Example Scenarios

|  |  |
| --- | --- |
| Scenario Name: | creatingElection |
| Actors: | Steve: Elections Officer |
| Flow of Events: | 1. Steve is an elections officer who needs to create an election for a seat in local government so that voters can vote on it. 2. Steve decides to use the evote system and opens the evote system on his desktop computer and enters his username and password for his previously made account before clicking login. 3. Steve navigates to the tools section on the homepage and hits create an election. 4. Steve then creates a political party for Republicans and another one for Democrats. 5. Steve then creates two election candidates and denotes one as a Republican and one as a Democrat. 6. Steve then creates an additional alternative candidate. 7. Steve then creates the electoral district for the election. 8. Steve then opens the election up for voting in the specified electoral district. 9. Voters vote in the election over a specified amount of time. 10. At any time Steve can access the system to view, delete and modify the election, electoral district, political parties or candidates. 11. Steve closes the election for voting once the time has expired and validates the election results. 12. Steve publishes the election results so that the public can view it and see who the winner is. 13. Steve can choose to create more elections or issues as needed. |

|  |  |
| --- | --- |
| Scenario Name: | creatingIssue |
| Actors: | Steve: Elections Officer |
| Flow of Events: | 1. Steve is an elections officer who needs to create an issue for an upcoming issue so that voters can vote on it and he decides to use the evote system. 2. Steve opens the evote system on his desktop computer and enters his username and password for his previously made account and then clicks login. 3. Steve navigates to the tools section on the homepage and clicks creates an issue. 4. Steve then creates the electoral district for the specified issue. 5. Steve then creates the issue and then writes a brief description of the issue and specifies that the voter can choose yes or no. 6. Steve then opens the system up for voting in the specified electoral district. 7. Voters vote in the election over a specified amount of time. 8. At any time Steve can access the system to view, delete and modify the issue or electoral district. 9. Steve closes the system for voting once the time has expired. 10. The issue is then either given the vote of yes or no depending on how the voters voted. 11. Steve can choose to create more elections or issues as needed. |

Use Cases

|  |  |
| --- | --- |
| Name | LogIn |
| ID | LOGIN-1 |
| Version | 1 |
| Author | Donovan Todd |
| Date | 9/6/2016 |
| Summary | This use case details the login steps of an Elections Officer; it is necessary to use the functions of the system. |
| Basic Path | 1. The system displays a prompt for the username and password. The login button is also displayed by the system. 2. The ElectionsOfficer enters the username and password and then his the login button. 3. The system locates the ElectionsOfficer’s user object in the database. The system verifies the ElectionsOfficer’s credentials and access rights. Then the system displays the overview of the ElectionsOfficer’s account. |
| Alternative Paths | N/A |
| Exception Paths | If in step 3 the ElectionsOfficer’s user object cannot be located in the database, the system displays the error message SystemUnavailable and terminates the use case.  If in step 3 the system cannot verify ElectionOfficer’s access rights, the system displays the error message UnknownUser and terminates the use case. |
| Extension Points | N/A |
| Triggers | The ElectionsOfficer decides to use the evote system. |
| Assumption | N/A |
| Pre-conditions | N/A |
| Post-conditions | The ElectionsOfficer’s user object is available to other use cases available to the same ElectionsOfficer. |

|  |  |
| --- | --- |
| Name | LogOut |
| ID | LOGOUT-1 |
| Version | 1 |
| Author | Donovan Todd |
| Date | 9/6/2016 |
| Summary | This use case details the logout steps of an Elections Officer; it is necessary to exit the system. |
| Basic Path | 1. ElectionsOfficer selects the Logout button. 2. The system logs the ElectionOfficer out and displays the start page. |
| Alternative Paths | N/A |
| Exception Paths | N/A |
| Extension Points | N/A |
| Triggers | The ElectionsOfficer has elected to logout. |
| Assumption | N/A |
| Pre-conditions | The ElectionsOfficer is currently logged in and the session has not expired. The ElectionsOfficer’s user object is available. |
| Post-conditions | The user is no longer logged in. The application displays a prompt for the username and password. The login button is also displayed by the application. |

|  |  |
| --- | --- |
| Name | Create a new Election |
| ID | CreateElection |
| Version | 1 |
| Author | Donovan Todd |
| Date | 9/6/2016 |
| Summary | Use case to create a new Election for an Elections Officer. |
| Basic Path | 1. The system displays the prompt to provide the following attributes of the new Election: name; Done and Cancel buttons are visible. 2. The ElectionsOfficer provides the required attribute values and clicks Done. 3. The system creates a new Election with the provided data and the current ElectionsOfficer is set as the owner of the new Election. 4. The system displays a screen showing all of the Election attributes provided so far. The screen displays three buttons: CreatePoliticalParty, Done, and Cancel, as well. 5. The ElectionsOfficer clicks Done. 6. The system displays a screen showing all of the Election attributes provided so far, as well as PoliticalParties provided so far. The screen displays three buttons: CreateCandidates, Done, and Cancel, as well. 7. The ElectionsOfficer clicks Done. 8. The system displays a screen showing all of the Election attributes provided so far as well as PolitcalParties and Candidates provided so far. The screen displays three buttons: CreateElectoralDistrict, Done, and Cancel, as well. 9. The ElectionsOfficer clicks Done. 10. The system displays a screen showing all of the Election attributes provided so far as well as PoliticalParties, Candidates, and ElectoralDistricts provided so far. The screen displays two buttons: Done and Cancel, as well. 11. The ElectionsOfficer clicks Done. |
| Alternative Paths | 1. In steps 2, 5, 7, 9, 11 if the user presses the Cancel button, the use case is terminated and no changes to the system are made. |
| Exception Paths | 1. In step 3, if the system determines that an Election with the user inputted name already exists, the system displays the error message DuplicateElectionName and the system goes back to step 1 in the Basic Path, retaining all previously provided attribute values. |
| Extension Points | 1. In step 4, if the user selects CreatePoliticalParty, the system executes the CreatePolitcalParty use case extension; at its completion, the system displays a screen showing all of the Election attributes and PoliticalParties provided so far and goes to step 5 in the Basic Path. 2. In step 6, if the user selects CreateCandidates, the system executes the CreateCandidates use case extension; at its completion, the system displays a screen showing all of the Election attributes, as well as PoliticalParties and Candidates provided so far and goes to step 7 in the Basic Path. 3. In step 6, if the user selects CreateElectoralDistrict, the system executes the CreateElectoralDistrict use case extension; at its completion, the system displays a screen showing all of the Election attributes, as well as PoliticalParties, Candidates, and ElectoralDistricts provided so far and goes to step 7 in the Basic Path. |
| Triggers | The ElectionsOfficer has elected to create a new Election. |
| Assumption | N/A |
| Pre-conditions | The ElectionsOfficer is currently logged in and the session has not expired. The ElectionsOfficer’s user object is available. |
| Post-conditions | The new Election with the defined attributes is available for use. Its owner is the current ElectionsOfficer and its PoliticalParties, Candidates, and ElectoralDistricts are set as specified. |

