Updated Document

**ntroduction**

We are designing a state of the art E-Voting software called Better Ballot, which will be used maintain a democracy or any other voting needs. Better Ballot will be designed to be versatile and configurable to the user needs, with the goal to make it easy to use so any one can use.

**Stakeholders**

Ballots can be hard to maintain, setup, and calculate accurately. Which make a easy process a confusing process. Better Ballot will be used all around the world insure every voice is heard. From governments, citizens, and businesses.

**Positioning**

Without an e-voting system, ballots can be confusing and hard to maintain accuracy of the votes. Our goal is to make voting easy and convenient as possible. We want Better Ballot to capable of being used by anyone and everyone. We also want administrations to be able to configure a ballot or ballots with ease. Everyone’s voice will be heard with Better Ballot.

Ballot Supplemental Specifications

. Functionality

a. Stable Remote Access

The ballot must be easily accessible via internet for anyone with a connection during voting periods.

2. Usability

a. Not OS Discriminatory

The ballot must be accessible for anyone regardless of their operating system.

b. Self Explanatory

Any user should not require outside knowledge or teaching to be able to use the Ballot system.

3. Reliability

a. Security

The Ballot should keep user voting data safe and away from other non-admin users

b. Legal Standards

The Ballot should support and adhere to current legal standards set by the U.S. government.

4. Performance

a. Quick Response Time

The system should respond quickly to user input.

b. Multiple User Compatibility

Multiple people should be able to use the ballot without any system issues.

5. Compatibility

a. The ballot should run on any and all modern day computers.

Risk List

|  |  |  |
| --- | --- | --- |
| ID | Risk Description | Impact |
| 1 | Hardware | Possibly expensive hardware |
| 2 | Time | Delayed Results |
| 3 | Legal | Leaks of possible important info |
|  |  |  |
|  |  |  |
|  |  |  |

BetterBallot Glossary  
9.19.2018

Alex Ross, Soren Christiansen, Johnnie Cooper

github.com/hotdoghotdoghot/gangster

horizontal line

BetterBallot - App designed to provide a streamlined and reliable voting service

Client - the process running on a machine that a user will use to place votes.

User - the person voting

Ballot Administrator - Individual tasked with the upkeep and oversight of the BetterBallot clients and their host machines

Operation Contracts

Contract: createAccount

Operation: createAccount(ActionEvent: event)

Cross References: SSD

Preconditions: Account has not already been created and user is in account creation screen.

Postconditions: Account creation starts and user is prompted to give an userId, password, firstName, and lastName.

Contract: userLogin

Operation: userLogin(ActionEvent: event)

Cross References: SSD

Preconditions: User has not logged in and has an account

Postconditions: User is prompted to provide userId and password, User is logged in as either admin or voter

Contract: goToLoginScreen

Operation: goToLoginScreen(ActionEvent: event)

Cross References: SSD

Preconditions: LoginStage is not active

Postconditions: LoginStage is active

Contract: goToCreateUser

Operation: goToCreateUser(ActionEvent: event)

Cross References: SSD

Preconditions: Not already at createAccount

Postconditions: Brings UI to createAccount

Contract: cancelLogin

Operation: cancelLogin(ActionEvent: event)

Cross References: SSD

Preconditions: None

Postconditions: Exits platform

Contract: adminView

Operation: adminView(ActionEvent: event)

Cross References: SSD

Preconditions: User must be logged in as admin

Postconditions: Shows mainMenuStage and AdminMainView

Contract: statsView

Operation: statsView(ActionEvent: event)

Cross References: SSD

Preconditions: User must be logged in as admin

Postconditions: Shows statsStage

Contract: settingsView

Operation: settingsView(ActionEvent: event)

Cross References: SSD

Preconditions: User must be logged in as admin

Postconditions: Shows settingsStage

Contract: configureBallets

Operation: configureBallets(ActionEvent: event)

