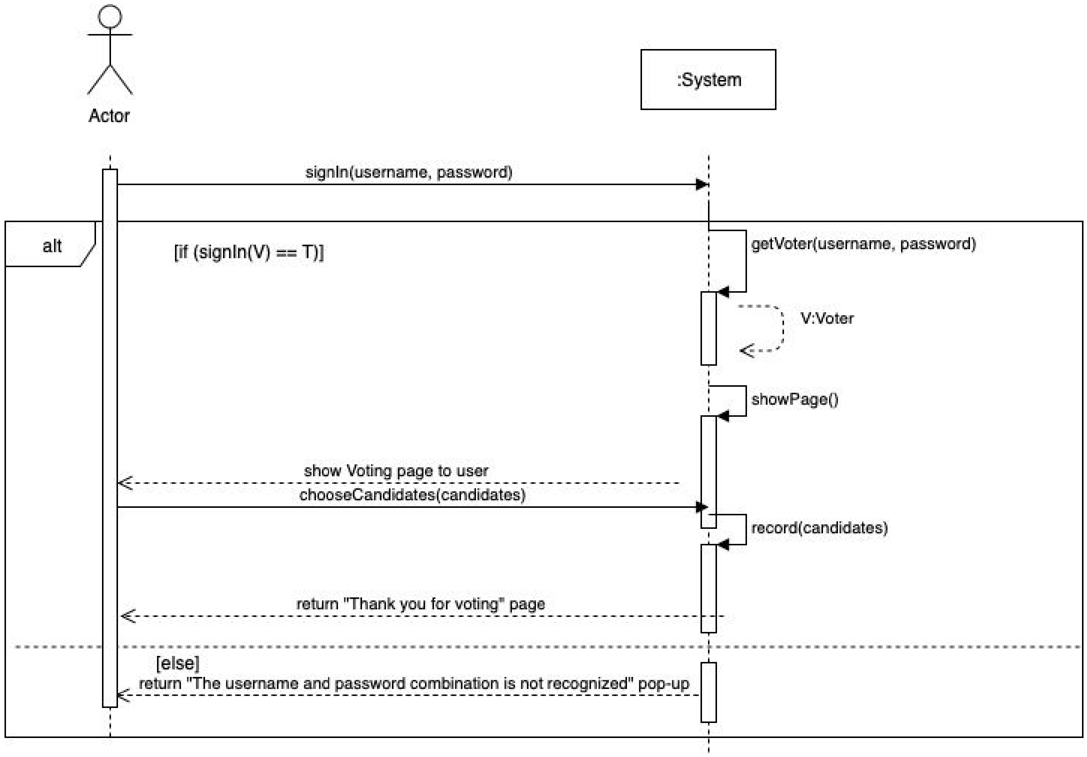
1) Voter Registration Page


Actor
:System
registration(Fname, Lname, DOB, SSN, password)
createVoter(Fname, Lname, DOB, SSN, password)
V:Voter
alt
[if (checkRegistration(V) == F and checkElligibility(V) == T]
add(V)
return "Thank you for registering" page
[else]
return "You are ineligible to register" pop

