

Chapter 1

INTRODUCTION

Project Context

Nowadays, the development and widespread use of information technology is changing the way voters participate into any type of voting process of the school and ultimately, the way they vote. Despite all these mobile voting and internet voting are on peak of popularity, e-voting system through web-based can lead to overcome from the voting station scarceness particularly in Gov. Alfonso D. Tan College (GADTC). In terms of providing convenience, it is an easier way to respond and select nominees to any type of voting process that may result high voting ratio.

According to Adams (2014) ones right to vote is very essential especially our voting principle. In fact, in the study of Dodson (2017), she traced the hard work of activist-women and men, of many races-that led them to gain the right to vote. On the other hand, Sedky & Hamed (2015), stated that election results may be modified when delivered to the Higher Elections Committee, unauthorized voter may vote instead of the eligible voter, a vote may not be calculated; also the voter has to ensure that nobody has the possibility to know his ballot data.

E-voting system will generate an accurate counting of votes to be able to know the result of winning candidate after voting process and ultimately, will have better satisfaction and convenience to admins and users. This web-based voting system use basic operations system that is a great deal to develop. Electronic digital signature device will use to take signature, fingerprint scanner will use to verify fingerprint and web-camera to take photo of the voters and the running candidates and voters.

The proponents conduct this study in order to give convenience to students and to improve traditional paper-based voting system. It can replace the inefficient and most importantly error prone human component. It can save the time of the voter and avoid the forgery of votes. This system also helps the college to easily conduct any type of election and most importantly to reduce manpower.

Purpose and Description

E-Voting System of Gov. Alfonso D. Tan College offers multiple advantages over traditional paper-based voting system that increase student access and encourage participation. Paper-based voting system used in GADTC today that is intended for the rendition of trusted election process that can also be delivered through web-based voting system like E-Voting System of GADTC.

Its common purpose is to avoid mistakes made by manual process. This has an assurance in providing better satisfaction in terms of data privacy, accuracy, reusability, eligibility and integrity.

Objectives of the study

This study aims to change voting process in Gov. Alfonso D. Tan College through web-based voting system. Specifically, it aims the following:

- To cater any type of election in GADTC
- To provide better satisfaction and convenience of using the system in voting
- To give faster and accurate tallying of votes
- To reduce human effort
- To announce the result within a short period of time

Scope and Limitation

The development of E-voting Voting System through web-based covers variety of scope. One is the registration of candidates and voters that will include digitalization on photos, fingerprint and signature. The viewing of results and ranks can be done on the other hand can be previewed using the system. It facilitates quick and accurate electronic counting of votes as well. It can include organizational election like SRO,SSB and SBO, student choice award and FAST Club elections or any type of elections in Gov. Alfonso D. Tan College, thus it is very capable to generate and print result of the election.

Meanwhile, E-Voting System has its limitations. In an organizational election, it will only accept votes from the registered voters. The system will accept right thumb only in fingerprint registration. In addition, it limits also the voters to gain numerous votes to the candidates. And lastly voters can participate voting process inside the college premises only.

Transactions not mentioned above are beyond the scope of this study.

Chapter 2

REVIEW OF RELATED LITERATURE AND RELATED STUDIES

This chapter presents the different researchers and related studies necessary in providing information through understanding the concept and function of this study.

Related Literature

The study of Divya (2014), entitled Advanced Online Voting system stated that the absentee voting refers to those registered voters who cannot make it to the polls on election day because of occupation, business, studies, travel, imprisonment (other than a convicted felon), illness, disability and hospitalization or resident in a long term care facility, and thus, may vote through absentee ballot. This voting can be committed through tele-voting which extends to the voting via a mobile cell phone so this also facilitates the absentee voting.

Meanwhile MohammedSidqi (2014) is a Master in Computer Science in Sulaimani Polytechnic University in Iraq Country states that there are more than one billion GSM users worldwide that represent a large user potential, not just for mobile telephony, but also for other mobile applications that exploit the mature GSM infrastructure. In this paper, an electronic voting scheme using GSM mobile technology is presented. By integrating an electronic voting scheme with the GSM infrastructure, we are able to exploit existing GSM authentication mechanisms and provide enhanced voter authentication and mobility while maintaining voter privacy.

In addition, Abdooz (2013) Mobile Technology is most important in the world and current generation. We can get updated information using mobile devices. Such device can be used for personal and business transactions. Moreover, mobile devices functionality has

evolved tremendously. Nowadays, it can be used for a variety of purposes like to locate places much like a GPS, to read bar codes, to play games, to browse the internet, to know weather reports, to vote, and so much more. Significantly the most important uses are the voice calls, video calls and Short Messaging System (SMS). Currently various models of mobile phones have come out in the market and each one having special advantages over the others. The SMS or message facility is important in the mobile device. It allows the sender to send message from one mobile to another mobile device. Most business establishments advertise through print media like the newspapers, magazines, pamphlets, brochures, and so much more. They also make use of the radio to promote their product and services. Furthermore, a lot business establishments conduct market survey like consumer to choose which product key like best or selecting the best brands and so on.

Robinson (2011) developed Mobile Voting System (MVS) that will operate in parallel with the existing manual and automated voting processes. It will enable legitimate voters to cast their vote from wherever they please using their mobile devices unlike other means that require the voter to appear at the polling station. This will help alleviate the nuisance of long queues at poll-sites which waste a lot of time. It will also ease the vote-counting process which will be done instantly as the voting progresses and a graphical display will be available on a site (attached to the system) for all stakeholders to view progress, hence ensuring transparency.

Nadja (2009), Mobile voting falls under a category of voting called e-voting, which is short for electronic voting, and refers to the option of using electronic means to vote in referendums and elections. There are systems such as DRE (Direct Electronic Recording) voting machines that record the vote without that vote being transmitted over the Internet

or another network. The interface of a DRE machine can be a touch screen or a scanner that scans the ballot paper where the voter marked the vote. The vote is then registered and stored in the voting machine. Then there is voting over the Internet that uses a PC with an Internet-connection to cast the vote and send it to be stored in another remote computer. Personal Digital Assistants (PDA's), telephones or mobile phones can also be used to cast a vote electronically.

Related Studies

E-Voting system using Android is a Masters Research Project by Rajesh Vedala, Rimon Gargho, Wachiravute Lackchaloemporn (Axe) MS in Wireless Communications and Laith Salih Hadi MS in Computer Science of School of Engineering, Technology, and Media National University San Diego, California. The process voting in this system is simply of casting the vote using an android device.

Android-Based Smart Voting System Conducted by Tawseef Mahmood, Shamah Mahbub Zoha & Ashis Kumar Das of BRAC University. The Smart Voting System divides the total voting process into distinct parts. Initially, the super administrator is in possession of the voter database that has been prepared. He or she loads it onto the internal memory of the Android device after verifying that he or she is the 14 authorized super administrator. After loading of the database, local administrators have the ability to start the voting process. Once started, voters undergo a verification process. They place their finger on the designated sensor and the biometric impression is stored in the memory of Arduino temporarily. The biometric fingerprint templates that are stored in the Android device are then passed to the Arduino, which places them in the memory of the sensor. Finally, the live biometric impression and the stored fingerprints are matched to verify the existence of

the fingerprint record, before it is cross-referenced against the database to make the final call.

Proxy Voting Scheme proposed by Zwattendorfer that allows the voter to delegate his or her voting power to a proxy who actually casts the ballots for all represented voters. Norway has used an Internet-based voting protocol for some years, and the vote privacy and correctness of this scheme have been demonstrated. Pan also proposed an electronic voting scheme that is based on the ring signature and is resistant to a clash attack. Several schemes with delegated voting functionality have been proposed.

Psu Bayambang Campus Voting System is in this light that the main concern of this software is the development of a College Student Council (CSC) Voting System that will enhance or upgrade the current one in use. This proposed system will be in accordance with the specifications required by the College Student Council Electoral Board. This software was primarily designed as a user friendly material; however, it presented the target report in use – as a form and counting performing functions.

Chapter 3

DESIGN AND METHODOLOGY

This chapter presents the design and methodology of the proposed study, E-Voting System of Gov. Alfonso D. Tan College. It describes the flow of the system that corresponds to the process and the software specification of the system is discussed.

METHODOLOGY

This study involves the idea of other people as we interview the officers of different student organizations, the president of FAST Club and the Prefect of Student Activities to know the manual voting process that exists in the college. In addition, we prepared a standardized questionnaire for some students to know their perceptions in having an electronic voting system. The proponents found out that developing a dynamic voting system is important. Hence, E-voting System of Gov. Alfonso D. Tan College to manage any type of elections.

The researchers use the iterative waterfall model in developing the system. Iterative waterfall model can be thought of as incorporating the necessary changes to the classical waterfall model to make it usable in practical software development projects. It is almost same as the classical waterfall model except some changes are made to increase the efficiency of the software development. The iterative waterfall model provides feedback paths from every phase to its preceding phases, which is the main difference from the classical waterfall model. When errors are detected at some later phase, these feedback paths allow correcting errors committed by programmers during some phase. The feedback paths allow the phase to be reworked in which errors are committed and these changes are reflected in the later phases. But, there is no feedback path to the

stage – feasibility study, because once a project has been taken, does not give up the project easily.

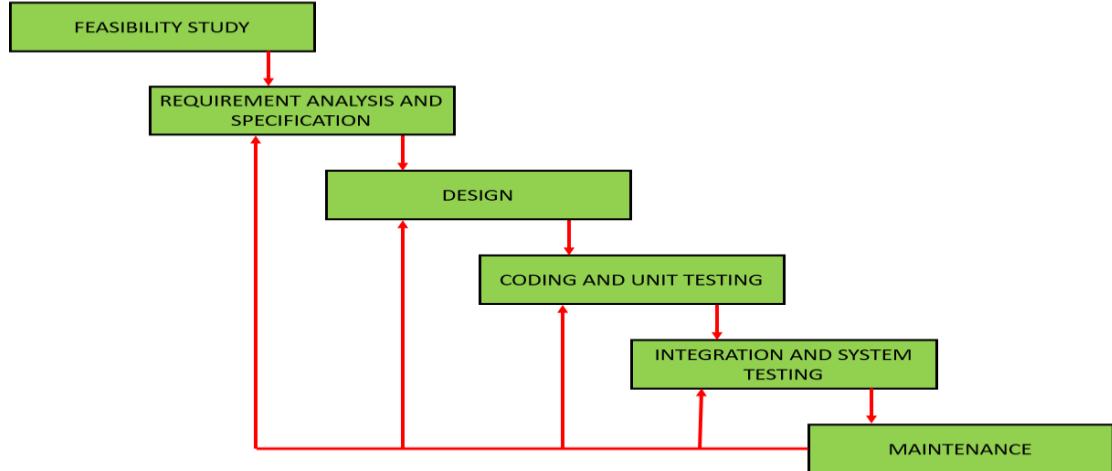
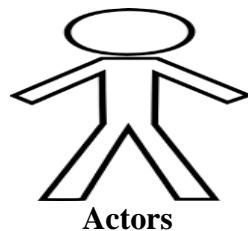


Figure 1: Iterative Waterfall Model

The unified modeling language and the data flow diagram is the method being used by the researchers. Unified Modeling Language, is a standardized modeling language consisting of an integrated set of diagrams, developed to help system and software developers for specifying, visualizing, constructing, and documenting the artifacts of software systems, as well as for business modeling and other non-software systems. A data flow diagram shows the way information flows through a process or system. It includes data inputs and outputs, data stores, and the various sub processes the data moves through. DFDs are built using standardized symbols and notation to describe various entities and their relationships. The researchers use these methods to visualize the interactions between user and the system and also to identify the flow of the data in the system.