|  |  |
| --- | --- |
| Name | Create a new Issue |
| ID | CreateIssue |
| Version | 1 |
| Author | Donovan Todd |
| Date | 9/6/2016 |
| Summary | Use case to create a new Issue for an Elections Officer. |
| Basic Path | 1. The system displays the prompt to provide the following attributes of the new Issue: name and description; Done and Cancel buttons are visible. 2. The ElectionsOfficer provides the required attribute values and clicks Done. 3. The system creates a new Issue with the provided data and the current ElectionsOfficer is set as the owner of the new Issue. 4. The system displays a screen showing all of the Issue attributes provided so far. The screen displays three buttons: CreateElectoralDistrict, Done, and Cancel, as well. 5. The ElectionsOfficer clicks Done. 6. The system displays a screen showing all of the Issue attributes provided so far as well as ElectoralDistricts provided so far. The screen displays two buttons: Done and Cancel, as well. 7. The ElectionsOfficer clicks Done. |
| Alternative Paths | 1. In steps 2, 5, 7, if the user presses the Cancel button, the use case is terminated and no changes to the system are made. |
| Exception Paths | 1. In step 3, if the system determines that an Issue with the user inputted name already exists, the system displays the error message DuplicateIssueName and the system goes back to step 1 in the Basic Path, retaining all previously provided attribute values. |
| Extension Points | 1. In step 4, if the ElectionsOfficer selects CreateElectoralDistrict, the system executes the CreateElectoralDistrict use case extension; at its completion, the system displays a screen showing all of the Issue attributes and ElectoralDistricts provided so far and goes to step 5 in the Basic Path. |
| Triggers | The ElectionsOfficer has elected to create a new Issue. |
| Assumption | N/A |
| Pre-conditions | The ElectionsOfficer is currently logged in and the session has not expired. The ElectionsOfficer’s user object is available. |
| Post-conditions | The new Issue with the defined attributes is available for use. Its owner is the current ElectionsOfficer and its ElectoralDistricts are set as specified. |

|  |  |
| --- | --- |
| Name | Create a new Electoral District |
| ID | CreateElectoralDistrict |
| Version | 1 |
| Author | Donovan Todd |
| Date | 9/6/2016 |
| Summary | An extension use case to create a new Electoral District to an Issue or Election; used by CreateIssue and CreateElection use cases |
| Basic Path | 1. The system displays the prompt to enter: name, area; FindElectoralDistrict, Done and Cancel buttons are visible. 2. The ElectionsOfficer provides the required information and clicks Done. 3. The system creates a new ElectoralDistrict with the provided data or uses an existing one, if one was selected during the FindElectoralDistrict use case execution. The ElectoralDistrict is linked to the Issue or Election being created or modified in the referring use case. The system displays a new screen in which all the created ElectoralDistricts are shown. The screen displays 3 buttons: CreateElectoralDistrict, Done and Cancel as well. |
| Alternative Paths | 1. In step 2 and 3, if the user presses the Cancel button, the use case is terminated and no changes to the system are made. 2. In step 2, if the user selects the FindElectoralDistrict button, the system executes the FindElectoralDistrict use case; at its resolution, the system returns to step 2 in the Basic Path, and if the ElectionsOfficer selected one of the already created ElectoralDitricts during the execution of the FindElectoralDistrict use case, all of the attributes of the selected ElectoralDistrict are preset as if entered by the user; otherwise no attributes are set in advance. 3. In step 3, if the ElectionsOfficer selects the CreateElectoralDistrict button, the system goes back to step 1 in the Basic Path. |
| Exception Paths | 1. In step 3, if the system determines that an ElectoralDistrict with the user inputted name already exists, the system displays the error message DuplicateElectoralDistrictName and the system goes back to step 1 in the Basic Path, retaining all previously provided attribute values. |
| Extension Points | 1. In step 3, if the ElectionsOfficer selects FindElectoralDistrict, the system executes the FindElectoralDistrict use case extension; at its completion, the system displays a screen showing all of the ElectoralDistricts provided so far and goes to step 2 in the Basic Path. |
| Triggers | The ElectionsOfficer has elected to create a new Electoral District within the CreateIssue or CreateElection use cases. |
| Assumption | N/A |
| Pre-conditions | The ElectionsOfficer is currently logged in and the session has not expired. The ElectionsOfficer’s user object is available. The Election or Issue from the extended use case is available. |
| Post-conditions | A new ElectoralDistrict is added to the Issue or Election. |