Cross References: SSD

Preconditions: User must be logged in as admin

Postconditions: Shows configBalletStage

Contract: configureUsers

Operation: configureUsers(ActionEvent: event)

Cross References: SSD

Preconditions: User must be logged in as admin

Postconditions: Shows configUsersStage

Contract: createBallet

Operation: createBallet(ballet: Ballet)

Cross References: SSD

Preconditions: User must be logged in as an admin

Postconditions: User is prompted to create ballet

Contract: editBallet

Operation: editBallet(ballet: Ballet)

Cross References: SSD

Preconditions: User must be logged in as an admin

Postconditions: User is prompted to select ballet to edit

Contract: deleteBallet

Operation: deleteBallet(ballet: Ballet)

Cross References: SSD

Preconditions: User must be logged in as an admin

Postconditions: User is prompted to select ballet to delete

Contract: editUser

Operation: editUser(user: User)

Cross References: SSD

Preconditions: User must be logged in as an admin

Postconditions: User is prompted to select user to edit

Contract: deleteUser

Operation: deleteUser(user: User)

Cross References: SSD

Preconditions: User must be logged in as an admin

Postconditions: User is prompted to select user to delete

Contract: selectVote

Operation: selectVote(choice: string)

Cross References: SSD

Preconditions: A vote selection is underway

Postconditions: The user’s vote choice is selected(variable update)

Contract: castVote

Operation: castVote()

Cross References: SSD

Preconditions: A vote choice has been selected

Postconditions: The user voting data is saved and updated on the server

Use Cases Scope Doc

https://github.com/hotdoghotdoghot/Gangster

horizontal line

# 05 October 2018

**Small list of features that will not be implemented in our use cases**

* There will be no functionality for users to contact representatives
* The program will not be able to submit any kind of legislation for consideration
* The software will not be made available for non-governmental elections
* The software will not be made available for use unless under the supervision of one or more government employees

## Use cases should should cover all user functionality

* All use cases that are required by governments for deployment in their respective nations will be implemented incrementally.
  + Support for different election styles and voter registration processes
* Most common use cases such as new user registering, and already registered user voting should be well covered as these are likely the most frequent use cases for our software.
  + Developmental priority should be placed on polishing these use cases as they are the basis on which the functionality of the project will be judged

**Use case UC1:** Cast Vote

**Context of use:** User cast vote to a ballot   
**Scope:** BetterBallot E-Voting Application  
**Level:** User-Goal  
**Primary Actor:** Voter  
**Stakeholders and Interests:**

* **Voters:** Wants to vote, in an easy non complicated manner which is accurately tracked, counted, and private.
* **Ballot Administer:** Wants a secure accurate count of all the votes fast as possible with the least amount of manual work possible.
* **Ballot Candidates:** Wants a secure accurate count of all the votes, which are accurately tracked for a fair and reliable ballot.

**Precondition:** Voter has their identification verified and authenticated.

**Success Guarantee:** Voter’s individual vote is saved and the current ballot results are updated for a live analysis of the ballots standings.

**Main Success Scenario**

* **Step 1 –** Voter arrives to E-Voting application and is at a screen where you can either login in or register to vote.
* **Step 2** – Voter Registers to Vote by entering all their info into the application.
* **Step 3 –** Voter then logs in with newly created account.
* **Step 4 –** System then displays the current ballot candidates.
* **Step 5 –** Voter then selects which candidate they would like to cast their vote for**.**
* **Step 6 –** System then presents the voters choose, then asks the user to verify that his chose is correct to avoid errors.
* **Step 7 –** System then verifies the voter has only cast one vote and is a valid entry.
* **Step 8 –** System then records vote into the ballot, and calculates the current ballot standings.
* **Step 9 –** System then displays a success message to the voter that their vote was successfully saved into the ballot.

**Extensions:**

\*a. At any time, The voter can login and change their vote.

Voter logs in and clicks “change vote”

System then displays the ballot candidates for the voter to select

Voter then selects the candidate that they would like to change their vote to.

System changes their vote and displays success message.