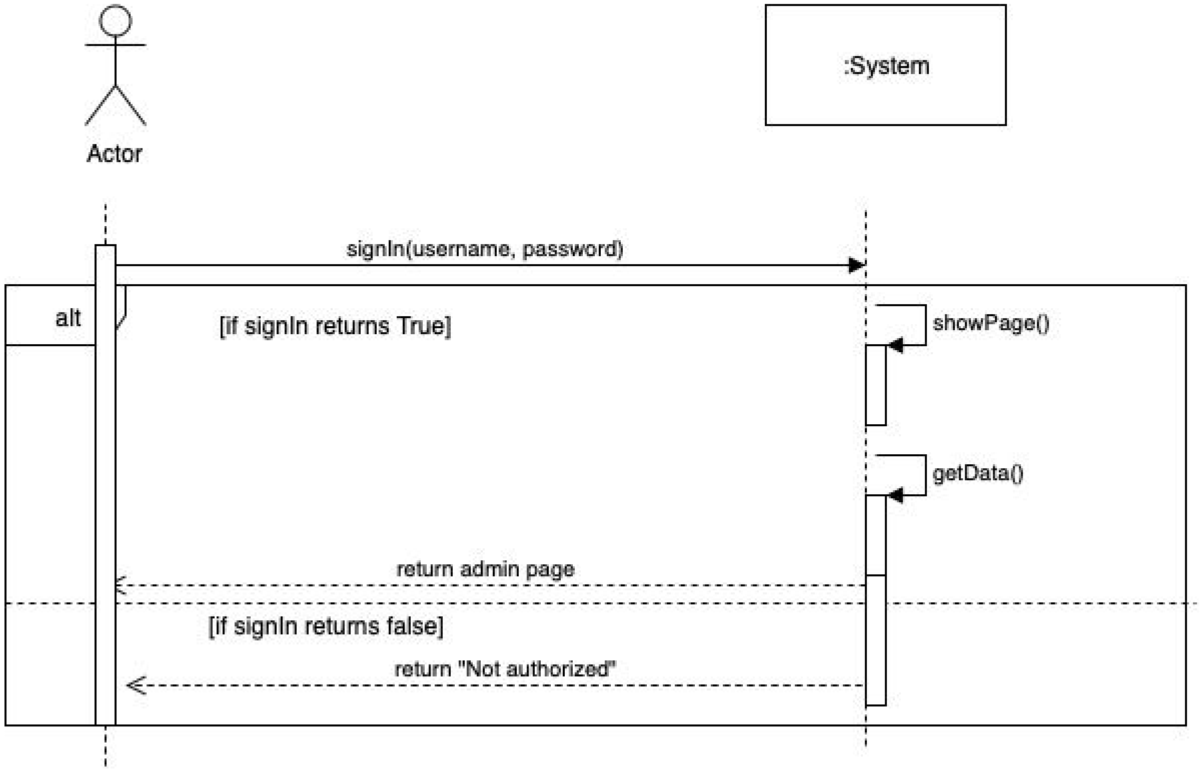
2) Check Registration Status Page


Actor
:System
checkRegistration(Fname, Lname, DOB, SSN, password)
createVoter(Fname, Lname, DOB, SSN, password)
V:Voter
alt
[if (checkRegistration(V) == T)]
return "You have registered" page
[else]
return "You have no registered" pop-up

3) Vote Page


Actor
:System
signIn(username, password)
alt
[if (signIn(V) == T)]
getVoter(username, password)
V:Voter
showPage()
show Voting page to user
chooseCandidates(candidates)
record(candidates)
return "Thank you for voting" page
[else]
return "The username and password combination is not recognized" pop-up

4) Election Official Page



Operation Contacts:

Contract 1: registration

Operation: registration(firstName, lastName, DOB, SSN, PSSW)

Cross Reference: Use Case -User Registers, SD - Voter Registration Page

Pre-Conditions: User clicks Register button on main page, types in their information and clicks submit. All text fields must be filled in to continue.

Post-Conditions:

- Initializes variables using text fields for: firstName, lastName, DOB, SSN, PSSW
- Passes variables to newVoter

Contract 2: newVoter

Operation: newVoter(firstName, lastName, DOB, SSN, PSSW)

Cross Reference: Use Case - User Registers, SD - Voter Registration Page

Pre-Conditions: Registration method is called and correct parameters are passed in.

Post-Conditions:

- Voter object is created with the fields specified

Contract 3: checkRegistration

Operation: checkRegistration(voter V)

Cross Reference: Use Case - Check Registration Status, SD - Voter Registration Page

Pre-Conditions: Voter object 'V' has been created.

Post-Conditions:

- Initializes "registered" variable as True which then creates a window telling the voter they are already registered
- Initializes "registered" variable as False which then creates a window telling the voter they are not registered

Contract 4: checkEligibility

Operation: checkEligibility(voter V)

Cross Reference: Use Case - User Registers, SD - Voter Registration Page

Pre-Conditions: A voter object has been created and checkRegistration method returned false.

Post-Conditions:

- Initializes "eligible" variable as True if user meets the criteria to vote
- Initializes "eligible" variable as False if use does not meet the criteria to vote

Contract 5: add

Operation: add(voter V)

Cross Reference: Use Case - User Registers, SD - Voter Registration Page

Pre-Conditions: The method checkRegistration returns false while checkEligibility returns true.

Post-Conditions:

- Voter object is added to the database

Contract 6: signIn

Operation: signIn(username, password)

Cross Reference: Use Case - User Votes, SD - Voting Page

Pre-Conditions: User chooses either the Election Official page or the voting page.

Post-Conditions:

- System checks database for information matching current user
- Initializes “registeredVoter” variable as True if the user is registered to vote, allowing them to continue to the voting process
- Initializes “registeredVoter” variable as False if the user is not registered, preventing them from accessing the voting page

Contract 7: getVoter

Operation: getVoter(username, password)

Cross Reference: Use Case - User Votes, SD - Voting Page

Pre-Conditions: The method signIn has been called and passed ‘True’ that the voter exists. The voter must not have already voted.

Post-Conditions:

- Calls the database to get the voters information
- Object of the correct voter is retrieved

Contract 8: record

Operation: record(V, candidates)

Cross Reference: Use Case - User Votes, SD - Voting Page

Pre-Conditions: The user chooses the candidates they want to vote for.

Post-Conditions:

- User object is given attribute “voted” initialized to True showing that the user has voted already, preventing further ballot submissions

Contract 9: showPage

Operations: showPage()