|  |  |
| --- | --- |
| Name | Create a new Political Party |
| ID | CreatePoliticalParty |
| Version | 1 |
| Author | Donovan Todd |
| Date | 9/6/2016 |
| Summary | An extension use case to create a new Political Party for an Election; used by CreateElection use cases |
| Basic Path | 1. The system displays the prompt to enter: name; FindPoliticalParty, Done and Cancel buttons are visible. 2. The ElectionsOfficer provides the required information and clicks Done. 3. The system creates a new PoliticalParty with the provided data or uses an existing one, if one was selected during the FindPoliticalParty use case execution. The PoliticalParty is linked to the Election being created or modified in the referring use case. The system displays a new screen in which all the created PoliticalParties are shown. The screen displays 3 buttons: CreatePoliticalParty, Done and Cancel as well. |
| Alternative Paths | 1. In step 2 and 3, if the user presses the Cancel button, the use case is terminated and no changes to the system are made. 2. In step 2, if the user selects the FindPoliticalParty button, the system executes the FindPoliticalParty use case; at its completion, the system returns to step 2 in the Basic Path, and if the ElectionsOfficer selected one of the already created PoliticalParty during the execution of the FindPoliticalParty use case, all of the attributes of the selected PoliticalParty are preset as if entered by the user; otherwise no attributes are set in advance. 3. In step 3, if the ElectionsOfficer selects the CreatePoliticalParty button, the system goes back to step 1 in the Basic Path. |
| Exception Paths | 1. In step 3, if the system determines that a PoliticalParty with the user inputted name already exists, the system displays the error message DuplicatePoliticalPartyName and the system goes back to step 1 in the Basic Path, retaining all previously provided attribute values. |
| Extension Points | 1. In step 3, if the ElectionsOfficer selects FindPoliticalParty, the system executes the FindPoliticalParty use case extension; at its completion, the system displays a screen showing all of the PoliticalParties provided so far and goes to step 2 in the Basic Path. |
| Triggers | The ElectionsOfficer has elected to create a new PoliticalParty within the CreateElection use case. |
| Assumption | N/A |
| Pre-conditions | The ElectionsOfficer is currently logged in and the session has not expired. The ElectionsOfficer’s user object is available. The Election from the extended use case is available. |
| Post-conditions | A new PoliticalParty is added to the Election. |

|  |  |
| --- | --- |
| Name | Create a new Candidate |
| ID | CreateCandidate |
| Version | 1 |
| Author | Donovan Todd |
| Date | 9/6/2016 |
| Summary | An extension use case to create a new Candidate for an Election; used by CreateElection use case |
| Basic Path | 1. The system displays the prompt to enter: name and political party; FindCandidate, Done and Cancel buttons are visible. 2. The ElectionsOfficer provides the required information and clicks Done. 3. The system creates a new Candidate with the provided data or uses an existing one, if one was selected during the FindCandidate use case execution. The Candidate is linked to the Election being created or modified in the referring use case. The system displays a new screen in which all the created Candidates are shown. The screen displays 3 buttons: CreateCandidate, Done and Cancel as well. |
| Alternative Paths | 1. In step 2 and 3, if the user presses the Cancel button, the use case is terminated and no changes to the system are made. 2. In step 2, if the user selects the FindCandidate button, the system executes the FindCandidate use case; at its completion, the system returns to step 2 in the Basic Path, and if the ElectionsOfficer selected one of the already created Candidate during the execution of the FindCandidate use case, all of the attributes of the selected Candidate are preset as if entered by the user; otherwise no attributes are preset. 3. In step 3, if the ElectionsOfficer selects the CreateCandidate button, the system goes back to step 1 in the Basic Path. |
| Exception Paths | 1. In step 3, if the system determines that a Candidate with the user inputted name already exists, the system displays the error message DuplicateCandidateName and the system goes back to step 1 in the Basic Path, retaining all previously provided attribute values. |
| Extension Points | 1. In step 3, if the ElectionsOfficer selects FindCandidate, the system executes the FindCandidate use case extension; at its completion, the system displays a screen showing all of the Candidates provided so far and goes to step 2 in the Basic Path. |
| Triggers | The ElectionsOfficer has elected to create a new Candidate within the CreateElection use case. |
| Assumption | N/A |
| Pre-conditions | The ElectionsOfficer is currently logged in and the session has not expired. The ElectionsOfficer’s user object is available. The Election from the extended use case is available. |
| Post-conditions | A new Candidate is added to the Election. |

|  |  |
| --- | --- |
| Name | Find an existing Electoral District |
| ID | FindElectoralDistrict |
| Version | 1 |
| Author | Donovan Todd |
| Date | 9/6/2016 |
| Summary | An extension use case to find an existing Electoral District; used by CreateElectoralDistrict use case |
| Basic Path | 1. The system displays all created ElectoralDistricts. Done and Cancel buttons are also visible. 2. The ElectionsOfficer selects a created ElectoralDistricts and clicks Done. |
| Alternative Paths | 1. In step 1, if the user presses the Cancel button, the use case is terminated and no changes to the system are made. |
| Exception Paths | N/A |
| Extension Points | N/A |
| Triggers | The ElectionsOfficer has elected to find an already created Electoral District within the CreateElectoralDistrict use case. |
| Assumption | N/A |
| Pre-conditions | The ElectionsOfficer is currently logged in and the session has not expired. The ElectionsOfficer’s user object is available. The CreateElectoralDistrict from the extended use case is available. |
| Post-conditions | An already created ElectoralDistrict is added to CreateElectoralDistrict. |