Use Case



- Administrator
- Commissioner
- Voter

Figure 2: Actors

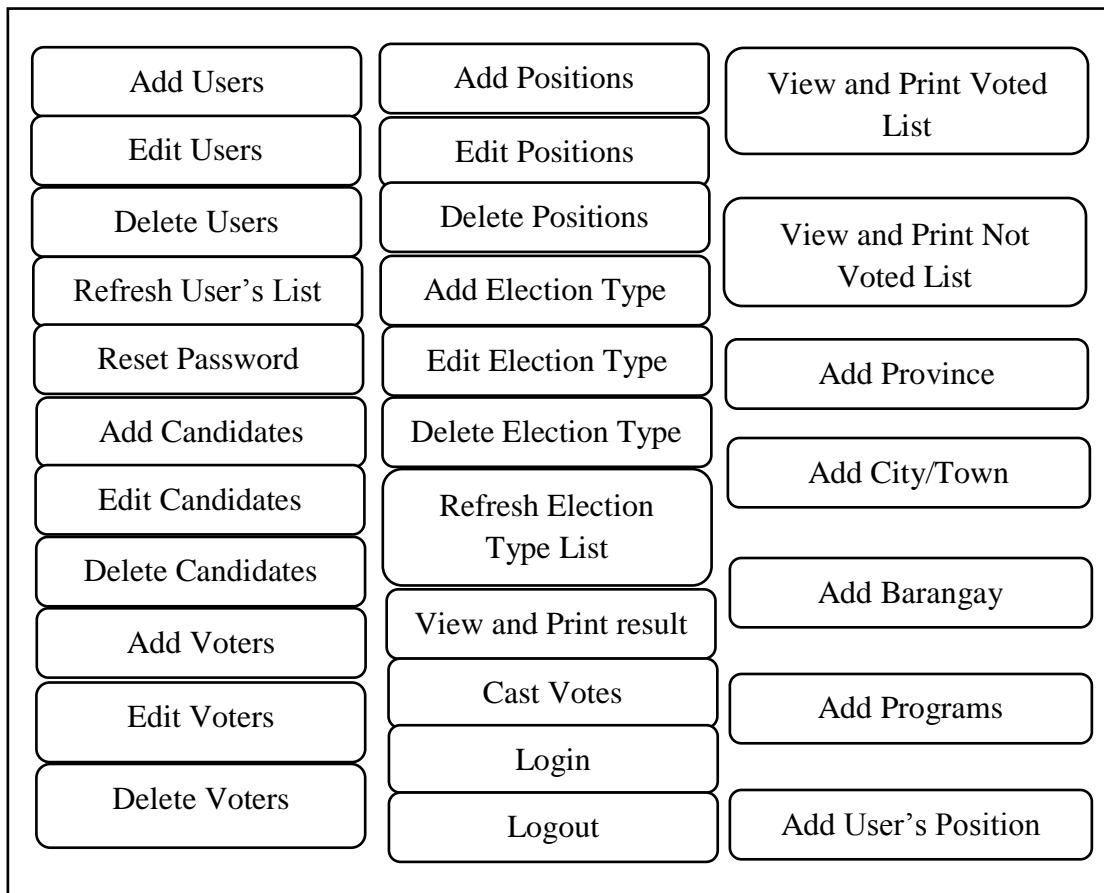


Figure 3: Use Cases

Use Case Diagram

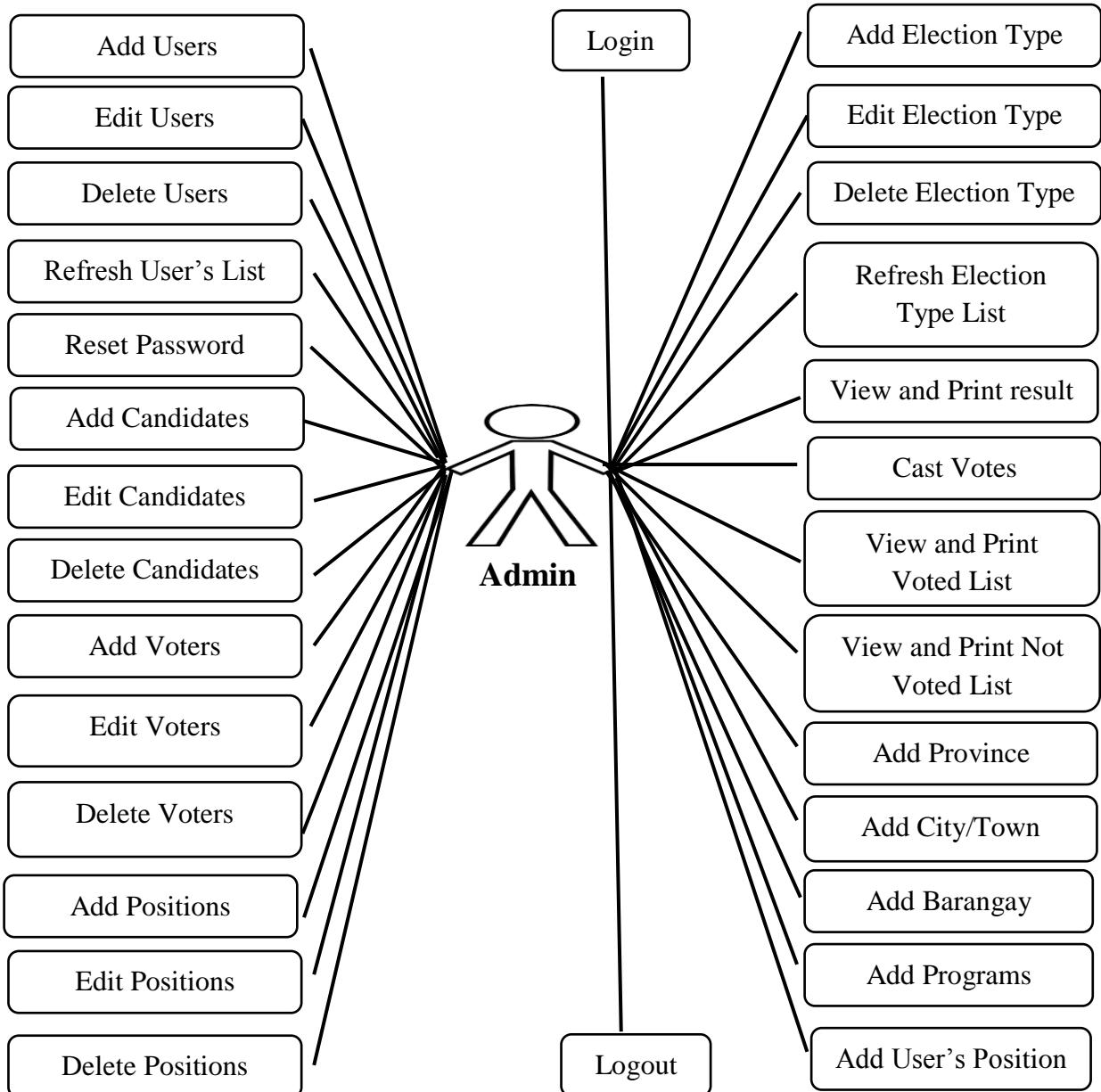


Figure 4: Use Case Diagram for Administrator

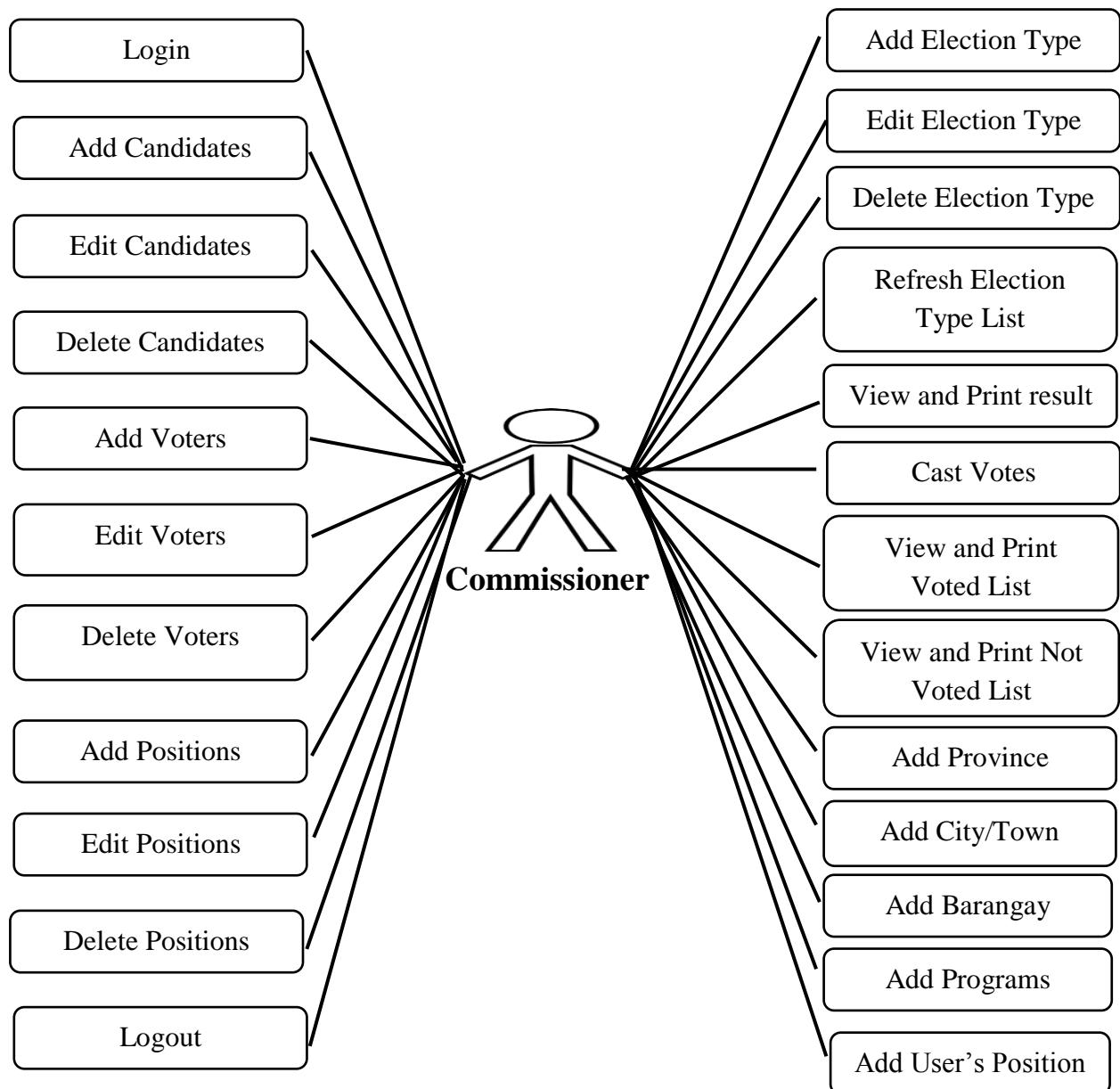


Figure 5: Use Case Diagram for Commissioner

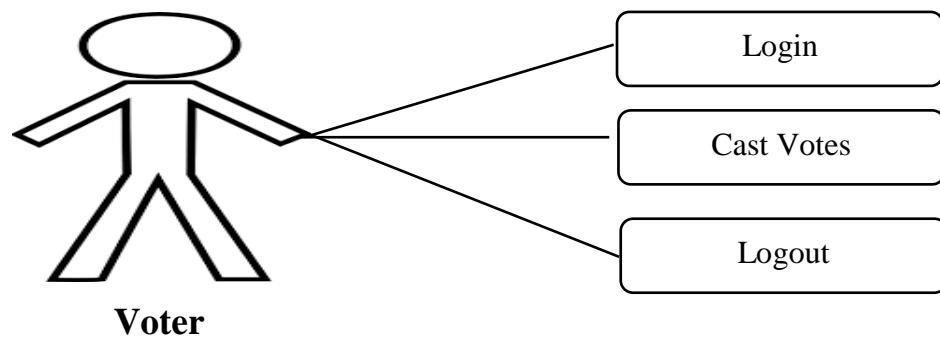


Figure 6: Use Case Diagram for Voter

Use Case Scenario

Table 1

Login

Use case name	Login
Actor	Administrator, Commissioner, Voter
Description	The Use Case is used by the actors to login to the system
Normal flow	1.The use case begins when the user opens the system. 2.Input Username & Password. 3.Click Login button. 4. The system Pops-up the main menu. 5.Use case instance terminates.
Precondition	Username and password inputted are correct.
Post condition	The actors have access to administrative functions in their domain.
Assumption	The actor is on the system

Table 2

Add User

Use case name	Add User
Actor	Administrator
Description	The Use Case is used by the actor to add users of the system.
Normal flow	1. The use case begins when the user clicks Account button. 2. Clicks Add button. 3. Input user's data. 4. Clicks Save button. 5. The system will save and will display the updated list of users. 6. Use case instance terminates.
Precondition	The actor must log in the system and information of the users to be registered must be available.
Post condition	The users is authorized to perform their respective task on the system
Assumption	The actor is on the system

Table 3
Modify User

Use case name	Modify User
Actor	Administrator
Description	The Use case is used by the actor to edit user's information to the system
Normal flow	<ol style="list-style-type: none"> 1. The use case begins when the actor clicks Account button. 2. Select user's information from user's list. 3. Click Modify button. 4. Input changes on user's information. 5. Clicks Save button. 6. The system will save the updated user's information and displays the updated list of users. 7. Use case instance terminates.
Precondition	The actor must logged in the system and User information must exist in the system.
Post condition	User's information is updated and authorized to perform task on the system
Assumption	The actor is on the system

Table 4
Refresh User List

Use case name	Refresh User List
Actor	Administrator
Description	The use case is used by the actor to refresh users list.
Normal flow	<ol style="list-style-type: none"> 1. The use case begins when the actor clicks Account button. 2. Clicks Refresh button. 3. The system will refresh users list. 4. Use case instance terminates.
Precondition	The actor must logged in the system.
Post-condition	User list is updated.
Assumption	The Actor is on the system.

Table 5

Reset Password

Use case name	Reset Password
Actor	Administrator
Description	The use case is used by the actor to reset password of a user.
Normal flow	<ol style="list-style-type: none"> 1. The use case begins when the actor clicks Account button. 2. The Actor will select User's account in the User's List. 3. Click Reset Password button. 4. Input New Password and click Save button. 5. The system will update the user's password. 6. The system will close if the admin reset his/her own password and admin will login again using the new password. 7. Use case instance terminates.
Precondition	User information must exist in the system
Post-condition	Actor can reset users password in the system.
Assumption	The Actor is on the system.

Table 6

Add Candidate

Use case name	Add Candidate
Actor	Administrator, Commissioner
Description	The Use case is used by the actors to add candidates to the system.
Normal flow	<ol style="list-style-type: none"> 1. The use case begins when the users click Candidates button. 2. Select New Candidates button. 3. Input Candidate's Data. 4. Click Save button. 5. The system will save and display the candidate's information. 6. Use case instance terminates.
Precondition	The actors is already logged in the system and the information of the candidate to be registered must be available.
Post condition	The candidates is now included to the candidates list to the system.
Assumption	Actor is on the system

Table 7

Edit Candidate

Use case name	Edit Candidate
Actor	Administrator, Commissioner
Description	The Use Case is be used by the actor to edit candidate's information to the system.
Normal flow	<ol style="list-style-type: none"> 1. The use case begins when the actors click Candidates button. 2. Select candidate's information. 3. Click Modify Candidate button. 4. Input changes to Candidate's Data. 5. Click Save button. 6. The system will save and display the updated candidate's information. 7. Use case instance terminates.
Precondition	The actor is already logged in the system and the information of the candidate to be registered is available.
Post condition	The candidates is now updated and included to the candidates list to the system
Assumption	Actor is on the system

Table 8

Delete candidate

Use case name	Delete Candidate
Actor	Administrator, Commissioner
Description	The Use Case is used by the actor to delete candidate's information from the system.
Normal flow	<ol style="list-style-type: none"> 1. The use case begins when the actors click Candidates button. 2. Select candidate's information. 3. Click Delete Candidate button. 4. The system will display a dialog box and will ask the user to delete permanently the candidate's information. 5. The user will select "Yes". 6. The system will remove the candidate's information from the list. 7. The system will display the updated list of candidates. 6. Use case instance terminates.

Precondition	The actors is already logged in the system and the information of the candidate to be register is available.
Post Condition	The candidate's information removed from the candidates list.
Assumption	The Actors is on the system.

Table 9

Add Voter

Use case name	Add Voter
Actor	Administrator, Commissioner
Description	The Use Case is used by the actors to add voters to the system.
Normal flow	<ol style="list-style-type: none"> 1. The use case begins when the actors click Unofficial Registered Voters button. 2. Click Add button. 3. Input voter's information including fingerprint, photo, and signature. 4. Click Save button. 5. The system will save and display the voter's information. 6. Use case instance terminates.
Precondition	The actors is already logged in the system and the information of the voter to be registered is available.
Post condition	The voter is now included to the unofficial registered voters list in the system
Assumption	Actor is on the system

Table 10

Edit Voter

Use case name	Edit Voter
Actor	Administrator, Commissioner
Description	The Use Case is used by the actor to Edit voter's information on the system.
Normal flow	<ol style="list-style-type: none"> 1. The use case begins when the actors click Unofficial Registered Voters button. 2. Select voter's information from the voter's list. 3. Click Edit button. 4. Input necessary changes for Voter's information. 5. Click save button.