\*b. At any time, if the system crashes during voting process

1. Ballot Administrator restarts the system.

2. Voter then logs back in, they then restarts the voting process(steps 4-9).

1-6a. Voter wants to cancel their vote.

1. System presents a “cancel vote” option at bottom of each screen.
2. Voter selects the “cancel vote”, which the system then navigates back to the main menu screen.
3. The user select the “log out” option.

6a. Voter selects wrong candidate

1. Voter selects “No”, when system asks if the candidate is correct at step 6.

2. System then restarts at step 4.

7a. Voter tried to cast more than one vote.

1. System signals error and prompts voter they can only vote once.

8a. System fails to save voters vote.

1. System signals error and prompts voter to notify a ballot administrator.
2. Ballot Administrator checks network connectivity, and database connectivity

2a Error found, contact technical support.

1. Ballot Administrator Restart system
2. Voter restarts voting process (steps 4-9).

**Special Requirements:**

Large text to accommodate any special needs.

Easy to use for everyone so everyone is able to vote.

Network Connectivity.

**Technology & Data Variations List:**

1-7a. Mouse and Keyboard to make selection and Voter registration.

**Frequency of Occurrence:** Could be nearly continuous, with up to 1 minute break in-between voters.

**Related Information:** None.

**Use case UC2:** Create Ballot

**Context of use:** Ballot administrator creates a ballot poll  
**Scope:** BetterBallot E-Voting Application  
**Level:** User-Goal  
**Primary Actor:** Ballot Administrator   
**Stakeholders and Interests:**

* **Voters:** Wants to vote, in an easy none complicated manner which is accurately tracked, counted, and private which requires a ballot to be configured.
* **Ballot Administer:** Wants a secure accurate count of all the votes fast as possible with the less amount of manual work possible.
* **Ballot Candidates:** Wants a secure accurate count of all the votes, which are accurately tracked for a fair accurate ballot.

**Precondition:** Administrator has an admin account to configure a new ballot.

**Success Guarantee:** Ballot Administrator configures a ballot so voters have a ballot to cast their vote.

**Main Success Scenario**

* **Step 1 –** Ballot Administrator logs on to their administrator account.
* **Step 2** – Ballot Administrator then navigates to administrator section and selects the “Create Ballot” option.
* **Step 3 –** System then presents a dialog screen that starts the setup of the ballot.
* **Step 4 –** The ballot administrator then names the ballot, a ballot statement for which the voters will be voting for, inserts the ballots candidates, start time, and end time.
* **Step 5 –** The system then confirms the entries are correct with the ballot administrator**.**
* **Step 6 –** The ballot is then saved into the system.
* **Step 7 –** The system will now display the ballot to the voters when it is opened.

**Extensions:**

\*a. At any time, the ballot administrator can delete, or edit an existing ballot.

The ballot administrator logs in and clicks “Administration” tab

System then displays the current ballots that are created.

The ballot administrator selects the ballot he wishes to edit or delete.

The system then displays configuration dialog screen which the ballot administrator can edit or delete the ballot.

1-5a. Ballot administrator wants to cancel the in the middle of creating a ballot.

1. System presents a “cancel ballot” option at bottom the ballot setup dialog screen.
2. Voter selects the “cancel ballot”, which the system then navigates back to the main menu screen.

6a. System fails to save ballot.

1. System signals error and prompts voter to notify a ballot administrator.
2. Ballot Administrator checks network connectivity, and database connectivity

2a Error found, contact technical support.

1. Ballot Administrator Restart system
2. Voter restarts voting process (steps 4-9).

**Special Requirements:**

Network Connectivity.

**Technology & Data Variations List:**

1-7a. Mouse and Keyboard to configure ballot.