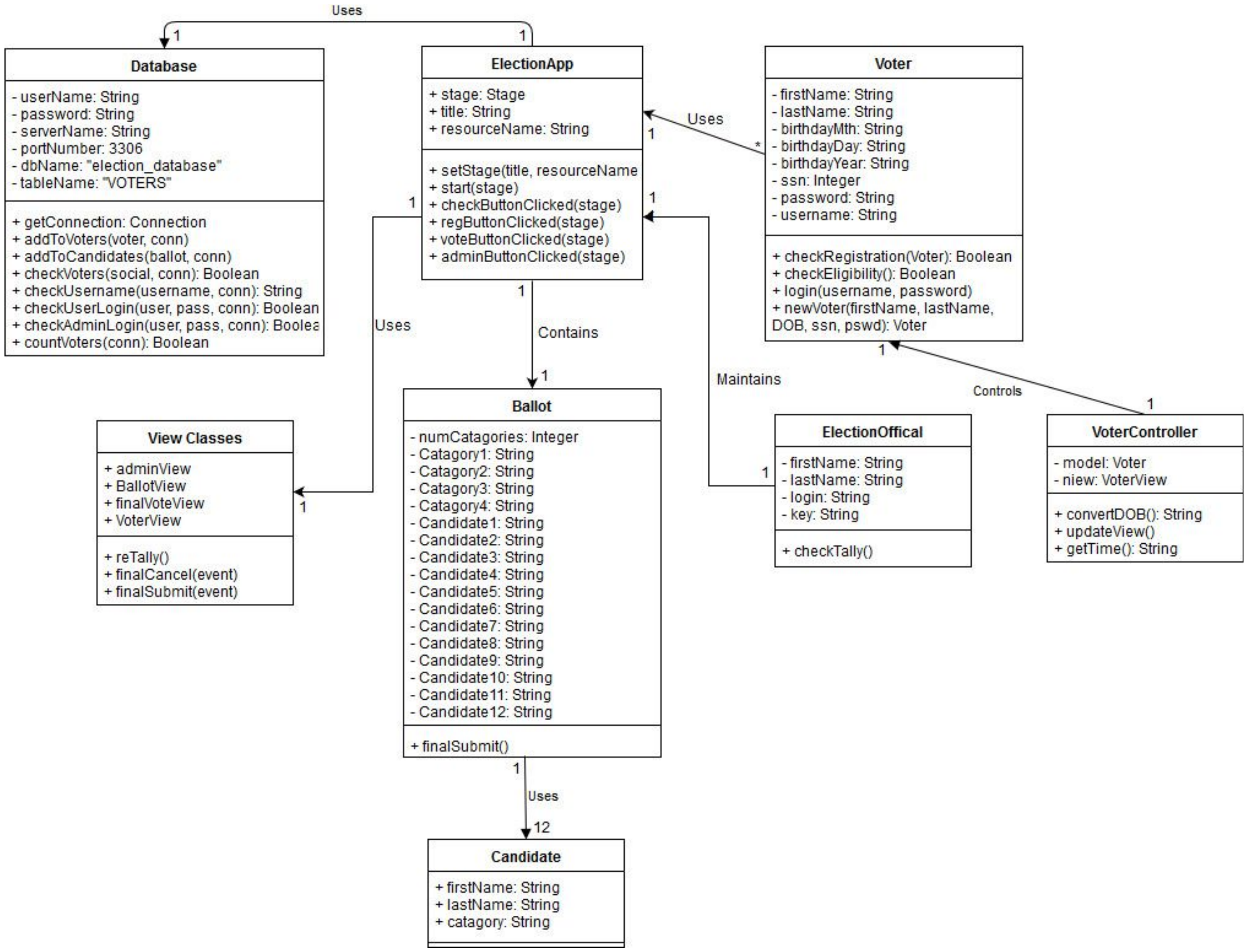
Cross Reference: Use Case - User Votes, SD - Voting Page, Election Official Page

Pre-Conditions: User is logged in.

Post-Conditions:

- Voting page is shown to the user or current tally page is shown to the election official

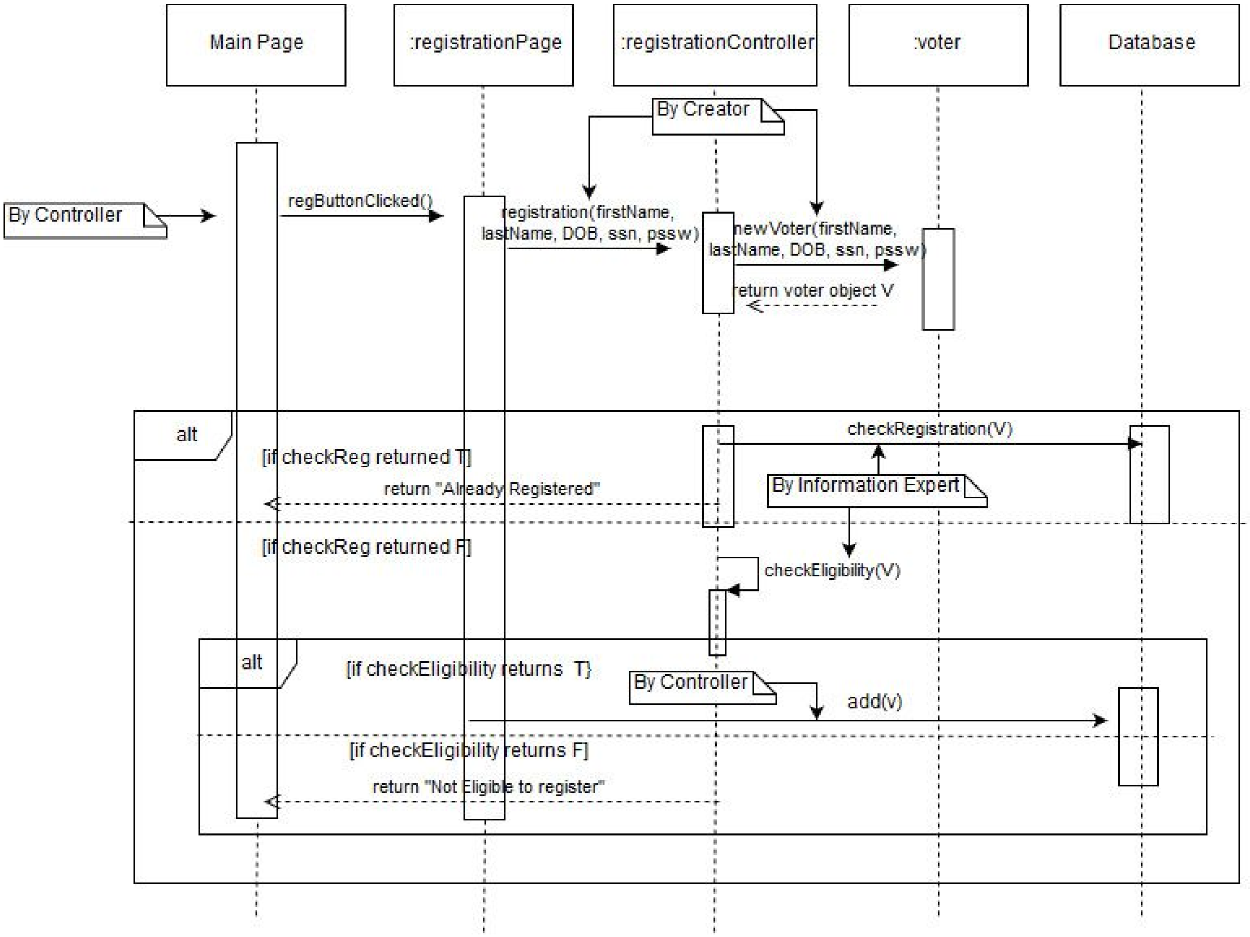
Class Diagram:



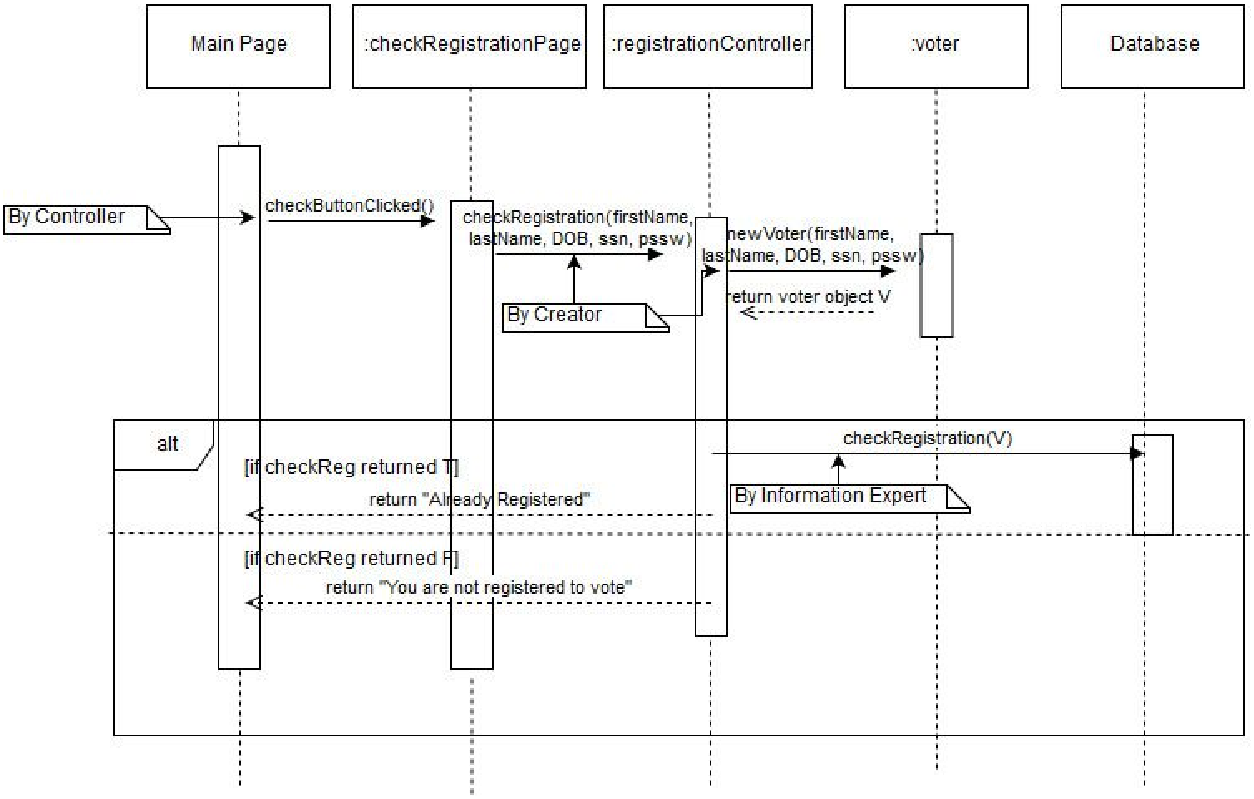
Sequence Diagrams:

The following sequence diagrams show how our use cases will be implemented in our application and potential methods being used.