	<p>6. The system will save and display the updated voter's information.</p> <p>7. Use case instance terminates.</p>
Precondition	The actors is already logged in the system and the information of the voter to be register is available.
Post Condition	The voter's information is now updated and included to the voter's list in the system.
Assumption	Actor is on the system.

Table 11

Delete Voter

Use case name	Delete Voter
Actor	Administrator, Commissioner
Description	The Use Case is used by the actor to delete voter in the system.
Normal flow	<p>1. The use case begins when the actors click Unofficial Registered Voters button.</p> <p>2. Select voter's information from the voter's list.</p> <p>3. Click Delete button.</p> <p>4. The system will display a dialog box and will ask the user to delete permanently the voter's information.</p> <p>5. The user will select "Yes".</p> <p>6. The system will remove the voter's information from the list.</p> <p>7. The system will display the updated list of voters.</p> <p>8. Use case instance terminates.</p>
Precondition	The actors is already logged in the system .
Post condition	The voter is now deleted to the unofficial registered voters list to the system
Assumption	Actor is on the system

Table 12

Add position

Use case name	Add Position
Actor	Administrator, Commissioner
Description	The Use Case is used by the actor to add position of the candidates to the system.
Normal flow	<p>1. The use case begins when the actors click Setup Positions button.</p> <p>2.Click New button then Select an Election Type</p> <p>3.Input position and other information.</p> <p>4. Click Save button.</p> <p>5. The system will save and display the newly added position.</p> <p>6. Use case instance terminates.</p>
Precondition	The actors is already logged in the system .
Post condition	The new positions is now created and is available in the system.
Assumption	Actor is on the system

Table 13

Edit position

Use case name	Edit Position
Actor	Administrator, Commissioner
Description	The Use Case is used by the actor to Edit position to the system.
Normal flow	<p>1.The use case begins when the actors click Setup Position button.</p> <p>2.Select position's information.</p> <p>3.Click Update button.</p> <p>5.Input changes.</p> <p>6. Click save button.</p> <p>7. The system will save changes and display the list of candidate's positions.</p> <p>8.Use case instance terminates.</p>
Precondition	The actors is already logged in the system .
Post condition	The positions is now created, updated and is available to the system.
Assumption	Actor is on the system

Table 14

Delete position

Use case name	Delete Position
Actor	Administrator, Commissioner
Description	The Use Case is used by the actor to delete position in the system.
Normal flow	<ol style="list-style-type: none"> 1. The use case begins when the actors click Setup Positions button. 2. Select position's information. 3. Click Delete button. 4. The system will display a dialog box and will ask the user to delete permanently the position's information. 5. The user will select "Yes". 6. The system will remove the position's information from the list. 7. The system will display the updated list of position. 8. Use case instance terminates.
Precondition	The actors is already logged in the system .
Post condition	The positions is now deleted and not available to the system.
Assumption	Actor is on the system.

Table 15

Add Election Type

Use case name	Add Election type
Actor	Administrator, Commissioner
Description	The Use Case is used by the actor to Add Election Type to the system.
Normal flow	<ol style="list-style-type: none"> 1. The use case begins when the actors click Setup Voting button. 2. Click Add button. 3. Select Academic Year and input Election name and date. 4. Click Save button. 5. The system will display the result. 6. Use case instance terminates.
Precondition	The actors is already logged in the system .
Post condition	The new Election type is now created and now available to the system.

Assumption	Actor is on the system
------------	------------------------

Table 16

Edit Election Type

Use case name	Edit Election type
Actor	Administrator, Commissioner
Description	The Use is be used by the actor to edit Election Type to the system.
Normal flow	<ol style="list-style-type: none"> 1. The use case begins when the actors click Election Setup button. 2. Select election type information from Election Type List. 3. Click Edit button. 4. Input Election type information. 5. The system will display the result. 6. Use case instance terminates.
Precondition	The actors is already logged in the system .
Post condition	The new election type is now updated, created and now available to the system.
Assumption	Actor is on the system

Table 17

Delete Election Type

Use case name	Delete Election type
Actor	Administrator, Commissioner
Description	The Use is be used by the actor to Delete Election Type to the system.
Normal flow	<ol style="list-style-type: none"> 1. The use case begins when the actors click Election Setup button. 2. Select election type information from Election Type List. 3. Click Delete button. 4. The system will display a dialog box and will ask the user to delete permanently the election type information. 5. The user will select “Yes”. 6. The system will remove the election type information from the list.

	7. The system will display the updated list of election type. 8. Use case instance terminates.
Precondition	The actors is already logged in the system .
Post condition	The Election type is now deleted and now available to the system.
Assumption	Actor is on the system

Table 18

Refresh Election Type

Use case name	Refresh Election type
Actor	Administrator, Commissioner
Description	The Use is be used by the actor to refresh election type to the system.
Normal flow	<p>1. The use case begins when the actors click Election Setup button.</p> <p>2.Click Refresh button.</p> <p>3.The system will refresh Election Type List.</p> <p>4. The system will display the result.</p> <p>5. Use case instance terminates.</p>
Precondition	The actors is already logged in the system .
Post condition	The Election type is now refresh and now available to the system.
Assumption	Actor is on the system

Table 19

view and print result

Use case name	View and print result
Actor	Administrator, Commissioner
Description	The Use Case is used for the actor to view and print result of the election to the system.
Normal flow	<p>1. The use case begins when the actors click Result button.</p> <p>2. Select an Institute.</p> <p>3. Click Preview.</p> <p>4. The system will display and print the result</p> <p>5. Use case instance terminates.</p>

Precondition	The Actor is already logged in the system and the election result is available to the system.
Post condition	The result of an election is now available
Assumption	The actor is on the system

Table 20

Cast vote

Use case name	Cast vote
Actor	Voter
Description	The Use Case will be used for the voter to cast votes.
Normal flow	<ol style="list-style-type: none"> 1. The use case begins when the actors connect to a default network used in voting. 2. Visit default IP Address. 3. Vote for a candidates /select various voting options. 4. Click Submit button. 5. The system will display result and the voter successfully voted. 6. Use case instance terminates.
Precondition	The voters is already logged in the system and the voter's registration has been verified
Post condition	The voter's vote is recorded in the system and recorded that the voter has voted in the election
Assumption	The actor is on the system

Table 21

View and Print Voted List

Use case name	View and Print Voted List
Actor	Administrator, Commissioner
Description	The Use Case is used by the actors to view and print voted list in the system.
Normal flow	<ol style="list-style-type: none"> 1. The use case begins when the actors click Report Button. 2. Select View Voted. 3. Select Election Type. 4. Select Institute.

	5. Click Search button. 6. The system will display and print the result. 7. Use case instance terminates.
Precondition	The Actor is already logged in the system and the voted list is available in the system.
Post condition	The list of voters who voted is now available.
Assumption	The actor is on the system

Table 22

View and Print Not Voted List

Use case name	View and Print Not Voted List
Actor	Administrator, Commissioner
Description	The Use Case will be used by the actors to view and print not voted list in the system.
Normal flow	1. The use case begins when the actors click Report button. 2. Select View Not Voted. 3. Select Election Type. 4. Select Institute. 5. Click Search button. 6. The system will display and print the result. 7. Use case instance terminates.
Precondition	The Actor is already logged in the system and the not voted list is available in the system
Post condition	The list of voters who are not voted is now available
Assumption	The actor is on the system

Table 23

Add Province

Use case name	Add Province
Actor	Administrator, Commissioner
Description	The Use Case is used by the actors to add province to the system.
Normal flow	1. The use case begins when the actors click Administrative button.

	<p>2. Select Province.</p> <p>3. Right click and select Add.</p> <p>4. Input Province Name.</p> <p>5. Click SAVE button.</p> <p>6. The system will display the province list.</p> <p>7. Use case instance terminates.</p>
Precondition	The actor is already logged in the system and the Province Name is not exist in the system.
Post condition	The province name is created and exist in the province list.
Assumption	The actor is on the system

Table 24

Add City/Town

Use case name	Add City/Town
Actor	Administrator, Commissioner
Description	The Use Case is used by the actors to Add City/Town to the system.
Normal flow	<p>1. The use case begins when the actors click Administrative button.</p> <p>2. Select City/Town</p> <p>3. Right click and select Add City/Town.</p> <p>4. Select a province name where the City/Town belong.</p> <p>5. Input City/Town name and zip code.</p> <p>6. Click Save button.</p> <p>7. The system will display the City/Town list.</p> <p>8. Use case instance terminates.</p>
Precondition	The actor is already logged in the system and the City/Town name is not exist in the system.
Post condition	The City/Town name is created and exist in the city/town list.
Assumption	The actor is on the system

Table 25

Add Barangay

Use case name	Add Barangay
Actor	Administrator, Commissioner
Description	The Use Case is used by the actors to Add Barangay to the system.
Normal flow	<ol style="list-style-type: none"> 1. The use case begins when the actors click Administrative button. 2. Select Barangay. 3. Right click and select Add. 4. Select a province and city/town name where the barangay belongs. 5. Click SAVE button. 6. The system will display the barangay list. 7. Use case instance terminates.
Precondition	The actor is already logged in the system and the barangay name is not exist in the system.
Post condition	The barangay name is created and exist in the barangay list.
Assumption	The actor is on the system

Table 26

Add Institute

Use case name	Add Institute
Actor	Administrator, Commissioner
Description	The Use Case is used by the actors to Add Institute to the system.
Normal flow	<ol style="list-style-type: none"> 1. The use case begins when the actors click Administrative button. 2. Select Institute 3. Click New button 4. Input Institute name and description. 5. Click Save button 6. The system will display Institute list. 7. Use case instance terminates.
Precondition	The actor is already logged in the system and the Institute is not exist in the system.

Post condition	The Institute name is created and exist in the Institute list.
Assumption	The actor is on the system

Table 27

Add Program

Use case name	Add Program
Actor	Administrator, Commissioner
Description	The Use Case is used by the actors to Add program to the system.
Normal flow	<ol style="list-style-type: none"> 1. The use case begins when the actors click Administrative button. 2. Select Programs. 3. Right click and select Add 4. Select institute and input program information. 5. Click Save button 6. The system will display program list. 7. Use case instance terminates.
Precondition	The actor is already logged in the system and the program is not exist in the system.
Post condition	The program name is created and exist in the program list.
Assumption	The actor is on the system

Table 28

Add User's Position

Use case name	Add User's Position
Actor	Administrator, Commissioner
Description	The Use Case is used by the actors to Add User's Position to the system.
Normal flow	<ol style="list-style-type: none"> 1. The use case begins when the actors click Administrative button. 2. Select Position 3. Click Add button 4. Input Position name 5. Click Save button

	6. The system will display user's position list. 7. Use case instance terminates.
Precondition	The actor is already logged in the system and the User's Position is not exist in the system.
Post condition	The user's position name is created and exist in the user's position list.
Assumption	The actor is on the system

Data Flow Diagram (Existing)

Election Process (Fast Club)

Narrative Flow: The Comelec Chairman will post the announcement that the filing of candidacy will start so that the running candidates may fill up the candidacy form then the candidates may undergo screening if they are qualified. The comelec chairman will post again the qualified candidates so they may start campaign to its entire member then the voting will start which is the same process with the voting process of the student election of GADTC.

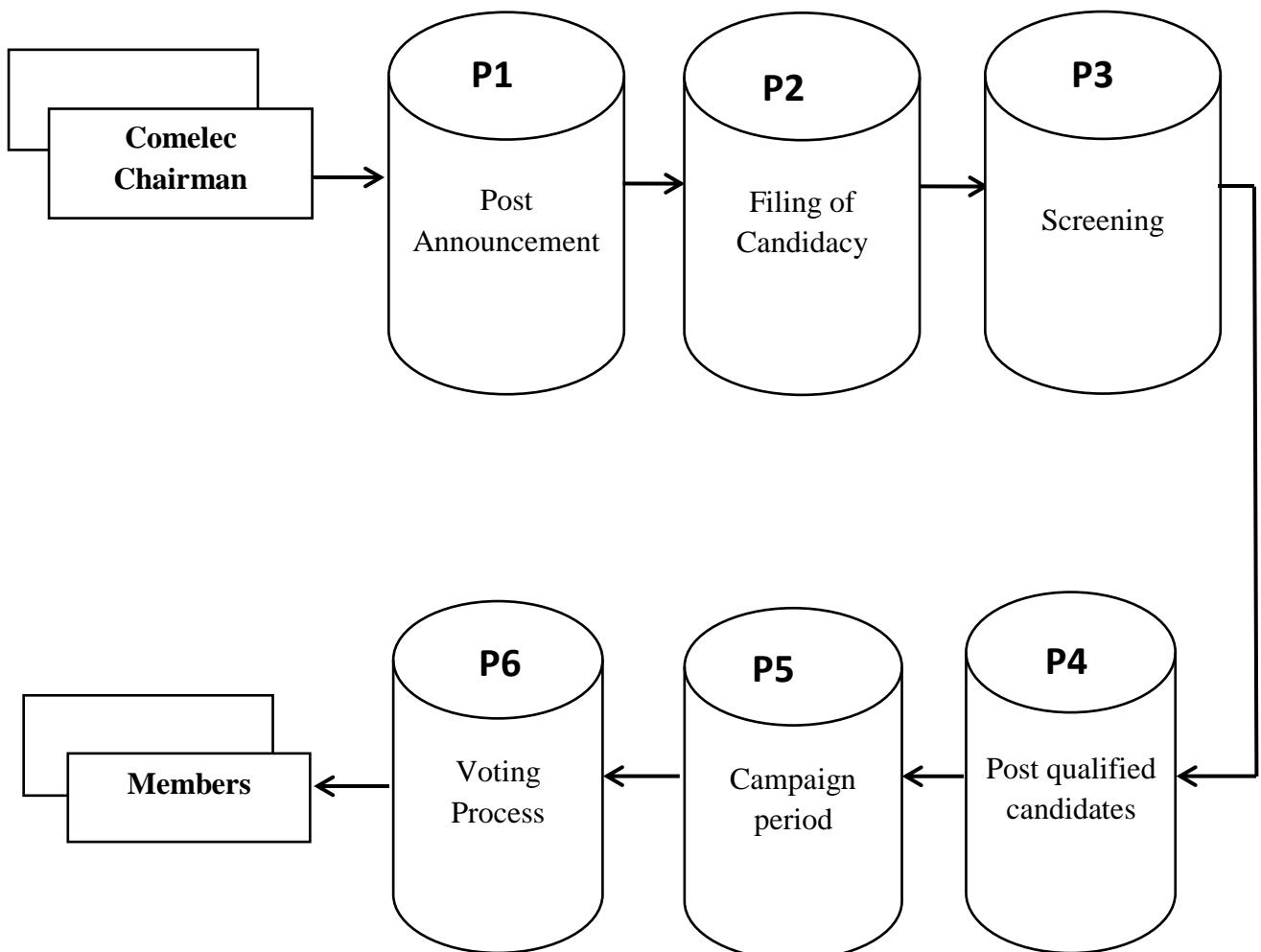


Figure 7: Election Process of FAST Club (Existing)

Selecting BEI

Narrative Flow: The CEO will select students as BEI then ask for the approval of the institute Dean. And then submit the list of students BEI to the prefect of the student activities.