|  |  |
| --- | --- |
| Name | Find an existing Political Party |
| ID | FindPoliticalParty |
| Version | 1 |
| Author | Donovan Todd |
| Date | 9/6/2016 |
| Summary | An extension use case to find an existing Political Party; used by CreatePoliticalParty use case |
| Basic Path | 1. The system displays all created PoliticalParties. Done and Cancel buttons are also visible. 2. The ElectionsOfficer selects a created PoliticalParty and clicks Done. |
| Alternative Paths | 1. In step 1, if the user presses the Cancel button, the use case is terminated and no changes to the system are made. |
| Exception Paths | N/A |
| Extension Points | N/A |
| Triggers | The ElectionsOfficer has elected to find an already created Political Party within the CreatePoliticalParty use case. |
| Assumption | N/A |
| Pre-conditions | The ElectionsOfficer is currently logged in and the session has not expired. The ElectionsOfficer’s user object is available. The CreatePoliticalParty from the extended use case is available. |
| Post-conditions | An already created PoliticalParty is added to CreatePoliticalParty. |

|  |  |
| --- | --- |
| Name | Find an existing Candidates |
| ID | FindCandidates |
| Version | 1 |
| Author | Donovan Todd |
| Date | 9/6/2016 |
| Summary | An extension use case to find an existing Candidate; used by CreateCandidates use cases |
| Basic Path | 1. The system displays all created Candidates. Done and Cancel buttons are also visible. 2. The ElectionsOfficer selects a created Candidates and clicks Done. |
| Alternative Paths | 1. In step 1, if the user presses the Cancel button, the use case is terminated and no changes to the system are made. |
| Exception Paths | N/A |
| Extension Points | N/A |
| Triggers | The ElectionsOfficer has elected to find an already created Candidate within the CreateCandidates use case. |
| Assumption | N/A |
| Pre-conditions | The ElectionsOfficer is currently logged in and the session has not expired. The ElectionsOfficer’s user object is available. The CreateCandidates from the extended use case is available. |
| Post-conditions | An already created Candidates is added to CreateCandidates. |

|  |  |
| --- | --- |
| Name | View an ElectoralDistrict |
| ID | ViewElectoralDistrict |
| Version | 1 |
| Author | Donovan Todd |
| Date | 9/6/2016 |
| Summary | A use case to show all existing Electoral District. |
| Basic Path | 1. The system displays all created ElectoralDistricts. Each ElectoralDistrict has a visible modify and delete button. A Cancel button is also visible. |
| Alternative Paths | 1. In step 1, if the user presses the Cancel button, the use case is terminated and no changes to the system are made. |
| Exception Paths | N/A |
| Extension Points | 1. In step 1, if the ElectionsOfficer selects modify, the system executes the ModifyElectoralDistrict use case extension; at its completion, the system displays a screen showing all of the ElectoralDistricts provided so far and goes to step 1 in the Basic Path. 2. In step 1, if the ElectionsOfficer selects delete, the system executes the DeleteElectoralDistrict use case extension; at its completion, the system displays a screen showing all of the ElectoralDistricts provided so far and goes to step 1 in the Basic Path. |
| Triggers | The ElectionsOfficer has elected to view an already created Electoral District. |
| Assumption | N/A |
| Pre-conditions | The ElectionsOfficer is currently logged in and the session has not expired. The ElectionsOfficer’s user object is available. The ElectoralDistrict user objects are available. |
| Post-conditions | N/A |

|  |  |
| --- | --- |
| Name | View a Candidate |
| ID | ViewCandidates |
| Version | 1 |
| Author | Donovan Todd |
| Date | 9/6/2016 |
| Summary | A use case to show all existing Candidates. |
| Basic Path | 1. The system displays all created Candidates. Each Candidate has a visible modify and delete button. A Cancel button is also visible. |
| Alternative Paths | 1. In step 1, if the user presses the Cancel button, the use case is terminated and no changes to the system are made. |
| Exception Paths | N/A |
| Extension Points | 1. In step 1, if the ElectionsOfficer selects modify, the system executes the ModifyCandidate use case extension; at its completion, the system displays a screen showing all of the Candidates provided so far and goes to step 1 in the Basic Path. 2. In step 1, if the ElectionsOfficer selects delete, the system executes the DeleteCandidate use case extension; at its completion, the system displays a screen showing all of the Candidates provided so far and goes to step 1 in the Basic Path. |
| Triggers | The ElectionsOfficer has elected to view an already created Candidate. |
| Assumption | N/A |
| Pre-conditions | The ElectionsOfficer is currently logged in and the session has not expired. The ElectionsOfficer’s user object is available. The Candidate user objects are available. |
| Post-conditions | N/A |

|  |  |
| --- | --- |
| Name | View a PoliticalParty |
| ID | ViewPoliticalParty |
| Version | 1 |
| Author | Donovan Todd |
| Date | 9/6/2016 |
| Summary | A use case to show all existing Political Parties. |
| Basic Path | 1. The system displays all created PoliticalParties. Each PoliticalParty has a visible modify and delete button. A Cancel button is also visible. |
| Alternative Paths | 1. In step 1, if the user presses the Cancel button, the use case is terminated and no changes to the system are made. |
| Exception Paths | N/A |
| Extension Points | 1. In step 1, if the ElectionsOfficer selects modify, the system executes the ModifyPoliticalParty use case extension; at its completion, the system displays a screen showing all of the PoliticalParties provided so far and goes to step 1 in the Basic Path. 2. In step 1, if the ElectionsOfficer selects delete, the system executes the DeletePoliticalParty use case extension; at its completion, the system displays a screen showing all of the PoliticalParties provided so far and goes to step 1 in the Basic Path. |
| Triggers | The ElectionsOfficer has elected to view an already created Political Party. |
| Assumption | N/A |
| Pre-conditions | The ElectionsOfficer is currently logged in and the session has not expired. The ElectionsOfficer’s user object is available. The PoliticalParty user objects are available. |
| Post-conditions | N/A |