**Frequency of Occurrence:** Not continuous, but could be often.

**Related Information:** None.

**Use case UC3:** Access Statistics

**Context of use:** Ballot administrator reviews statistics  
**Scope:** BetterBallot E-Voting Application  
**Level:** User-Goal  
**Primary Actor:** Ballot Administrator   
**Stakeholders and Interests:**

* **Voters:** Wants to vote, in an easy none complicated manner which is accurately tracked, counted, and private which requires a ballot to be configured.
* **Ballot Administer:** Wants a secure accurate count of all the votes fast as possible with the less amount of manual work possible.
* **Ballot Candidates:** Wants a secure accurate count of all the votes, which are accurately tracked for a fair accurate ballot.

**Precondition:** Administrator has an admin account to access statistical information.

**Success Guarantee:** Ballot Administrator accesses information via the “Statistics” page and is able to visualize poll information

**Main Success Scenario**

* **Step 1 –** Ballot Administrator logs on to their administrator account.
* **Step 2** – Ballot Administrator then navigates to administrator section and selects the “Statistics” option.
* **Step 3 –** System then presents a screen that presents statistical data on the current ballot(s)

**Extensions:**

\*a. At any time, the ballot administrator can access this information so long as it is in accordance with the laws surrounding the current election/ ballot.

The ballot administrator logs in and clicks “Administration” tab

The ballot administrator selects the “Statistics” option

The system then displays the current statistics.

3a. System fails to access statistics.

1. System signals an error and no statistics are shown.
2. Ballot Administrator checks network connectivity, and database connectivity 2a Error found, contact technical support.
3. Ballot Administrator Restart system
4. Administrator then logs in and statistics are shown.

**Special Requirements:**

Network Connectivity.