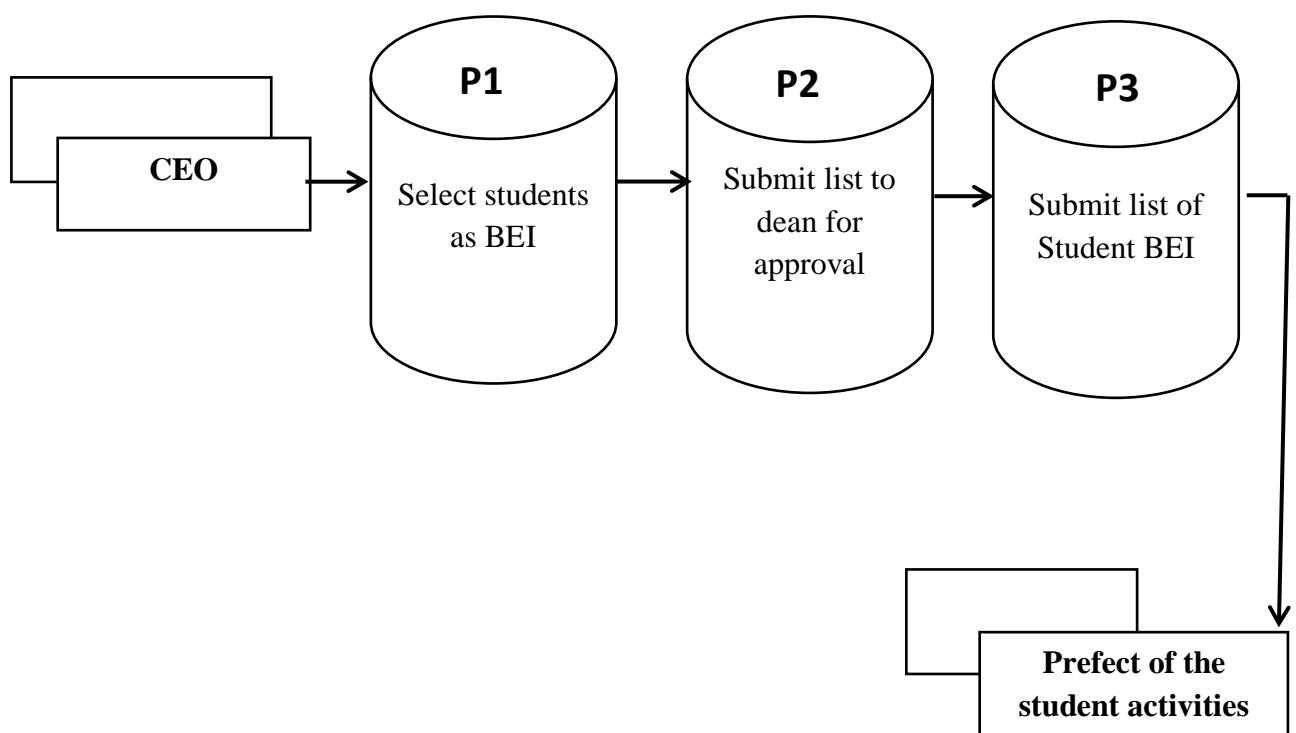


Figure 8: Selecting BEI (Existing)

Filing of Candidates

Narrative Flow: The Comelec Chairman will give a candidacy form to the running Candidates. The Candidates then will fill up the candidacy form and return it to the Comelec Chairman.

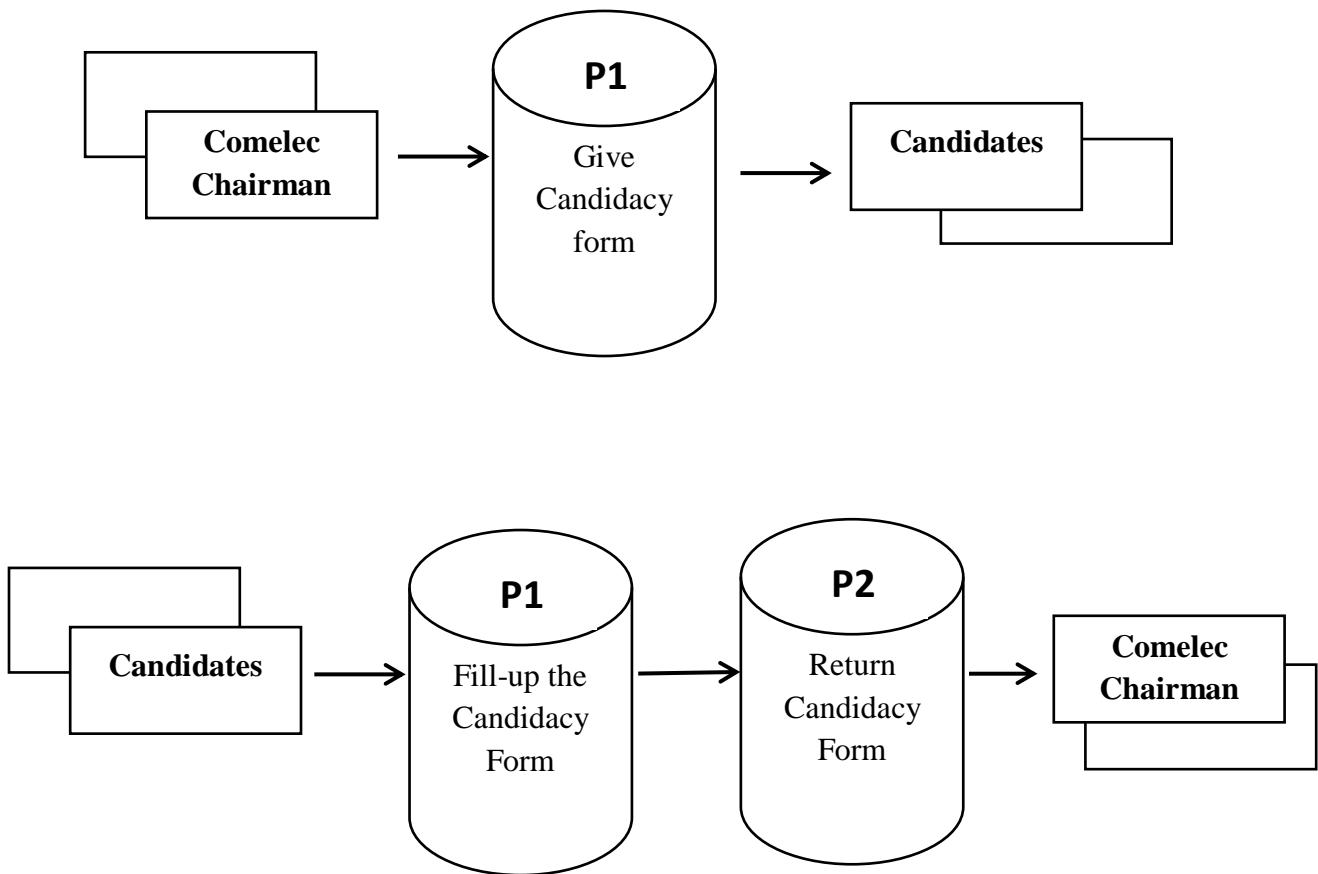


Figure 9: Filing of Candidates (Existing)

Assigning of Room

Narrative Flow: The Comelec Chairman will assign Room for each voter and give it to the BEI. The BEI then will post the room assignment per academic program to every classroom.

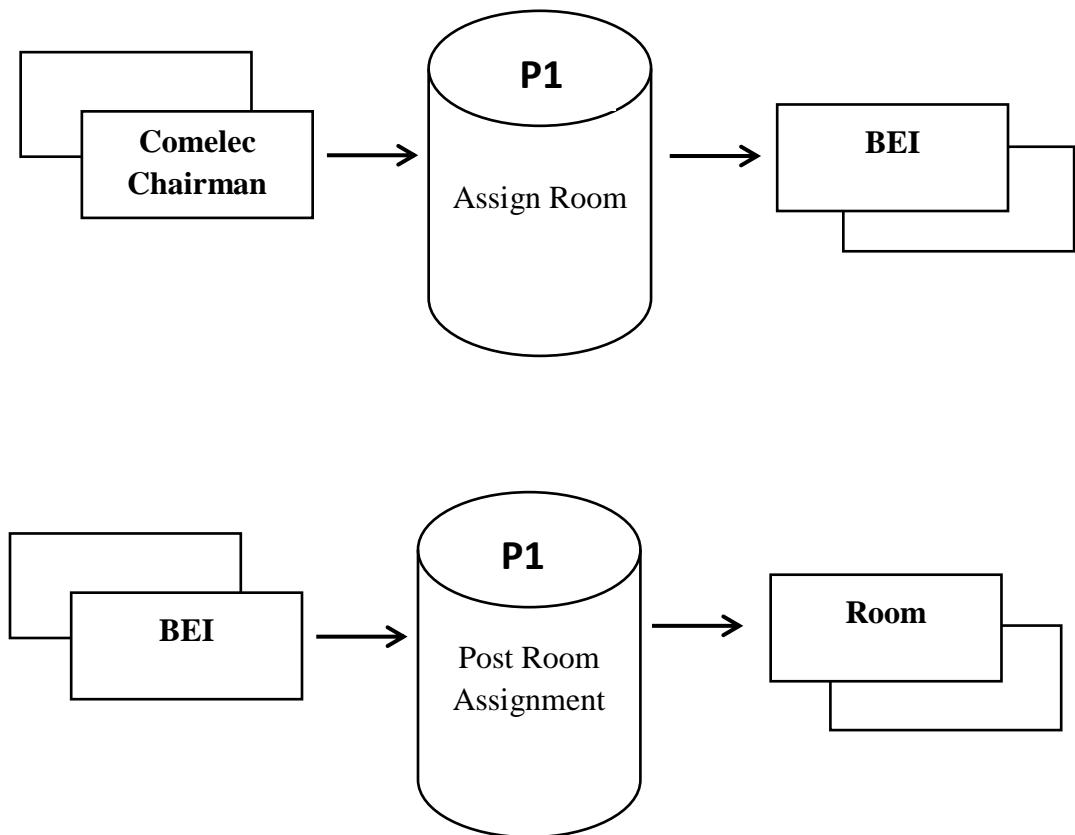


Figure 10: Assigning of rooms (Existing)

Voting Process

Narrative Flow: Voters will look for their assigned room where they will be given a Ballot paper from the BEI. The voters then can now start to vote and then return the ballot paper to BEI.

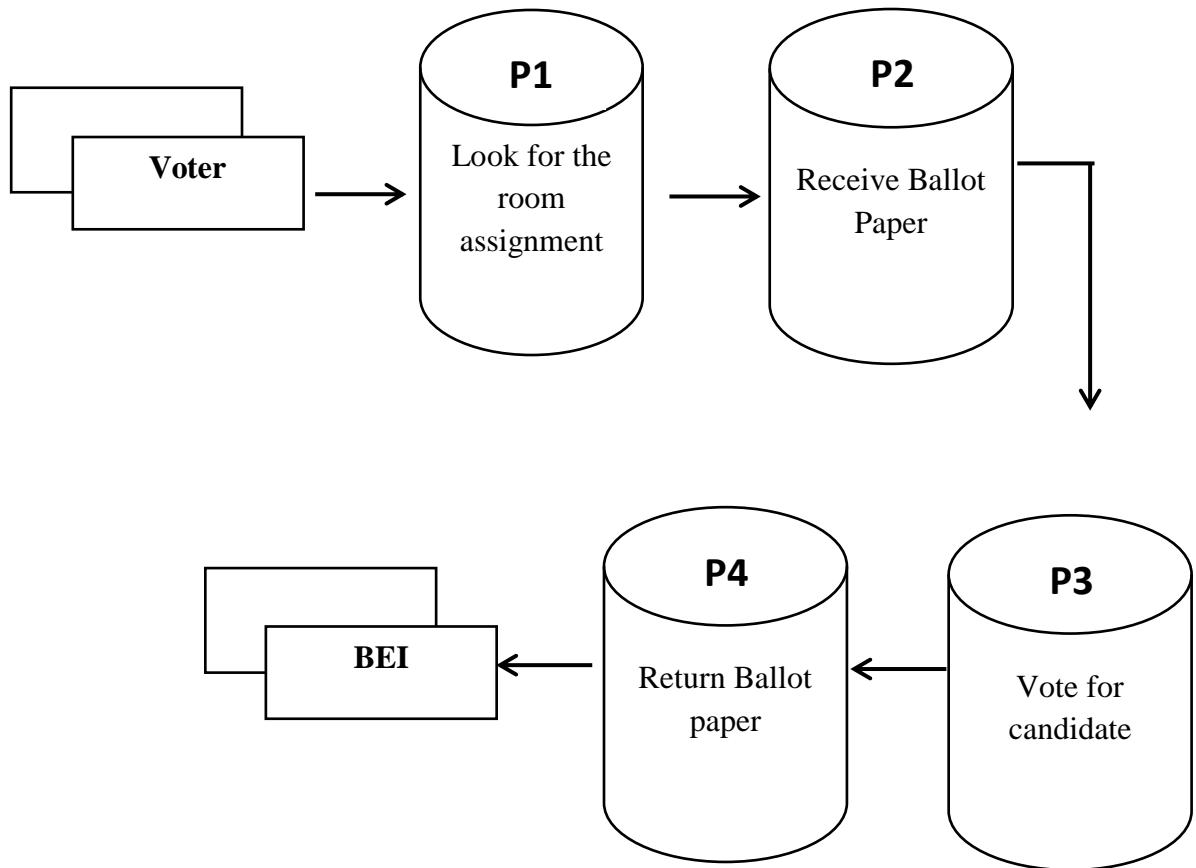


Figure 11: Voting Process (Existing)

Tallying and Counting of Votes

Narrative Flow: The BEI will tally votes and then count the total number of votes and write final result on a piece of paper. They will prepare then the election return and give then the general report to the Comelec Chairman.