|  |  |
| --- | --- |
| Name | View an Election |
| ID | ViewElection |
| Version | 1 |
| Author | Donovan Todd |
| Date | 9/6/2016 |
| Summary | A use case to show all existing Elections. |
| Basic Path | 1. The system displays all created Elections. Each Election has a visible modify, delete, OpenSystem, CloseSystem, and PublishResults buttons. A Cancel button is also visible. |
| Alternative Paths | 1. In step 1, if the user presses the Cancel button, the use case is terminated and no changes to the system are made. |
| Exception Paths | N/A |
| Extension Points | 1. In step 1, if the ElectionsOfficer selects modify, the system executes the ModifyElection use case extension; at its completion, the system displays a screen showing all of the Elections provided so far and goes to step 1 in the Basic Path. 2. In step 1, if the ElectionsOfficer selects delete, the system executes the DeleteElection use case extension; at its completion, the system displays a screen showing all of the Elections provided so far and goes to step 1 in the Basic Path. 3. In step 1, if the ElectionsOfficer selects OpenSystem, the system executes the OpenSystem use case extension; at its completion, the system displays a screen showing all of the Elections provided so far and goes to step 1 in the Basic Path. 4. In step 1, if the ElectionsOfficer selects CloseSystem, the system executes the CloseSystem use case extension; at its completion, the system displays a screen showing all of the Elections provided so far and goes to step 1 in the Basic Path. 5. In step 1, if the ElectionsOfficer selects PublishResults, the system executes the PublishResults use case extension; at its completion, the system displays a screen showing all of the Elections provided so far and goes to step 1 in the Basic Path. |
| Triggers | The ElectionsOfficer has elected to view an already created Election. |
| Assumption | N/A |
| Pre-conditions | The ElectionsOfficer is currently logged in and the session has not expired. The ElectionsOfficer’s user object is available. The Election user objects are available. |
| Post-conditions | N/A |

|  |  |
| --- | --- |
| Name | View an Issue |
| ID | ViewIssue |
| Version | 1 |
| Author | Donovan Todd |
| Date | 9/6/2016 |
| Summary | A use case to show all existing Issues. |
| Basic Path | 1. The system displays all created Issues. Each Issue has a visible modify, delete, OpenSystem, CloseSystem, and PublishResults buttons. A Cancel button is also visible. |
| Alternative Paths | 1. In step 1, if the user presses the Cancel button, the use case is terminated and no changes to the system are made. |
| Exception Paths | N/A |
| Extension Points | 1. In step 1, if the IssuesOfficer selects modify, the system executes the ModifyIssue use case extension; at its completion, the system displays a screen showing all of the Issues provided so far and goes to step 1 in the Basic Path. 2. In step 1, if the ElectionsOfficer selects delete, the system executes the DeleteIssue use case extension; at its completion, the system displays a screen showing all of the Issues provided so far and goes to step 1 in the Basic Path. 3. In step 1, if the ElectionsOfficer selects OpenSystem, the system executes the OpenSystem use case extension; at its completion, the system displays a screen showing all of the Issues provided so far and goes to step 1 in the Basic Path. 4. In step 1, if the ElectionsOfficer selects CloseSystem, the system executes the CloseSystem use case extension; at its completion, the system displays a screen showing all of the Issues provided so far and goes to step 1 in the Basic Path. 5. In step 1, if the ElectionsOfficer selects PublishResults, the system executes the PublishResults use case extension; at its completion, the system displays a screen showing all of the Issues provided so far and goes to step 1 in the Basic Path. |
| Triggers | The ElectionsOfficer has elected to view an already created Issue. |
| Assumption | N/A |
| Pre-conditions | The ElectionsOfficer is currently logged in and the session has not expired. The ElectionsOfficer’s user object is available. The Issue user objects are available. |
| Post-conditions | N/A |

|  |  |
| --- | --- |
| Name | Modify an existing Electoral District |
| ID | ModifyElectoralDistrict |
| Version | 1 |
| Author | Donovan Todd |
| Date | 9/6/2016 |
| Summary | An extension use case to modify an existing Electoral District; used by ViewElectoralDistrict use case. |
| Basic Path | 1. The system displays the prompt to enter: name, area; Done and Cancel buttons are visible. 2. The ElectionsOfficer provides the required information and clicks Done. 3. The system modifies an existing ElectoralDistrict with the provided data, if one was selected during the FindElectoralDistrict use case execution. The system displays a new screen in which all the created ElectoralDistricts are shown. The screen displays 2 buttons: Done and Cancel as well. |
| Alternative Paths | 1. In step 1 and 3, if the user presses the Cancel button, the use case is terminated and no changes to the system are made. |
| Exception Paths | 1. In step 3, if the system determines that an ElectoralDistrict with the user inputted name already exists, the system displays the error message DuplicateElectoralDistrictName and the system goes back to step 1 in the Basic Path, retaining all previously provided attribute values. |
| Extension Points |  |
| Triggers | The ElectionsOfficer has elected to modify an existing Electoral District within the ViewElectoralDistrict use case. |
| Assumption | N/A |
| Pre-conditions | The ElectionsOfficer is currently logged in and the session has not expired. The ElectionsOfficer’s user object is available. The ElectoralDistricts from the extended use case is available. |
| Post-conditions | A modified ElectoralDistrict user objects is now available to other use cases available to the same ElectionsOfficer. |