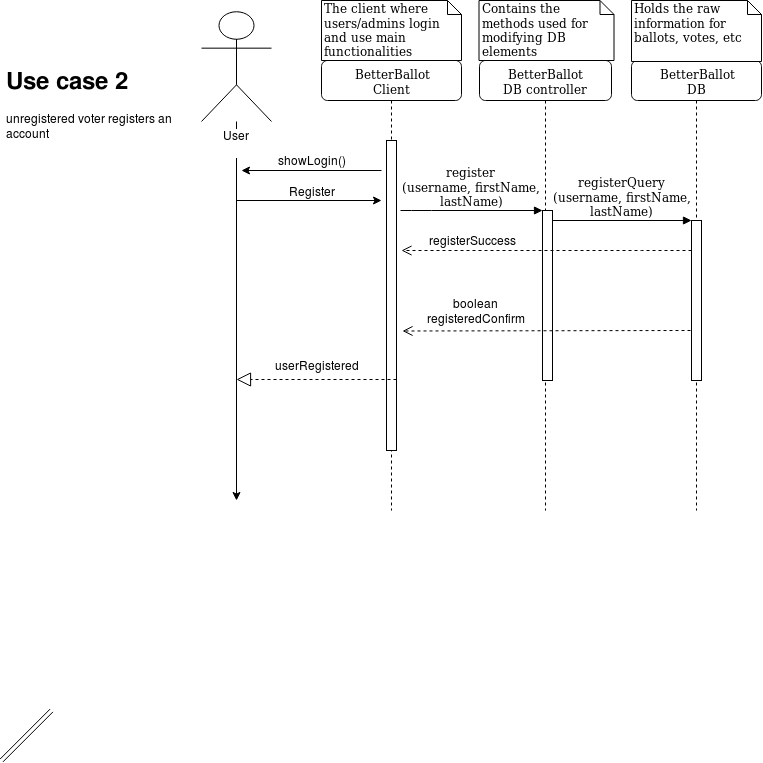
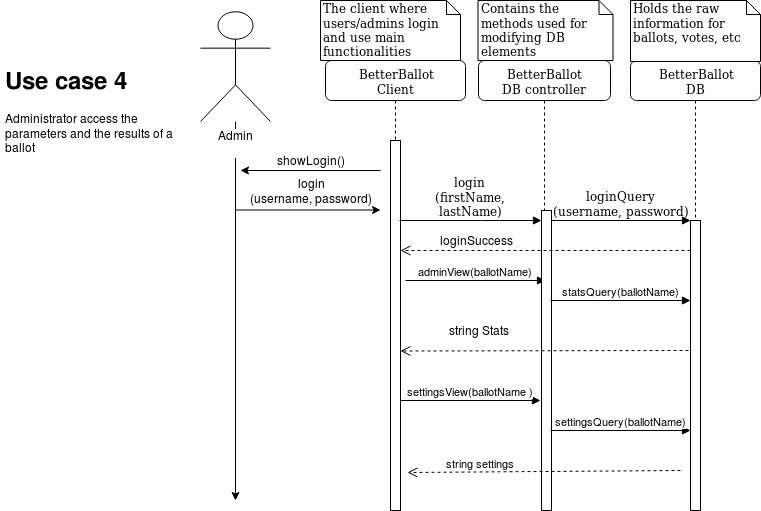
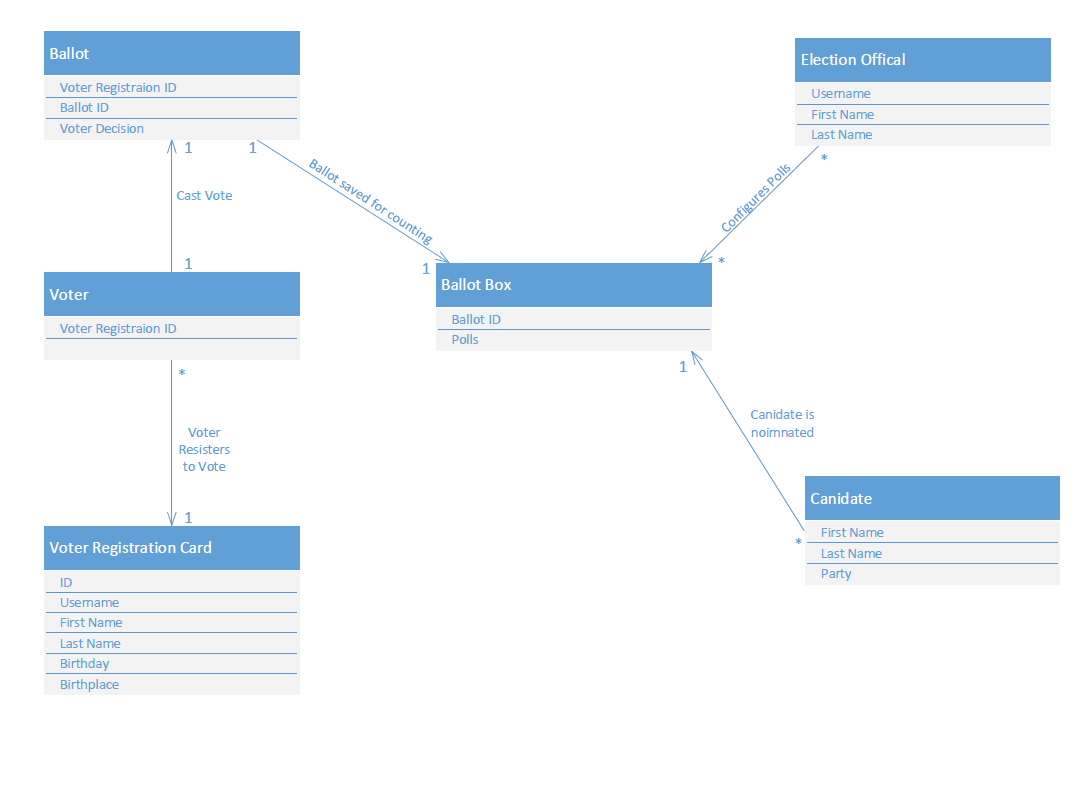
**Technology & Data Variations List:**

1-7a. Mouse and Keyboard to navigate system.

**Frequency of Occurrence:** Not continuous, but could be often.

**Related Information:** None.

Domain Model

System Sequence Diagrams