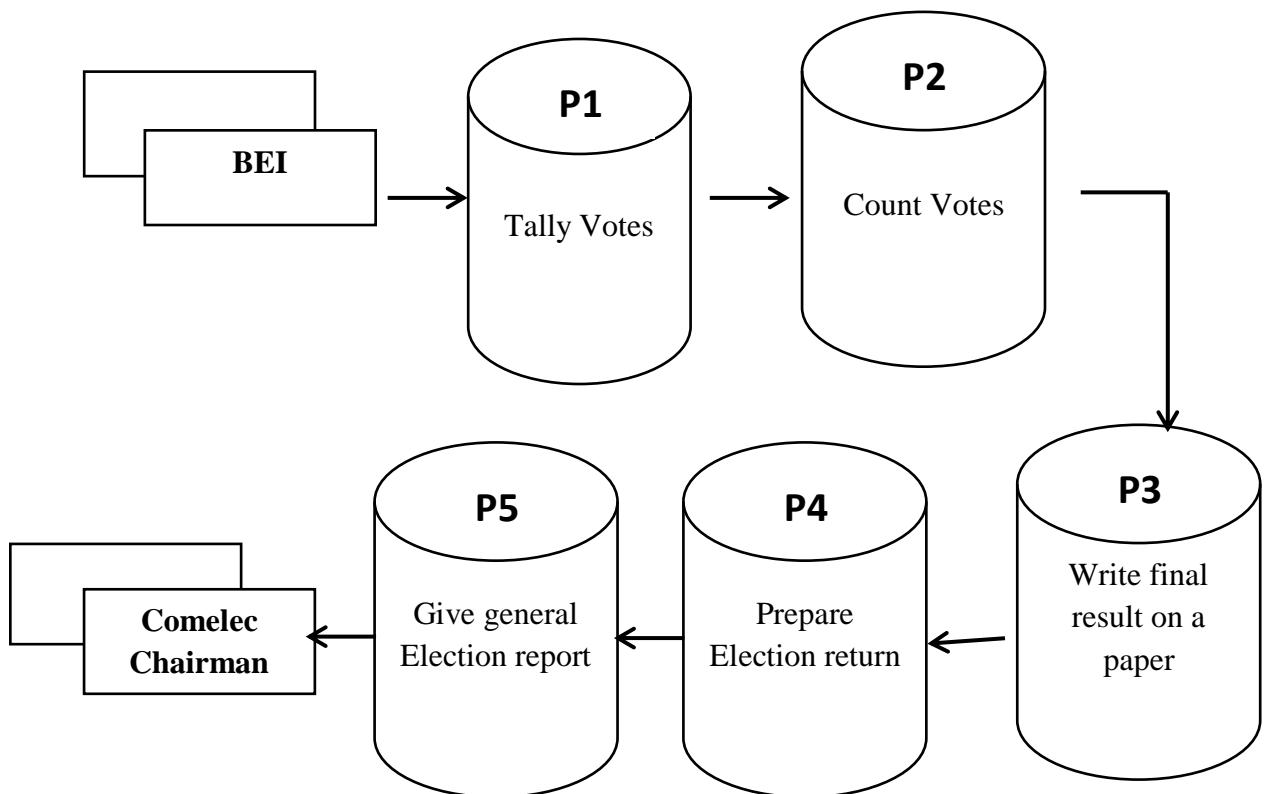


Figure 12: Tallying and counting of votes (Existing)

Data Flow Diagram (Proposed)

Setup Province

Narrative Flow: To manage Province information, choose right click and select Add and may perform modify, refresh and delete. To add, input Province name and save; To modify, input changes and click update.

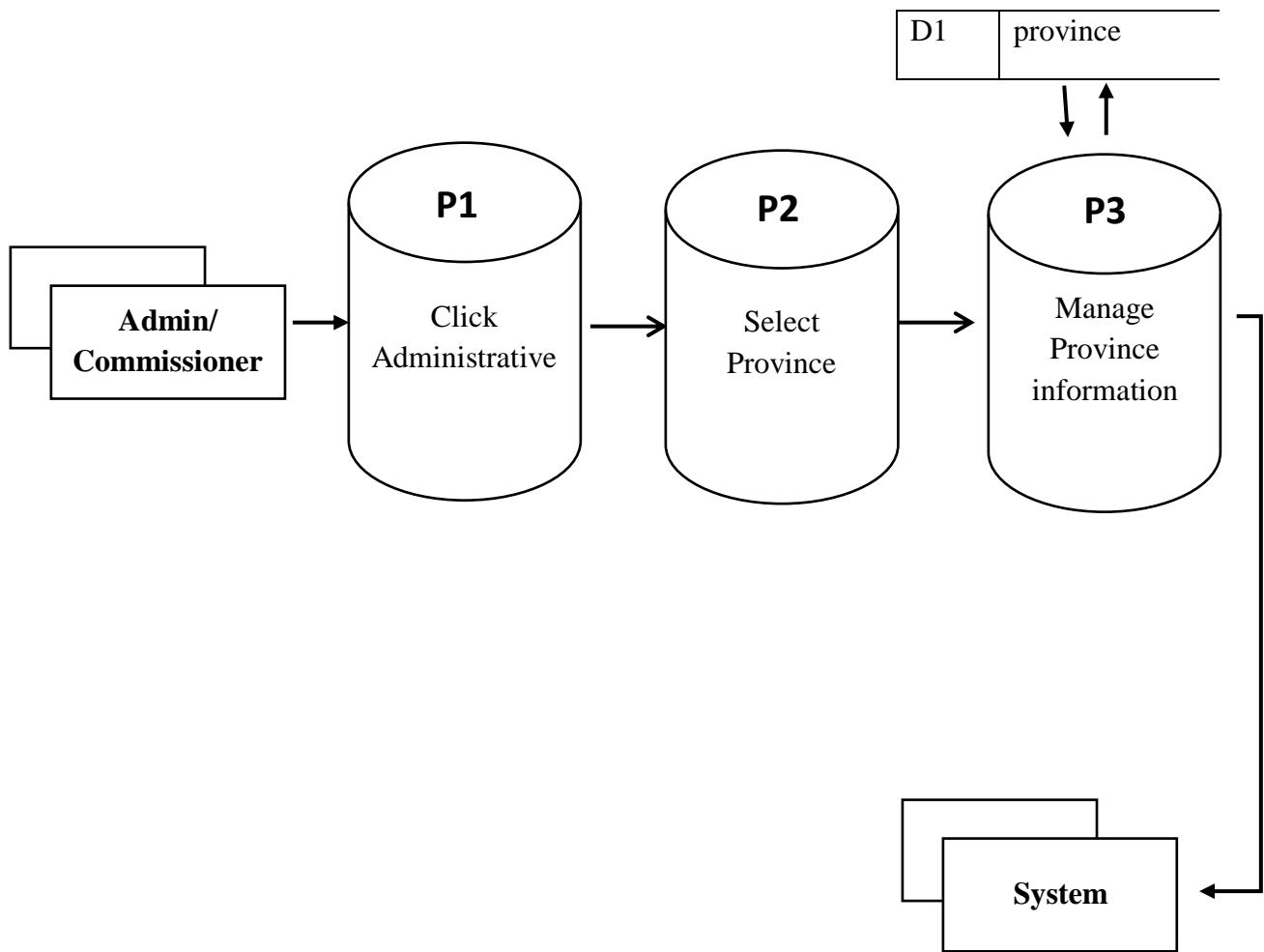


Figure 13: Setup Province (Proposed)

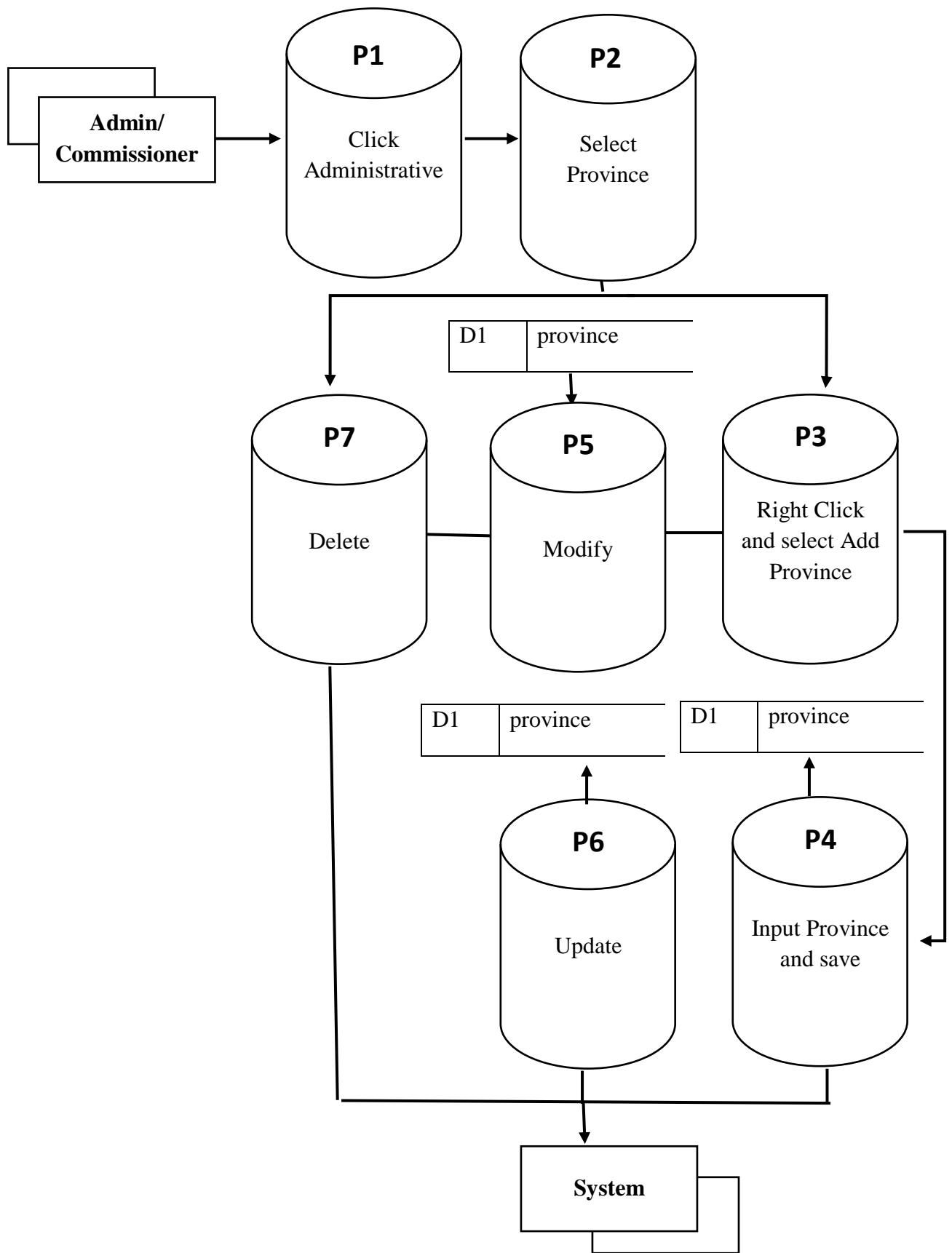


Figure 13: Setup Province (Proposed)

Setup City/Town

Narrative Flow: To manage City/Town information, choose right click and select Add and may perform modify, refresh and delete. To add, input City/Town name and save; To modify, input changes and click update.

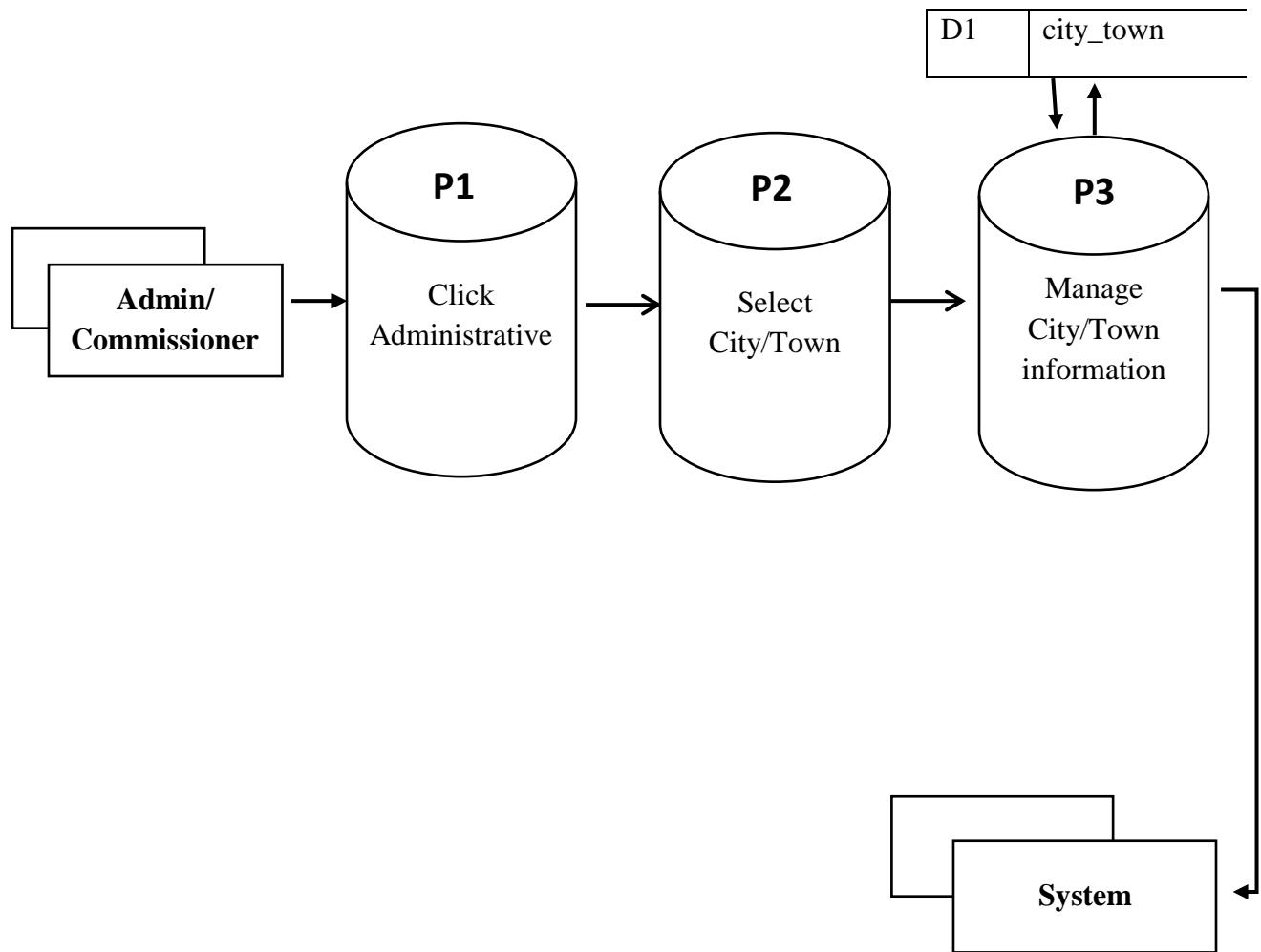


Figure 14: Setup City/Town (Proposed)