|  |  |
| --- | --- |
| Name | Modify an existing Political Party |
| ID | ModifyPoliticalParty |
| Version | 1 |
| Author | Donovan Todd |
| Date | 9/6/2016 |
| Summary | An extension use case to modify an existing Political Party; used by ViewPoliticalParty use case. |
| Basic Path | 1. The system displays the prompt to enter: name; Done and Cancel buttons are visible. 2. The ElectionsOfficer provides the required information and clicks Done. 3. The system modifies an existing PoliticalParty with the provided data, if one was selected during the FindPoliticalParty use case execution. The system displays a new screen in which all the created PoliticalParties are shown. The screen displays 2 buttons: Done and Cancel as well. |
| Alternative Paths | 1. In step 1 and 3, if the user presses the Cancel button, the use case is terminated and no changes to the system are made. |
| Exception Paths | 1. In step 3, if the system determines that a PoliticalParty with the user inputted name already exists, the system displays the error message DuplicatePoliticalPartyName and the system goes back to step 1 in the Basic Path, retaining all previously provided attribute values. |
| Extension Points |  |
| Triggers | The ElectionsOfficer has elected to modify an existing Political Party within the ViewPoliticalParty use case. |
| Assumption | N/A |
| Pre-conditions | The ElectionsOfficer is currently logged in and the session has not expired. The ElectionsOfficer’s user object is available. The PoliticalParties from the extended use case is available. |
| Post-conditions | A modified PoliticalParty user objects is now available to other use cases available to the same ElectionsOfficer. |

|  |  |
| --- | --- |
| Name | Modify an existing Candidate |
| ID | ModifyCandidate |
| Version | 1 |
| Author | Donovan Todd |
| Date | 9/6/2016 |
| Summary | An extension use case to modify an existing Candidate; used by ViewCandidate use case. |
| Basic Path | 1. The system displays the prompt to enter: name, political party; Done and Cancel buttons are visible. 2. The ElectionsOfficer provides the required information and clicks Done. 3. The system modifies an existing Candidate with the provided data, if one was selected during the FindCandidate use case execution. The system displays a new screen in which all the created Candidates are shown. The screen displays 2 buttons: Done and Cancel as well. |
| Alternative Paths | 1. In step 1 and 3, if the user presses the Cancel button, the use case is terminated and no changes to the system are made. |
| Exception Paths | 1. In step 3, if the system determines that a Candidate with the user inputted name already exists, the system displays the error message DuplicateCandidateName and the system goes back to step 1 in the Basic Path, retaining all previously provided attribute values. |
| Extension Points |  |
| Triggers | The ElectionsOfficer has elected to modify an existing Candidate within the ViewCandidate use case. |
| Assumption | N/A |
| Pre-conditions | The ElectionsOfficer is currently logged in and the session has not expired. The ElectionsOfficer’s user object is available. The Candidates from the extended use case is available. |
| Post-conditions | A modified Candidate user objects is now available to other use cases available to the same ElectionsOfficer. |

|  |  |
| --- | --- |
| Name | Modify an existing Election |
| ID | ModifyElection |
| Version | 1 |
| Author | Donovan Todd |
| Date | 9/6/2016 |
| Summary | An extension use case to modify an existing Election; used by ViewElection use case. |
| Basic Path | 1. The system displays the prompt to enter: name, candidates and electoral districts; Done and Cancel buttons are visible. 2. The ElectionsOfficer provides the required information and clicks Done. 3. The system modifies an existing Election with the provided data, if one was selected during the FindElection use case execution. The system displays a new screen in which all the created Elections are shown. The screen displays 2 buttons: Done and Cancel as well. |
| Alternative Paths | 1. In step 1 and 3, if the user presses the Cancel button, the use case is terminated and no changes to the system are made. |
| Exception Paths | 1. In step 3, if the system determines that an Election with the user inputted name already exists, the system displays the error message DuplicateElectionName and the system goes back to step 1 in the Basic Path, retaining all previously provided attribute values. |
| Extension Points |  |
| Triggers | The ElectionsOfficer has elected to modify an existing Election within the ViewElection use case. |
| Assumption | N/A |
| Pre-conditions | The ElectionsOfficer is currently logged in and the session has not expired. The ElectionsOfficer’s user object is available. The Elections from the extended use case is available. |
| Post-conditions | A modified Election user objects is now available to other use cases available to the same ElectionsOfficer. |

|  |  |
| --- | --- |
| Name | Modify an existing Issue |
| ID | ModifyIssue |
| Version | 1 |
| Author | Donovan Todd |
| Date | 9/6/2016 |
| Summary | An extension use case to modify an existing Issue; used by ViewIssue use case. |
| Basic Path | 1. The system displays the prompt to enter: name, ElectoralDistricts; Done and Cancel buttons are visible. 2. The ElectionsOfficer provides the required information and clicks Done. 3. The system modifies an existing Issue with the provided data, if one was selected during the FindIssue use case execution. The system displays a new screen in which all the created Issues are shown. The screen displays 2 buttons: Done and Cancel as well. |
| Alternative Paths | 1. In step 1 and 3, if the user presses the Cancel button, the use case is terminated and no changes to the system are made. |
| Exception Paths | 1. In step 3, if the system determines that an Issue with the user inputted name already exists, the system displays the error message DuplicateIssueName and the system goes back to step 1 in the Basic Path, retaining all previously provided attribute values. |
| Extension Points |  |
| Triggers | The ElectionsOfficer has elected to modify an existing Issue within the ViewIssue use case. |
| Assumption | N/A |
| Pre-conditions | The ElectionsOfficer is currently logged in and the session has not expired. The ElectionsOfficer’s user object is available. The Issues from the extended use case is available. |
| Post-conditions | A modified Issue user objects is now available to other use cases available to the same ElectionsOfficer. |