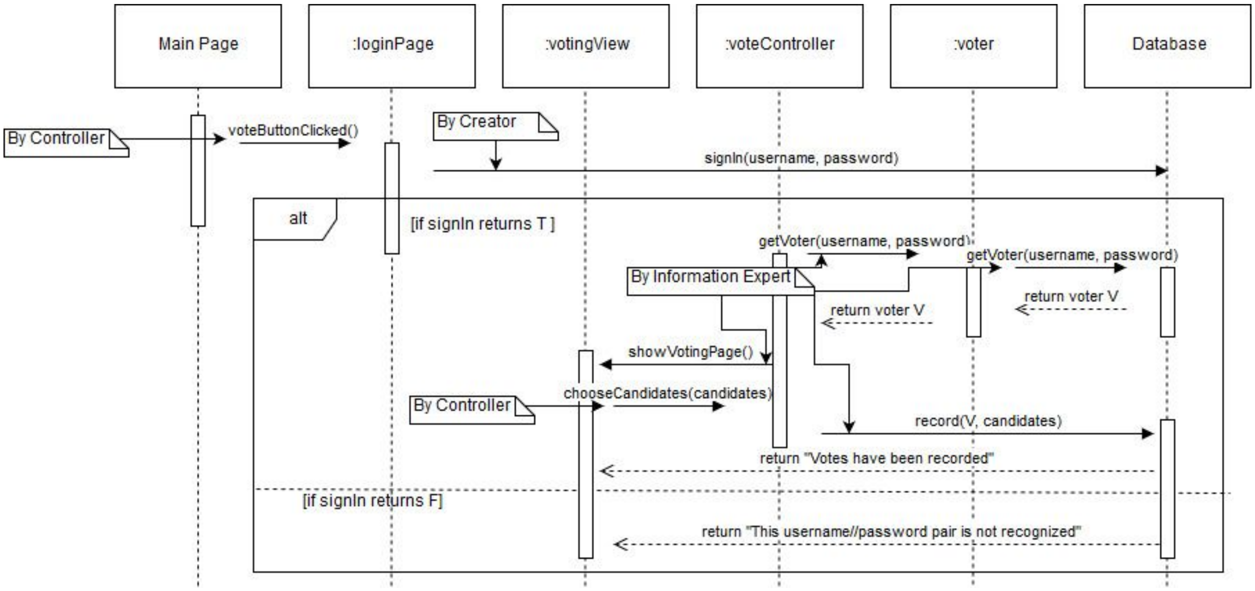
1) **Voter registration**


Main Page
:registrationPage
:registrationController
:voter
Database
By Creator
By Controller
regButtonClicked()
registration(firstName, lastName, DOB, ssn, pssw)
newVoter(firstName, lastName, DOB, ssn, pssw)
return voter object V
alt
checkRegistration(V)
[if checkReg returned T]
return "Already Registered"
By Information Expert
[if checkReg returned F]
checkEligibility(V)
alt
[if checkEligibility returns T}
By Controller
add(v)
[if checkEligibility returns F]
return "Not Eligible to register"

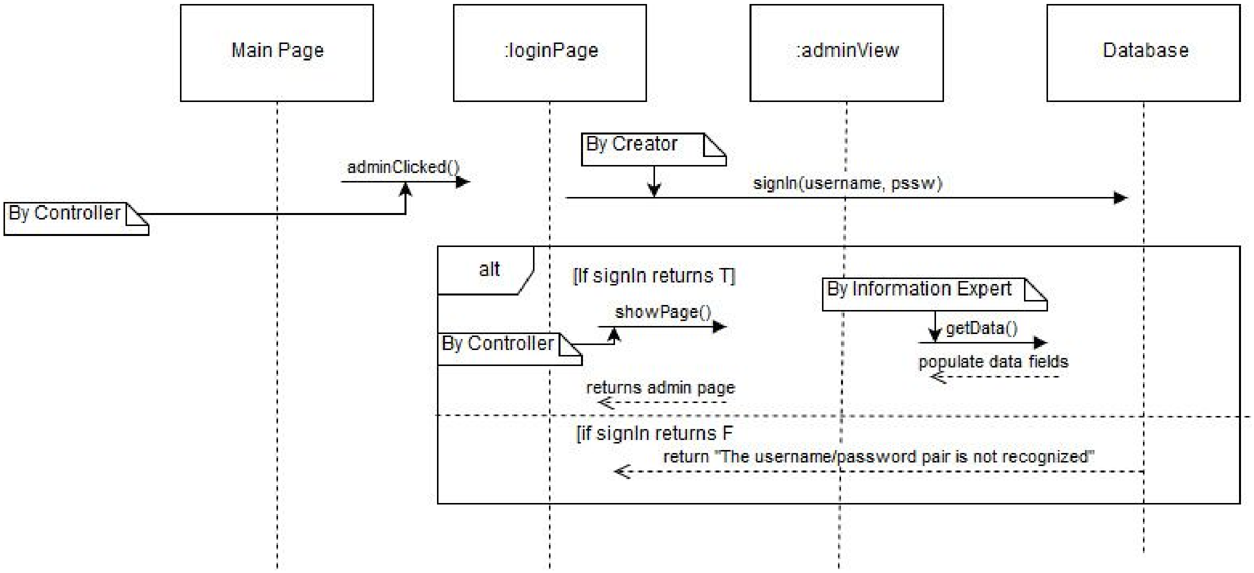
2) Check Registration Status



3) Voting Page



4) Election Official Page

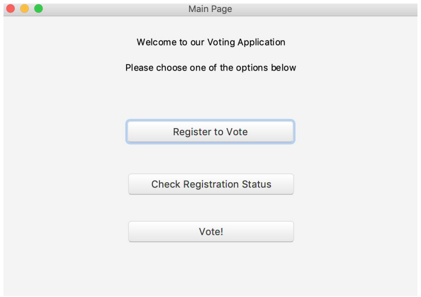


GUI Versions:

The following are screenshots from past and current versions of our application to show the GUI.