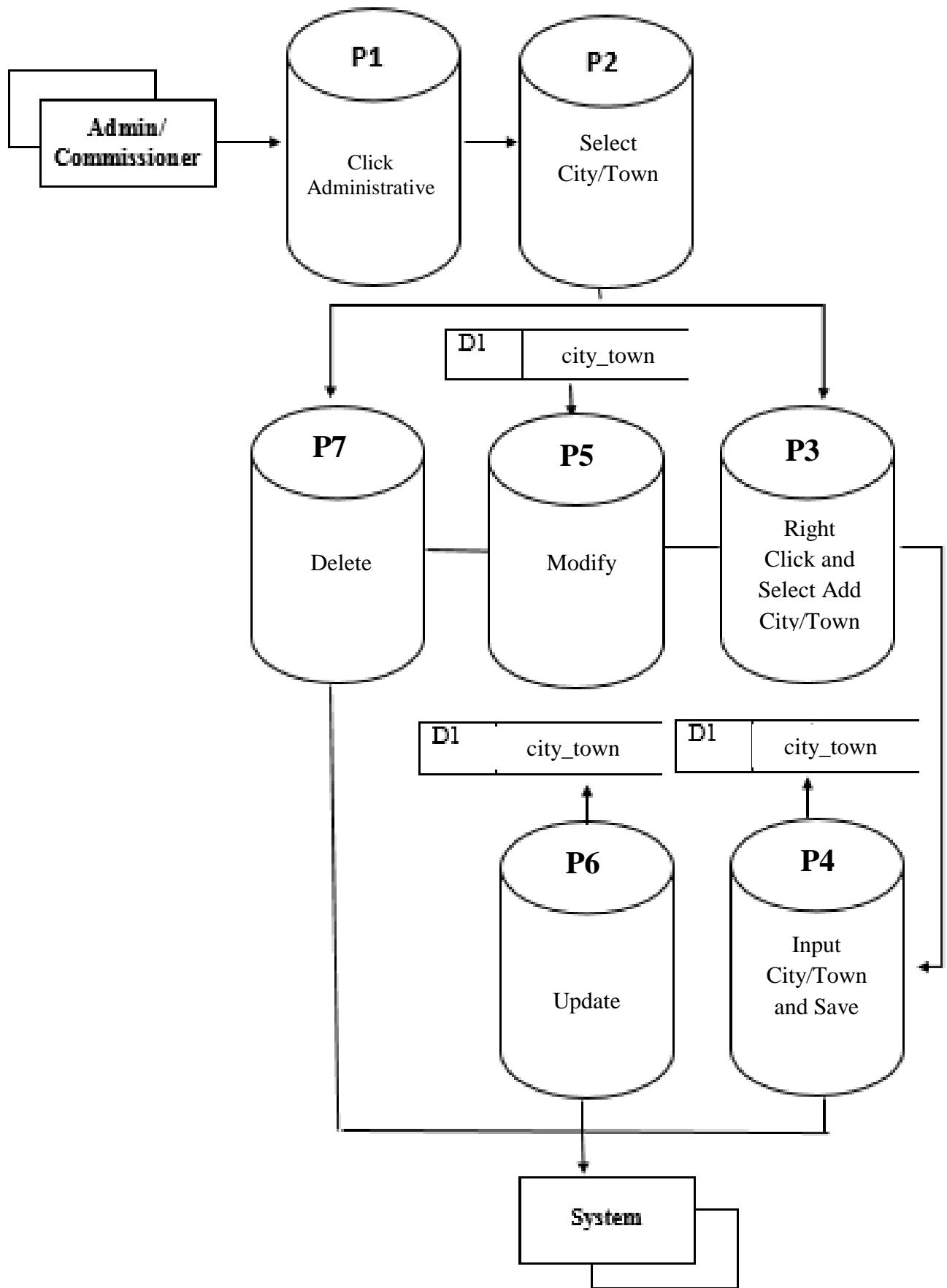


Figure 14: Setup City/Town (Proposed)

Setup Barangay

Narrative Flow: To manage Barangay information, choose right click and select Add and may perform modify, refresh and delete. To add, input Baranagy name and save; To modify, input changes and click update.

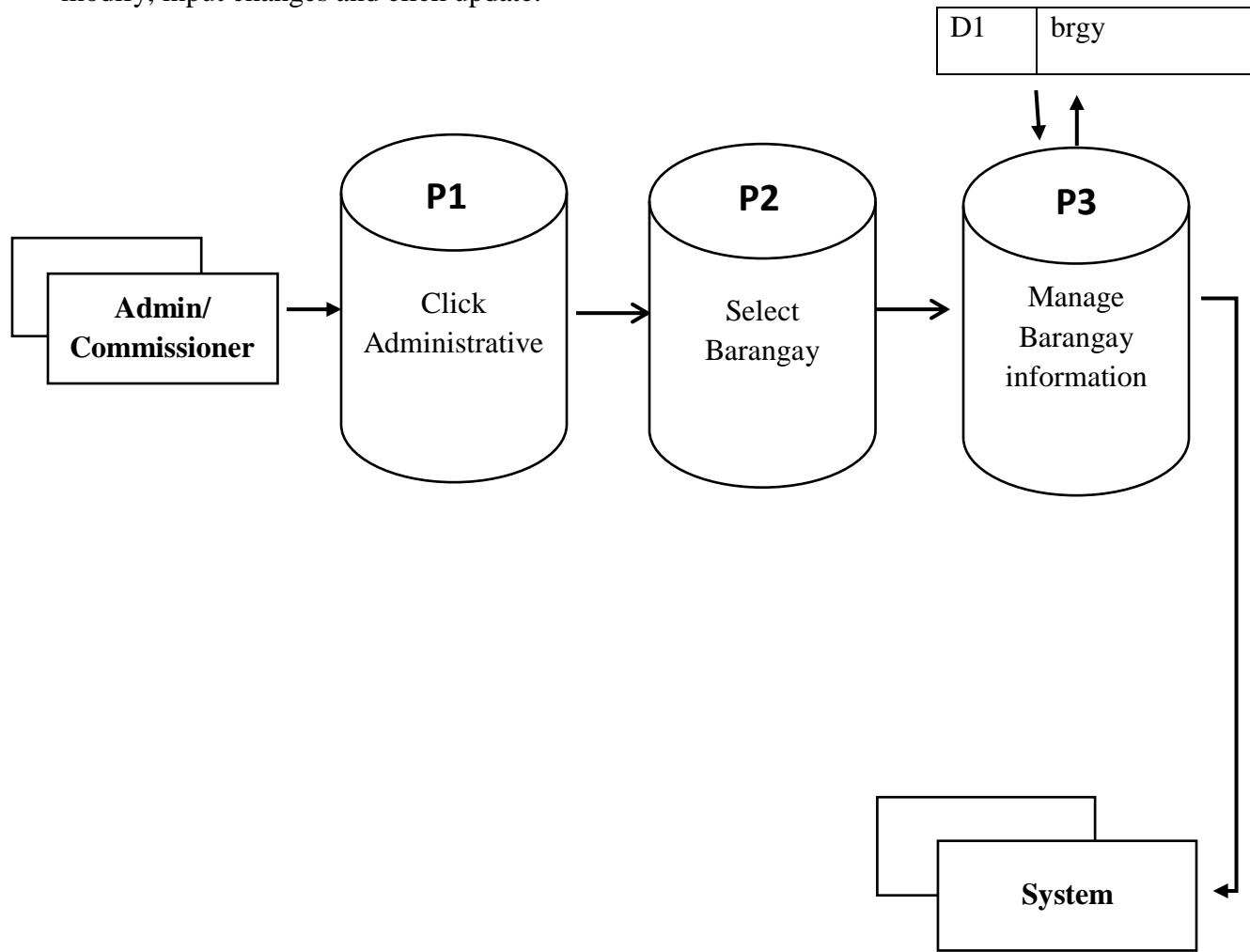


Figure 15: Setup Barangay (Proposed)

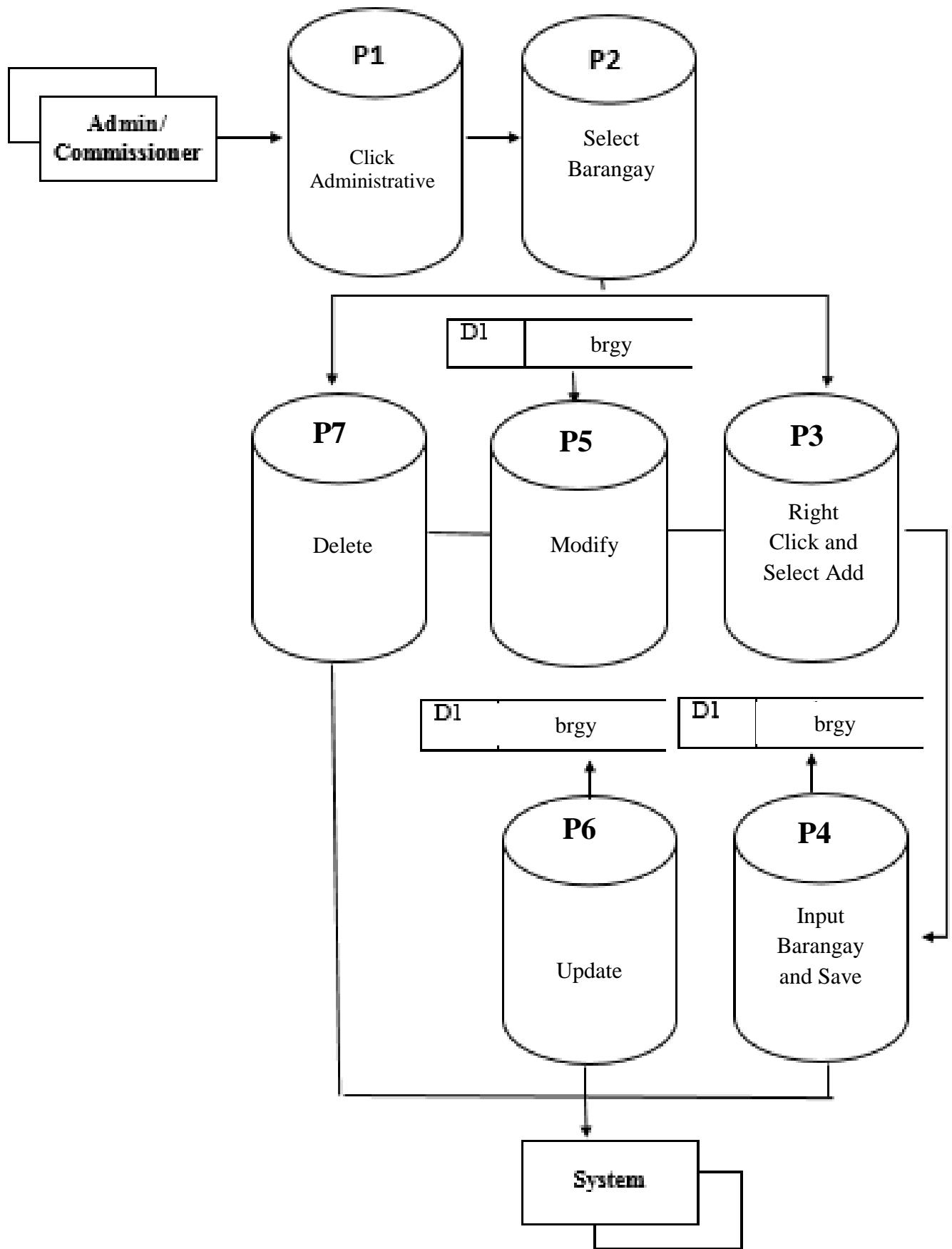


Figure 15: Setup Barangay (Proposed)

Setup Institute

Narrative Flow: To manage Institute information, choose right click and select Add and may perform modify, refresh and delete. To add, input Institute name and save; To modify, input changes and click update.

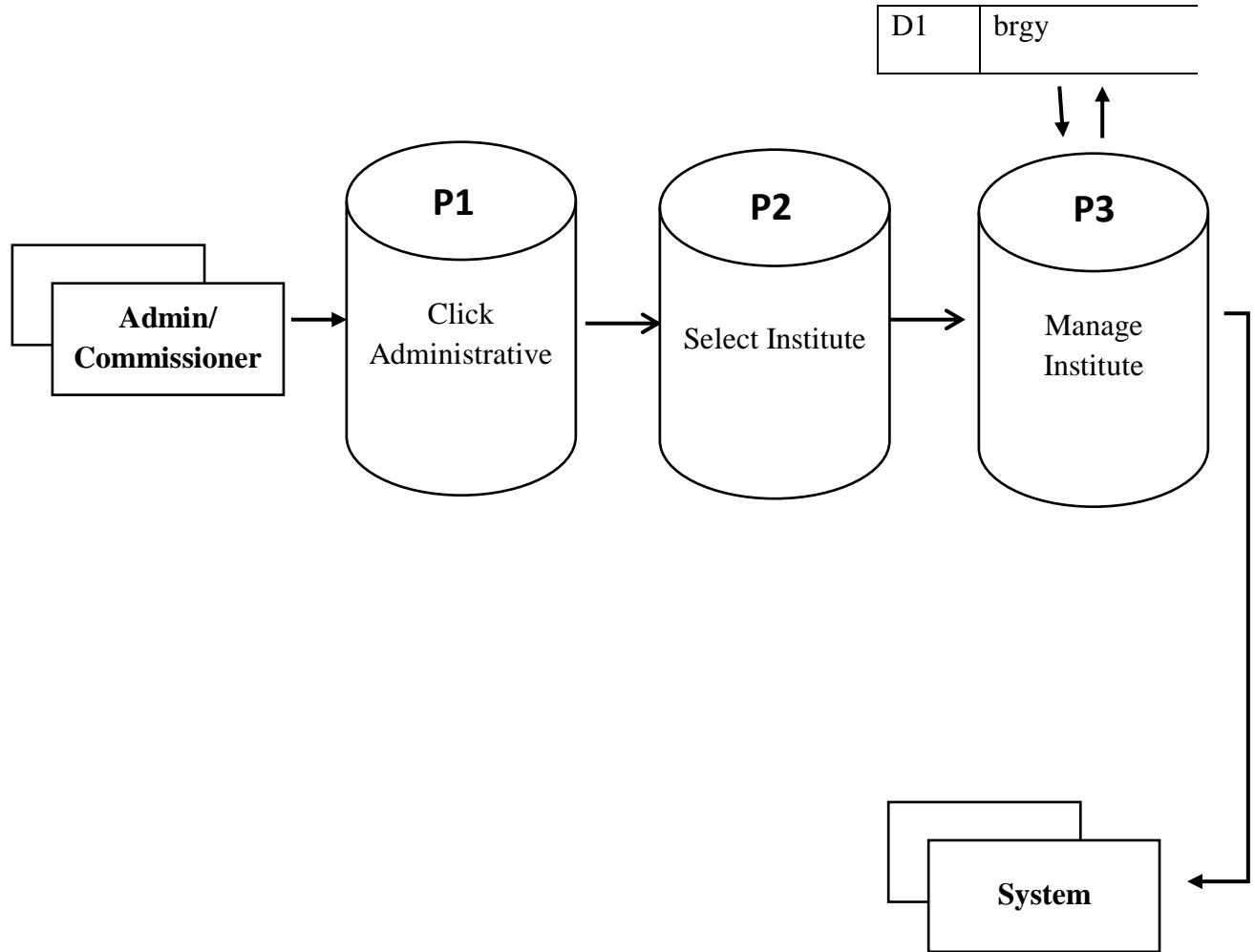


Figure 16: Setup Institute (Proposed)