|  |  |
| --- | --- |
| Name | Modify an the Elections Officer’s Profile |
| ID | ModifyProfile |
| Version | 1 |
| Author | Donovan Todd |
| Date | 9/6/2016 |
| Summary | This use case details modifying the ElectionsOfficer’s profile. |
| Basic Path | 1. The system displays the prompt to enter: name, email, password; Done and Cancel buttons are visible. 2. The ElectionsOfficer provides the required information and clicks Done. 3. The system modifies an existing ElectionsOfficer user object with the provided data. The system displays a new screen in which all the user info is shown. The screen displays 2 buttons: Done and Cancel as well. |
| Alternative Paths | 1. In step 1 and 3, if the user presses the Cancel button, the use case is terminated and no changes to the system are made. |
| Exception Paths | N/A |
| Extension Points |  |
| Triggers | The ElectionsOfficer has elected to modify his profile. |
| Assumption | N/A |
| Pre-conditions | The ElectionsOfficer is currently logged in and the session has not expired. The ElectionsOfficer’s user object is available. |
| Post-conditions | A modified ElectionsOfficer user objects is now available to other use cases available to the same ElectionsOfficer. |

|  |  |
| --- | --- |
| Name | Delete an existing Electoral District |
| ID | DeleteElectoralDistrict |
| Version | 1 |
| Author | Donovan Todd |
| Date | 9/6/2016 |
| Summary | An extension use case to delete an existing Electoral District; used by ViewElectoralDistrict use case. |
| Basic Path | 1. The system displays the prompt with the buttons Done and Cancel. 2. The ElectionsOfficer clicks Done. 3. The system deletes the selected ElectoralDistrict, The system displays a new screen in which all the created ElectoralDistricts are shown. The screen displays 2 buttons: Done and Cancel as well. |
| Alternative Paths | 1. In step 1 and 3, if the user presses the Cancel button, the use case is terminated and no changes to the system are made. |
| Exception Paths | N/A |
| Extension Points | N/A |
| Triggers | The ElectionsOfficer has elected to delete an existing Electoral District within the ViewElectoralDistrict use case. |
| Assumption | N/A |
| Pre-conditions | The ElectionsOfficer is currently logged in and the session has not expired. The ElectionsOfficer’s user object is available. The ElectoralDistricts from the extended use case is available. |
| Post-conditions | The deleted ElectoralDistrict user objects is no longer available to other use cases available to the same ElectionsOfficer. |

|  |  |
| --- | --- |
| Name | Delete an existing Political Party |
| ID | DeletePoliticalParty |
| Version | 1 |
| Author | Donovan Todd |
| Date | 9/6/2016 |
| Summary | An extension use case to delete an existing Political Party; used by ViewPoliticalParty use case. |
| Basic Path | 1. The system displays the prompt with the buttons Done and Cancel. 2. The ElectionsOfficer clicks Done. 3. The system deletes the selected PoliticalParty, The system displays a new screen in which all the created PoliticalParties are shown. The screen displays 2 buttons: Done and Cancel as well. |
| Alternative Paths | 1. In step 1 and 3, if the user presses the Cancel button, the use case is terminated and no changes to the system are made. |
| Exception Paths | N/A |
| Extension Points | N/A |
| Triggers | The ElectionsOfficer has elected to delete an existing Political Party within the ViewPoliticalParty use case. |
| Assumption | N/A |
| Pre-conditions | The ElectionsOfficer is currently logged in and the session has not expired. The ElectionsOfficer’s user object is available. The PoliticalParties from the extended use case is available. |
| Post-conditions | The deleted PoliticalParty user objects is no longer available to other use cases available to the same ElectionsOfficer. |

|  |  |
| --- | --- |
| Name | Delete an existing Candidate |
| ID | DeleteCandidate |
| Version | 1 |
| Author | Donovan Todd |
| Date | 9/6/2016 |
| Summary | An extension use case to delete an existing Candidate; used by ViewCandidate use case. |
| Basic Path | 1. The system displays the prompt with the buttons Done and Cancel. 2. The ElectionsOfficer clicks Done. 3. The system deletes the selected Candidate, The system displays a new screen in which all the created Candidates are shown. The screen displays 2 buttons: Done and Cancel as well. |
| Alternative Paths | 1. In step 1 and 3, if the user presses the Cancel button, the use case is terminated and no changes to the system are made. |
| Exception Paths | N/A |
| Extension Points | N/A |
| Triggers | The ElectionsOfficer has elected to delete an existing Candidate within the ViewCandidate use case. |
| Assumption | N/A |
| Pre-conditions | The ElectionsOfficer is currently logged in and the session has not expired. The ElectionsOfficer’s user object is available. The Candidates from the extended use case is available. |
| Post-conditions | The deleted Candidate user objects is no longer available to other use cases available to the same ElectionsOfficer. |

|  |  |
| --- | --- |
| Name | Delete an existing Election |
| ID | DeleteElection |
| Version | 1 |
| Author | Donovan Todd |
| Date | 9/6/2016 |
| Summary | An extension use case to delete an existing Election; used by ViewElection use case. |
| Basic Path | 1. The system displays the prompt with the buttons Done and Cancel. 2. The ElectionsOfficer clicks Done. 3. The system deletes the selected Election, The system displays a new screen in which all the created Elections are shown. The screen displays 2 buttons: Done and Cancel as well. |
| Alternative Paths | 1. In step 1 and 3, if the user presses the Cancel button, the use case is terminated and no changes to the system are made. |
| Exception Paths | N/A |
| Extension Points | N/A |
| Triggers | The ElectionsOfficer has elected to delete an existing Election within the ViewElection use case. |
| Assumption | N/A |
| Pre-conditions | The ElectionsOfficer is currently logged in and the session has not expired. The ElectionsOfficer’s user object is available. The Elections from the extended use case is available. |
| Post-conditions | The deleted Election user objects is no longer available to other use cases available to the same ElectionsOfficer. |