GUI Prototype Version 1:

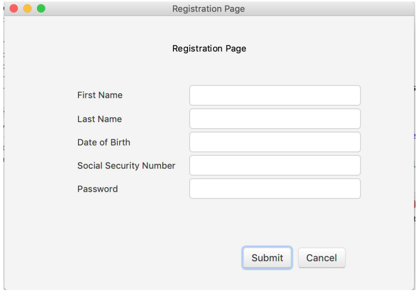
Main Page:


Main Page

Welcome to our Voting Application

Please choose one of the options below

Voter Registration Page:


Registration Page

Registration Page

First Name

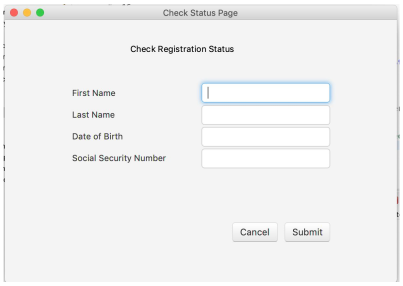
Last Name

Date of Birth

Social Security Number

Password

Check Registration Status Page:


Check Status Page

Check Registration Status

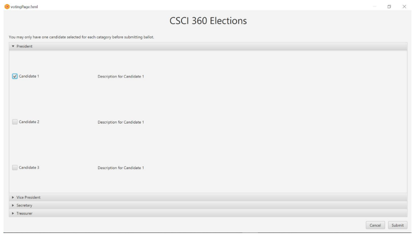
First Name

Last Name

Date of Birth

Social Security Number

Voting Page:


votingPage.fxml

CSCI 360 Elections

You may only have one candidate selected for each catagory before submitting ballot.

▼ President

☑ Candidate 1 Description for Candidate 1

□ Candidate 2 Description for Candidate 1

□ Candidate 3 Description for Candidate 1

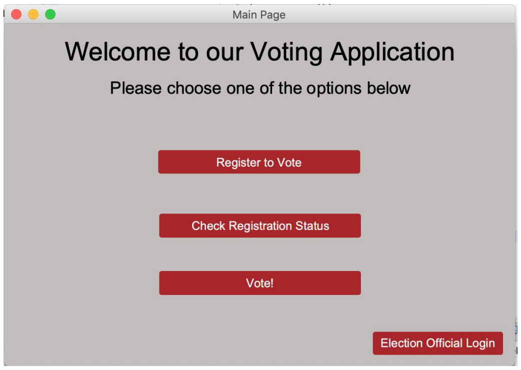
▶ Vice President

▶ Secretary

▶ Treasurer

Version 2:

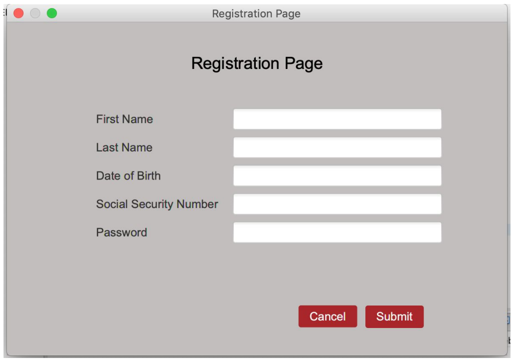
Main Page:



Welcome to our Voting Application

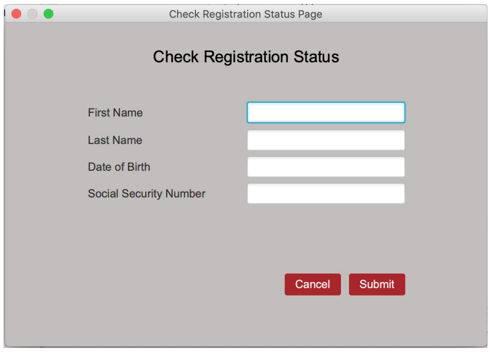
Please choose one of the options below

Registration Page:



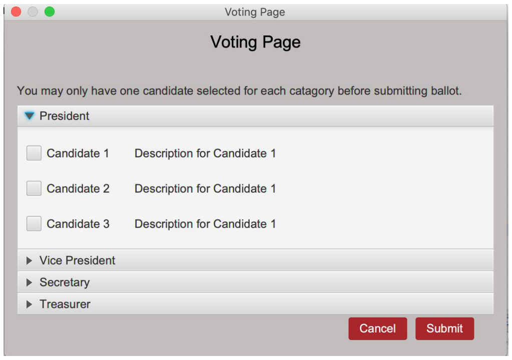
Registration Page

Check Registration Status Page:



Check Registration Status

Voting Page:



Voting Page

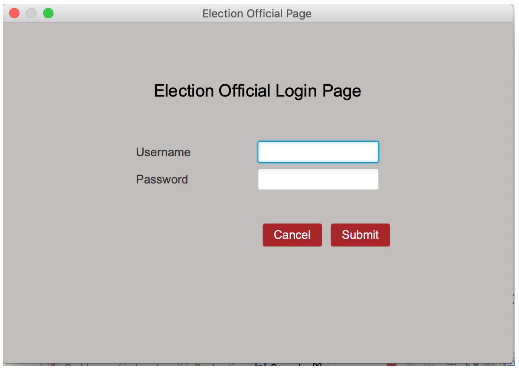
You may only have one candidate selected for each catagory before submitting ballot.

