Voting-System Operations & Setup

Setup
Before running the program, make sure to create a database called "SystemDatabase." Before the code is run, this database should have no tables. If tables do already exist, an error may occur due to the code trying to create a table that already has been made, and if the table has data from the previous voting session, it could be a security risk as well as causing the system to incorrectly handle data operations.

The GUI will look best when run from NetBeans (8.1), but is still functional in other environments. Before running the program, there are several static final variables that should be checked. Depending on your MySQL settings, you will likely need to change USERNAME, PASSWORD, etc., and possibly change the DATABASE_NAME variable if you choose to use a different name for the database. These setup variables can be found in the VotingSystem class.

```
 * @author Spenser, Casey
 *
 */
public class VotingSystem {
    public final static String dbURL = "jdbc:mysql://localhost:3306/systemdatabase";
        public final static String USERNAME = "root";
        public final static String PASSWORD = "sesame"; //change to something more secure later
        public final static String SERVER_NAME = "localhost";
        public final static int PORT_NUMBER = 3306;
        public final static String DATABASE_NAME = "SystemDatabase";
```
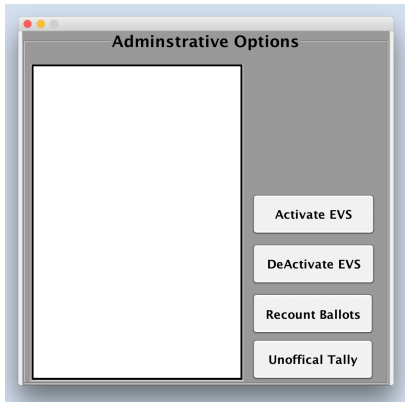
User Credentials
For valid administrator credentials, you can either access admin.txt or use "select * from [database name].administrators;" from the SQL command line. Similarly, for valid voter credentials, either access Voters.txt or use "select ID, fname, lname from [database name].administrators;" from the command line. [database name] refers to the name of the database, which should be SystemDatabase by default (alternatively, simply input the use [database name] command to avoid having to type it in each time).
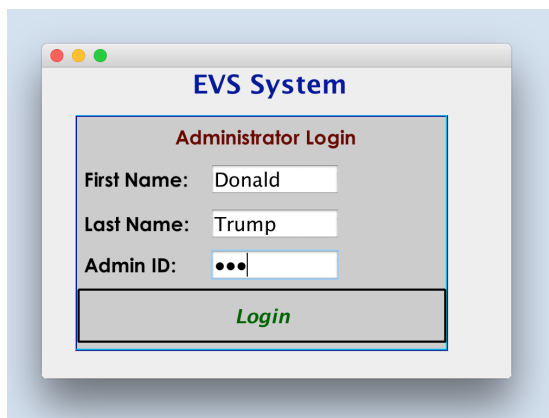
Running the Application
To first start up the system, you must input proper administrator credentials into their respective fields, and then click "login." After that, you can click Activate EVS, DeActivate EVS, Recount Ballots, or Unofficial Tally. To allow voters to vote, click the Activate EVS button. From there, you can input valid voter credentials at the top (voters who already voted will be invalid), make your choices, and click submit for each choice. After all 3 choices have been made, you will be presented with a red "x" button and a green check button. Click the red button to cancel your choices and start over, and click the green button to confirm your choices. You will then be presented with a confirmation that you have voted, at which time the current voter would leave and the next voter would arrive.

Administrative Options

To return to the administrative options screen, simply input valid admin credentials in the same fields you input voter credentials. To get an unofficial tally, click the unofficial tally button, which will present you with a list of candidates in the following format: "[last name], [first name] - [chair]: [tallies]". For a recount, click the Recount Ballots button, which will verify if the number of ballots matches the number of voters who have voted, and if it is not verified, the administrator will have to open up the LiveResults.txt file (which will be decrypted for the administrator), and manually recount the votes (which are anonymous).

Termination

To end the session, the administrator should click the DeActivate EVS button. The administrator should then drop the database by typing "drop database [database name];" (which should be "drop database SystemDatabase; by default) to clear all results. Alternatively, the administrator could individually drop the tables within the database by calling "drop table [table name];" (The "candidates" and "voters" tables are the most important to drop).

About LiveResults.txt

This system simulates having a list of registered voters from a proper and authenticated authority. The voters.txt file is populated with these registered voters. All voters have an unique ID which is used to login to cast a vote.

After running the system there will be two versions of the LiveResults.txt file generated, one encrypted and the other not. These will be located in the same directory that the project is saved in. The encrypted version simulates how the

```
PRES    SENT    HOUS    TIMESTAMP
002     007     012     Thu Dec 08 20:11:29 EST 2016
001     011     008     Thu Dec 08 20:13:13 EST 2016
004     006     009     Thu Dec 08 20:14:09 EST 2016
003     005     010     Thu Dec 08 20:15:30 EST 2016
901     901     008     Thu Dec 08 20:24:22 EST 2016
```

system would protect a voters identification and their votes. The file will consist of time stamps of when every voter casted their vote and the candidates they voted for. The voter id is not listed in the file for security purposes. Candidates in the file are identified by a unique 3-digit ID. In the event that there needs to be a manual recount, this file would be accessed, decrypted, then used by an Administrator to do so.

Security & Other Operations

To activate the system, an administrator will have no more than 3 attempts to enter valid login credentials. If this is exceeded the program will automatically terminate.

At anytime in which the system is activated and running, an Administrator can login in through the voter login interface to access the various system options. When this happens the system is "paused" until the administrator reactivates the system. At no time is the administrative option window open simultaneously with the VoterForm window. This is a security measure prevent unauthorized access to administrative options.

After a voters ballot has been cast that voter will not be allowed to vote again. This is tracked in the database by 1's & 0's in the "hasVoted" column in the voters table. '0' indicates a voter has not voted.

Recount & Vote Verification
Vote verification can be conducted in two ways, by a manual recount using the LiveResults file or via administrative options (Recount Ballots). This method verifies the vote count by summing the number of casted ballots, then comparing it to the sum of registered voters who have voted. If the results are not equal the system will report that the votes are not certified.