|  |  |
| --- | --- |
| Name | Delete an existing Issue |
| ID | DeleteIssue |
| Version | 1 |
| Author | Donovan Todd |
| Date | 9/6/2016 |
| Summary | An extension use case to delete an existing Issue; used by ViewIssue use case. |
| Basic Path | 1. The system displays the prompt with the buttons Done and Cancel. 2. The ElectionsOfficer clicks Done. 3. The system deletes the selected Issue, The system displays a new screen in which all the created Issues are shown. The screen displays 2 buttons: Done and Cancel as well. |
| Alternative Paths | 1. In step 1 and 3, if the user presses the Cancel button, the use case is terminated and no changes to the system are made. |
| Exception Paths | N/A |
| Extension Points | N/A |
| Triggers | The ElectionsOfficer has elected to delete an existing Issue within the ViewIssue use case. |
| Assumption | N/A |
| Pre-conditions | The ElectionsOfficer is currently logged in and the session has not expired. The ElectionsOfficer’s user object is available. The Issues from the extended use case is available. |
| Post-conditions | The deleted Issue user objects is no longer available to other use cases available to the same ElectionsOfficer. |

|  |  |
| --- | --- |
| Name | Open System For Voting |
| ID | OpenSystem |
| Version | 1 |
| Author | Donovan Todd |
| Date | 9/7/2016 |
| Summary | An extension use case to open an existing Issue or Election; used by ViewElections and ViewIssues use cases. |
| Basic Path | 1. The system displays the prompt with the buttons Done and Cancel. 2. The ElectionsOfficer clicks Done. 3. The system opens the selected Election or Issue for voting by voters, the system displays a new screen in which all the created Elections or Issues are shown. The screen displays 2 buttons: Done and Cancel as well. |
| Alternative Paths | 1. In steps 1 and 3, if the user presses the Cancel button, the use case is terminated and no changes to the system are made. |
| Exception Paths | 1. In step 1, if the system determines that an Issue or Election is already open for voting, the system displays the error message AlreadyOpen and the system goes back to step 1 in the Basic Path, retaining all previously provided attribute values. |
| Extension Points | N/A |
| Triggers | The ElectionsOfficer has elected to open an existing Election or Issue within the ViewElections or ViewIssues use cases. |
| Assumption | N/A |
| Pre-conditions | The ElectionsOfficer is currently logged in and the session has not expired. The ElectionsOfficer’s user object is available. The Issues or Elections from the extended use cases are available. |
| Post-conditions | The selected Issue or Election is now open and can now be voted on by voters in the assigned electoral district. |

|  |  |
| --- | --- |
| Name | Close System For Voting |
| ID | CloseSystem |
| Version | 1 |
| Author | Donovan Todd |
| Date | 9/7/2016 |
| Summary | An extension use case to open an existing Issue or Election; used by ViewElections and ViewIssues use cases. |
| Basic Path | 1. The prompt displays the buttons Done and Cancel. 2. The ElectionsOfficer clicks Done. 3. The system closes the selected Election or Issue for voting by voters, the system displays a new screen in which all the created Elections or Issues are shown. The screen displays 2 buttons: Done and Cancel as well. |
| Alternative Paths | 1. In steps 1 and 3, if the user presses the Cancel button, the use case is terminated and no changes to the system are made. |
| Exception Paths | 1. In step 1, if the system determines that an Issue or Election is already closed for voting, the system displays the error message AlreadyClosed and the system goes back to step 1 in the Basic Path, retaining all previously provided attribute values. |
| Extension Points | N/A |
| Triggers | The ElectionsOfficer has elected to close an existing Election or Issue within the ViewElections or ViewIssues use cases. |
| Assumption | N/A |
| Pre-conditions | The ElectionsOfficer is currently logged in and the session has not expired. The ElectionsOfficer’s user object is available. The Issues or Elections from the extended use cases are available. |
| Post-conditions | The selected Issue or Election is now closed and can now not be voted on by voters in the assigned electoral district. |

|  |  |
| --- | --- |
| Name | Publish Results |
| ID | PublishResults |
| Version | 1 |
| Author | Donovan Todd |
| Date | 9/7/2016 |
| Summary | An extension use case to publish the results of an Issue or Election; used by ViewElections and ViewIssues use cases. |
| Basic Path | 1. The system displays the results of the Issue or Election so that it can be validated by the ElectionsOfficer. The prompt also displays the buttons Done and Cancel. 2. The ElectionsOfficer clicks Done. 3. The system publishes the results of the selected Election or Issue by sending the results to all voters that voted in the selected Election or Issue, the system displays a new screen in which all the created Elections or Issues are shown. The screen displays 2 buttons: Done and Cancel as well. |
| Alternative Paths | 1. In steps 1 and 3, if the user presses the Cancel button, the use case is terminated and no changes to the system are made. |
| Exception Paths | 1. In step 1, if the system determines that an Issue or Election isnt closed for voting, the system displays the error message StillOpen and the system goes back to step 1 in the Basic Path, retaining all previously provided attribute values. 2. In step 1, if the system determines that an Issue or Election was never opened for voting and the system displays the error message NoVotesAvailable and the system goes back to step 1 in the Basic Path, retaining all previously provided attribute values. |
| Extension Points | N/A |
| Triggers | The ElectionsOfficer has elected to publish the results of a closed Election or Issue within the ViewElections or ViewIssues use cases. |
| Assumption | N/A |
| Pre-conditions | The ElectionsOfficer is currently logged in and the session has not expired. The ElectionsOfficer’s user object is available. The Issues or Elections from the extended use cases are available. |
| Post-conditions | The selected Issue or Election’s results are now published to all voters that voted on it and can now not be viewed by voters in the assigned electoral district. |