□ Candidate 1 Description for Candidate 1

□ Candidate 2 Description for Candidate 1

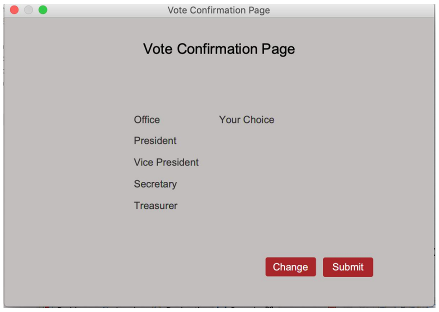
□ Candidate 3 Description for Candidate 1

Election Official Login page:



Election Official Login Page

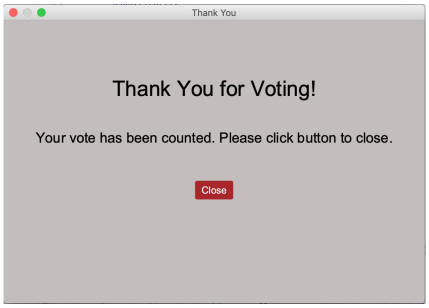
Voting Confirmation Page (displays candidates the voter chose):



Vote Confirmation Page

Office	Your Choice
President	
Vice President	
Secretary	
Treasurer	

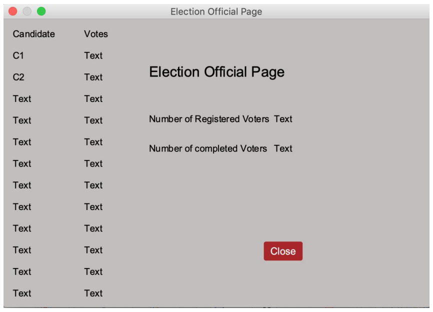
FInal Page, click to close application:



Thank You for Voting!

Your vote has been counted. Please click button to close.

Shows unofficial tally of votes and other useful information:



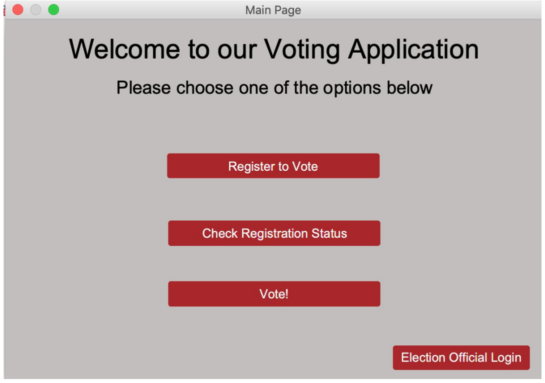
Candidate	Votes
C1	Text
C2	Text
Text	Text
Text	Text
Text	Text
Text	Text
Text	Text
Text	Text
Text	Text
Text	Text
Text	Text
Text	Text

Election Official Page

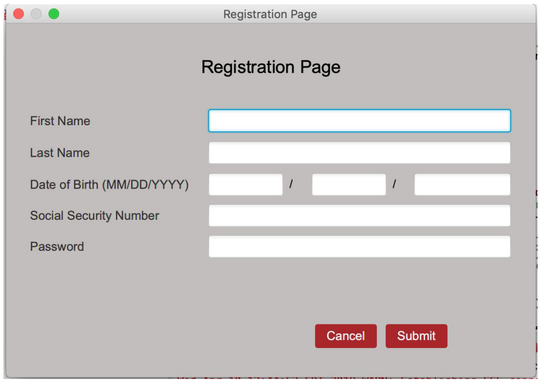
Number of Registered Voters Text

Number of completed Voters Text

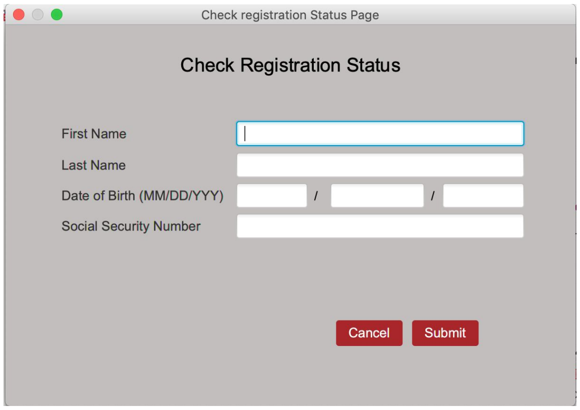
Version 3:


Main Page
Welcome to our Voting Application
Please choose one of the options below
Register to Vote
Check Registration Status
Vote!
Election Official Login