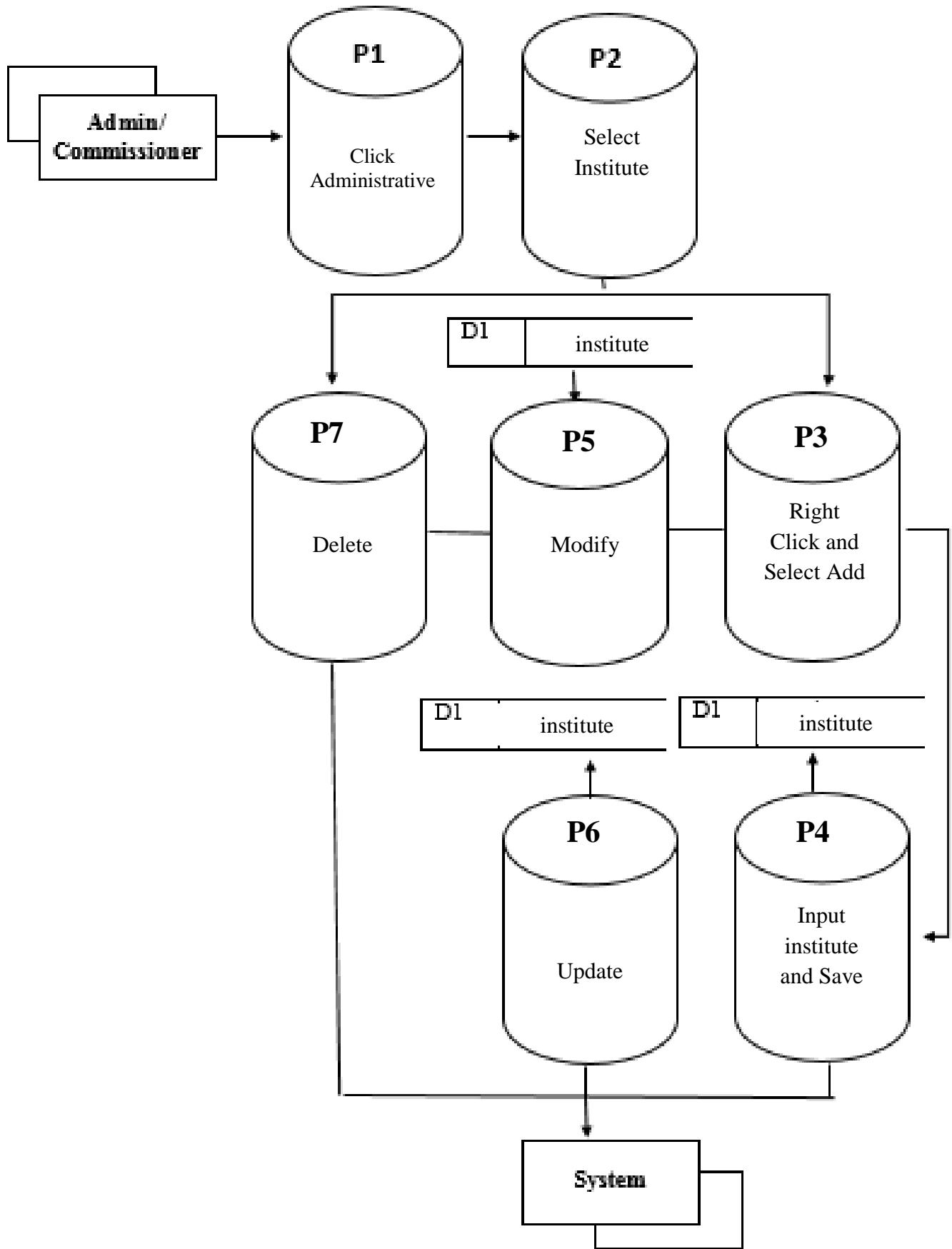


Figure 16: Setup Institute (Proposed)

Setup Programs

Narrative Flow: To manage Institute information, choose right click and select Add and may perform modify, refresh and delete. To add, input Institute name and save; To modify, input changes and click update.

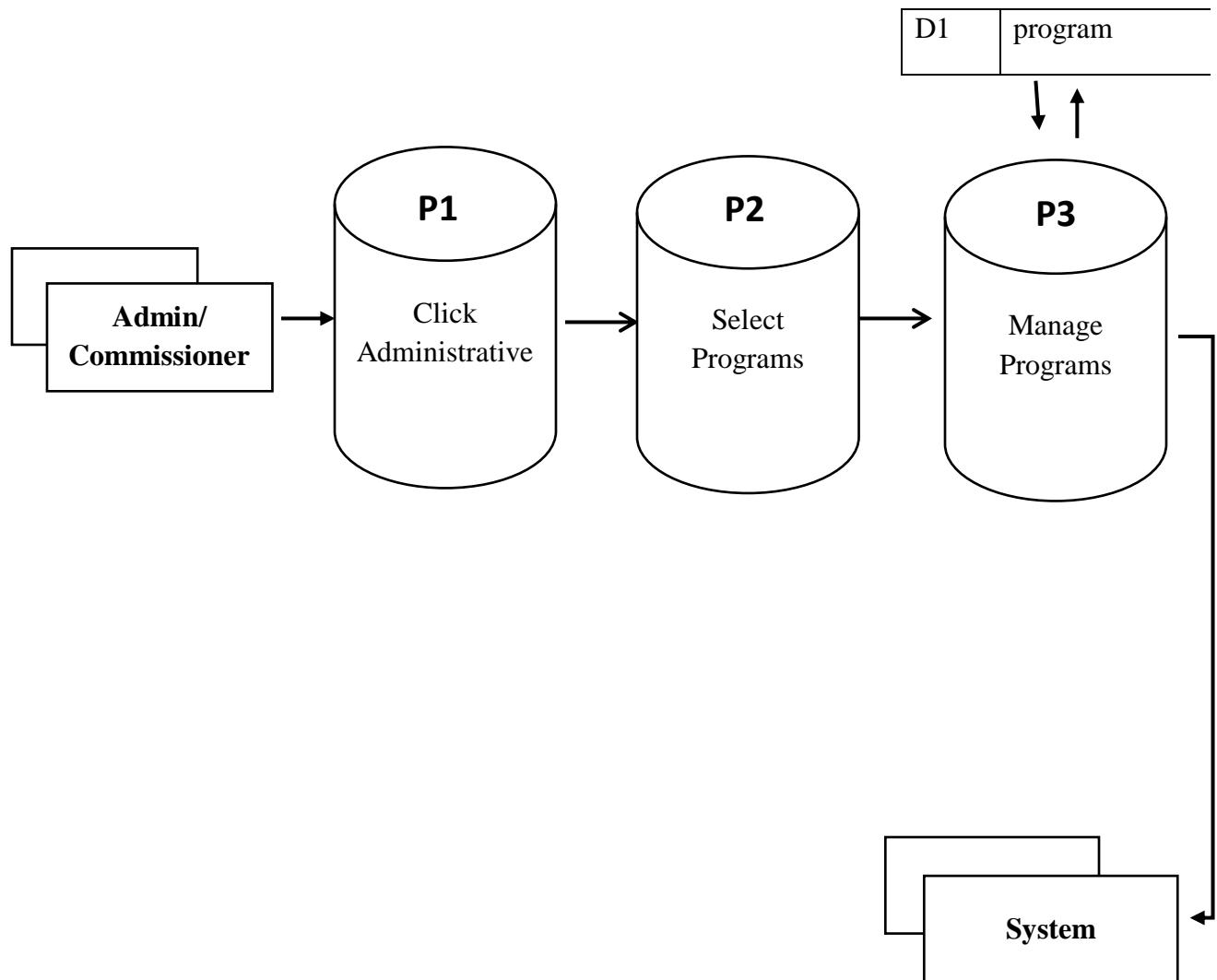


Figure 17: Setup Programs (Proposed)

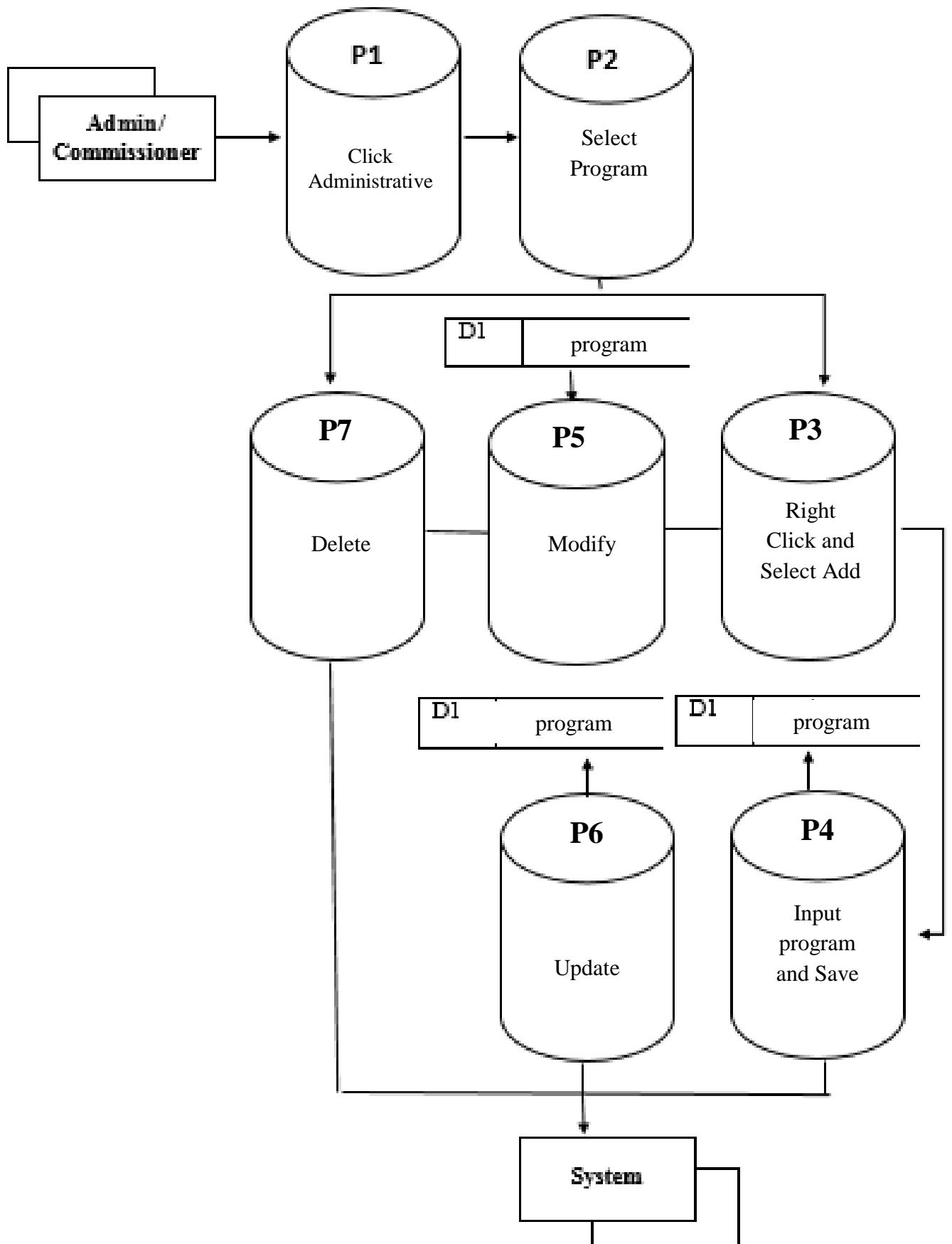


Figure 17: Setup Programs (Proposed)

Setup User's Position

Narrative Flow: To manage User's Position information, choose right click and select Add and may perform modify, refresh and delete. To add, input User's Position name and save; To modify, input changes and click update.