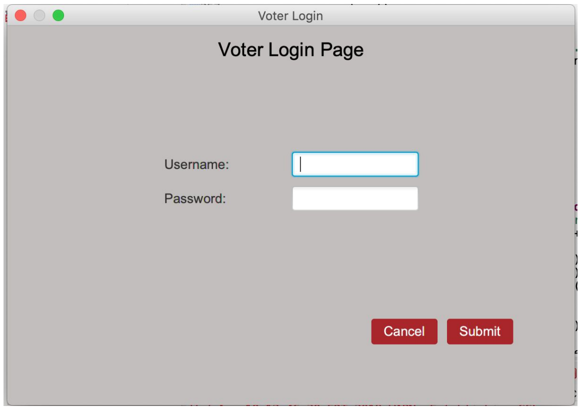
Main Page



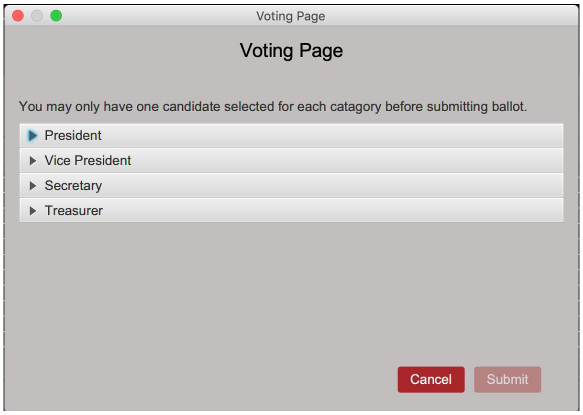
Registration Page


Check registration Status Page
Check Registration Status
First Name
Last Name
Date of Birth (MM/DD/YYY)
/
/
Social Security Number
Cancel
Submit

Check Registration Status Page

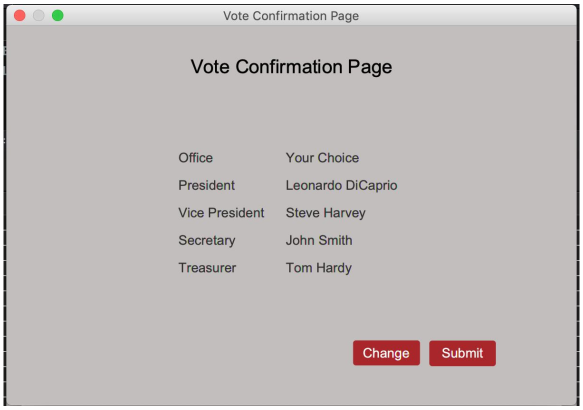


Voter Login Page



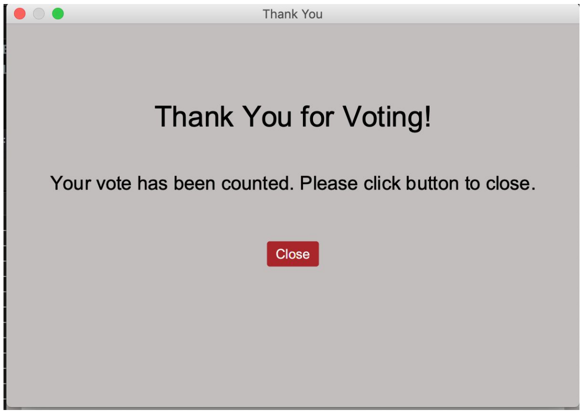
You may only have one candidate selected for each catagory before submitting ballot.

Voting Page



Office	Your Choice
President	Leonardo DiCaprio
Vice President	Steve Harvey
Secretary	John Smith
Treasurer	Tom Hardy

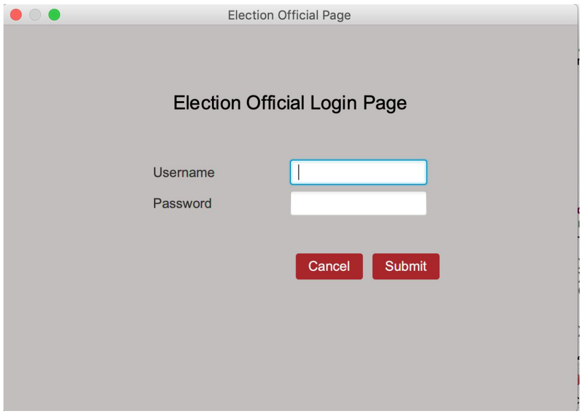
Votes confirmation page



Thank You for Voting!

Your vote has been counted. Please click button to close.

Thank you page that closes application

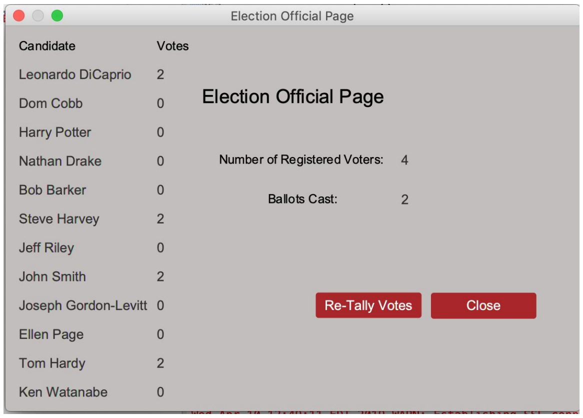


Election Official Login Page

Username

Password

Admin login



Candidate	Votes
Leonardo DiCaprio	2
Dom Cobb	0
Harry Potter	0
Nathan Drake	0
Bob Barker	0
Steve Harvey	2
Jeff Riley	0
John Smith	2
Joseph Gordon-Levitt	0
Ellen Page	0
Tom Hardy	2
Ken Watanabe	0

Election Official Page

Number of Registered Voters: 4

Ballots Cast: 2

Admin information page