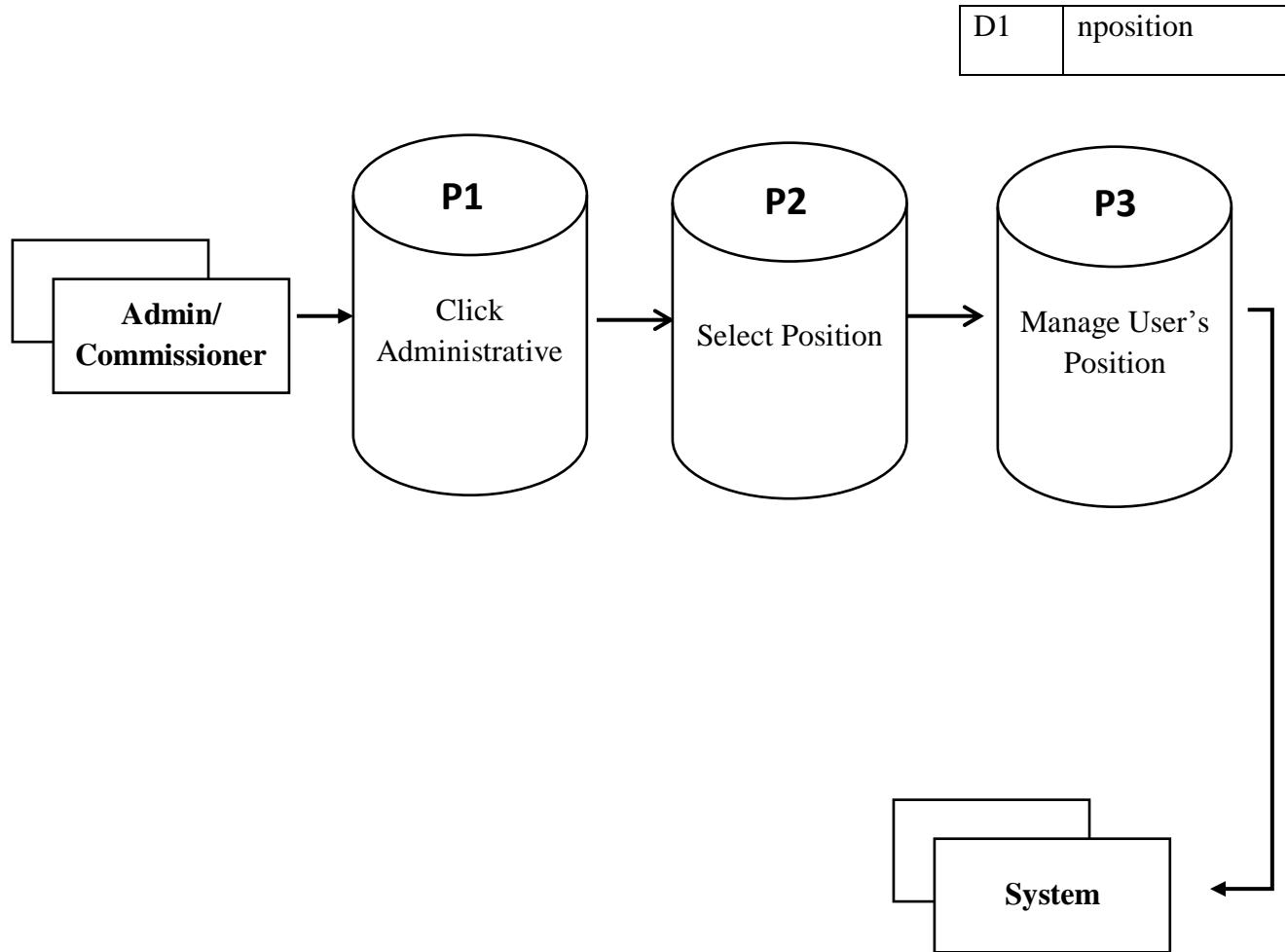


Figure 18: Setup User's Position (Proposed)

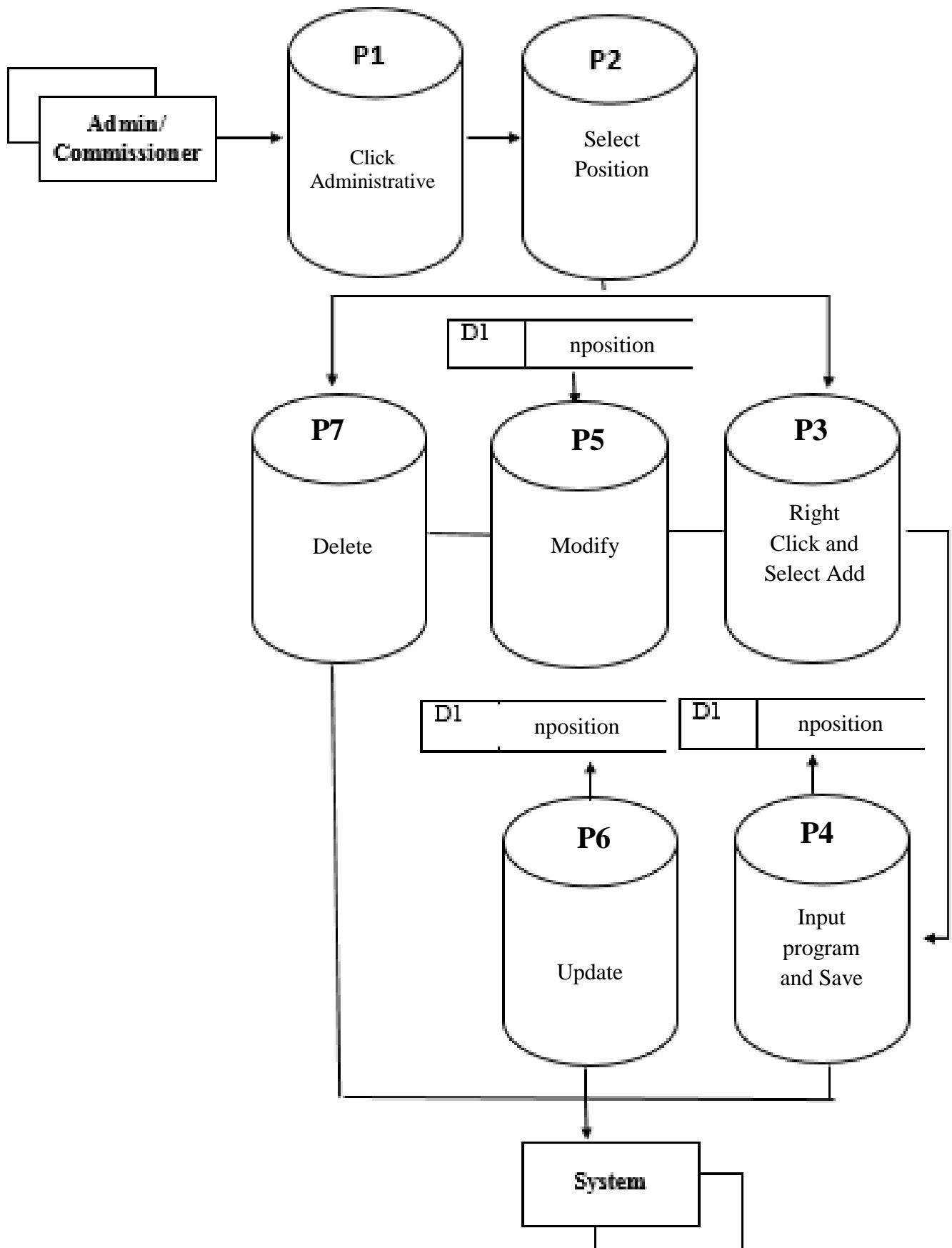


Figure 18: Setup User's Position (Proposed)

Registration (Users)

Narrative Flow: The Administrator will click the Account button then select add to input user's information then click save. The data will save then to the user_info table in database of the system.

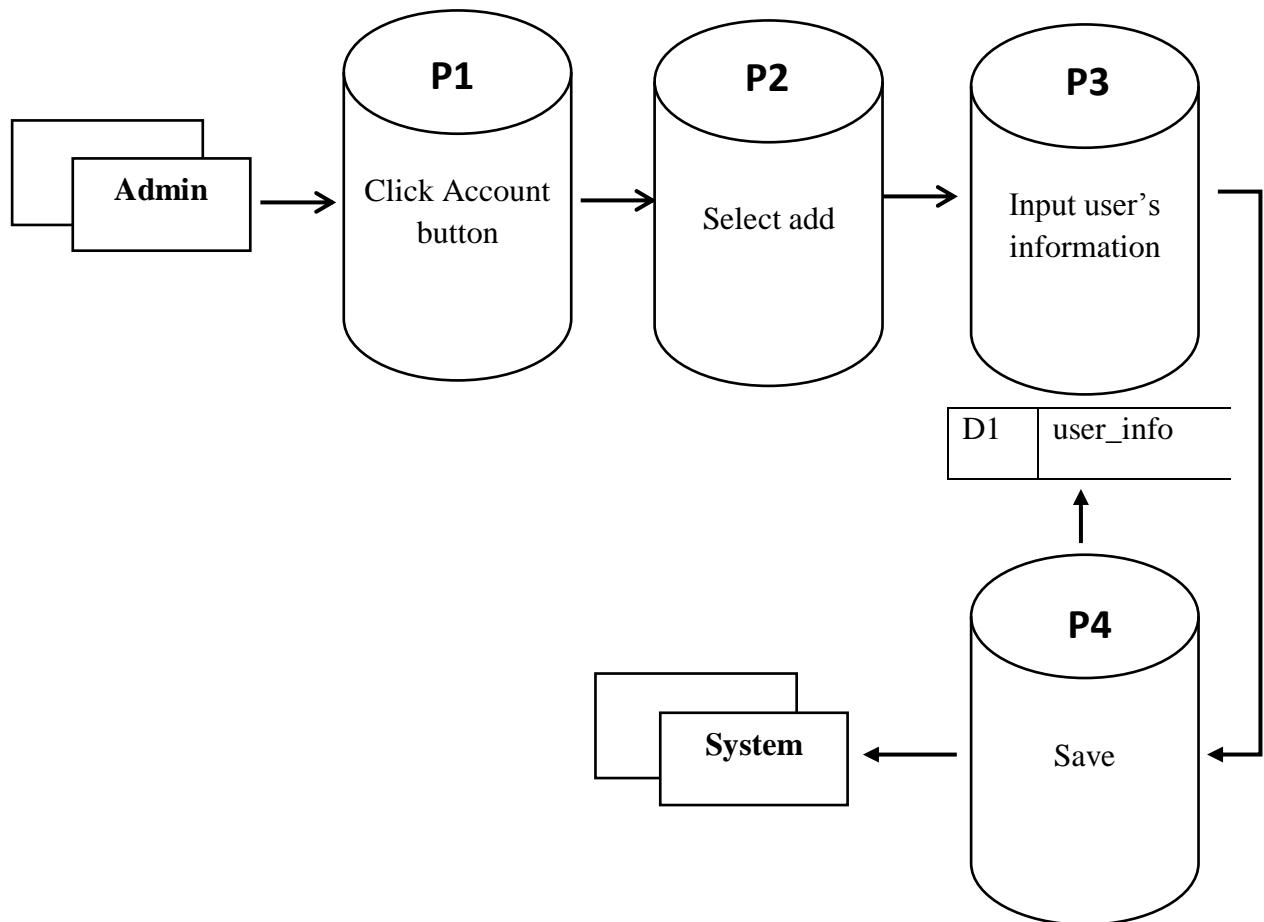


Figure 19: Registration of Users (Proposed)

Setup Voting (Add)

Narrative Flow: The Administrator/Commissioner will Click Setup Voting button then click add button to setup election type. To add default election setup, the admin/commissioner will select Academic Year then input election Name and Date and it will then be saved to the database in the system.

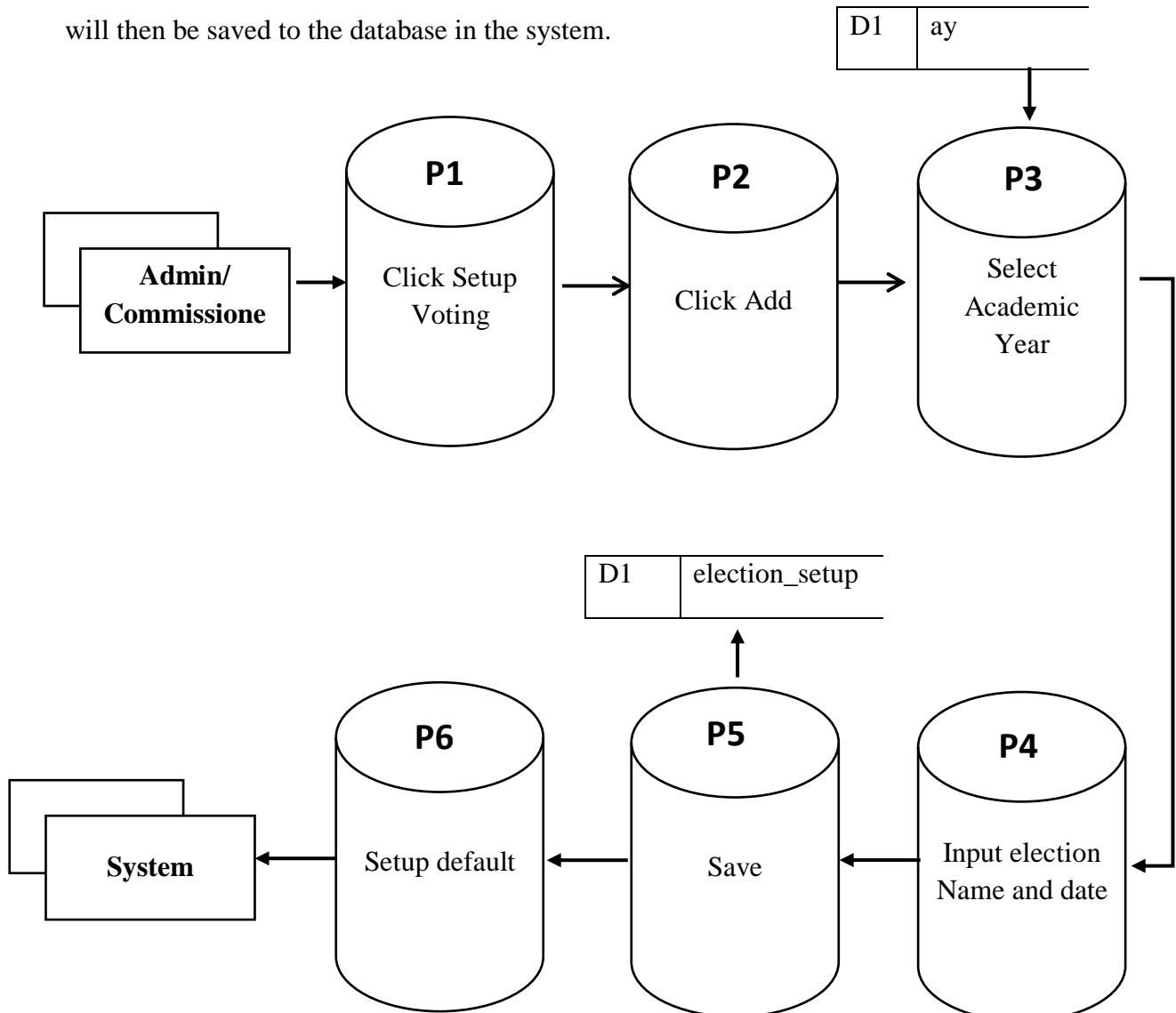


Figure 20: Setup Voting-Add (Proposed)

Setup Voting (Edit)

Narrative Flow: The Administrator/Commissioner will Click Setup Voting button then select an election type from the list and then click Edit button to input any necessary changes and then click Save button and the information is being updated and save it to the database in the system.

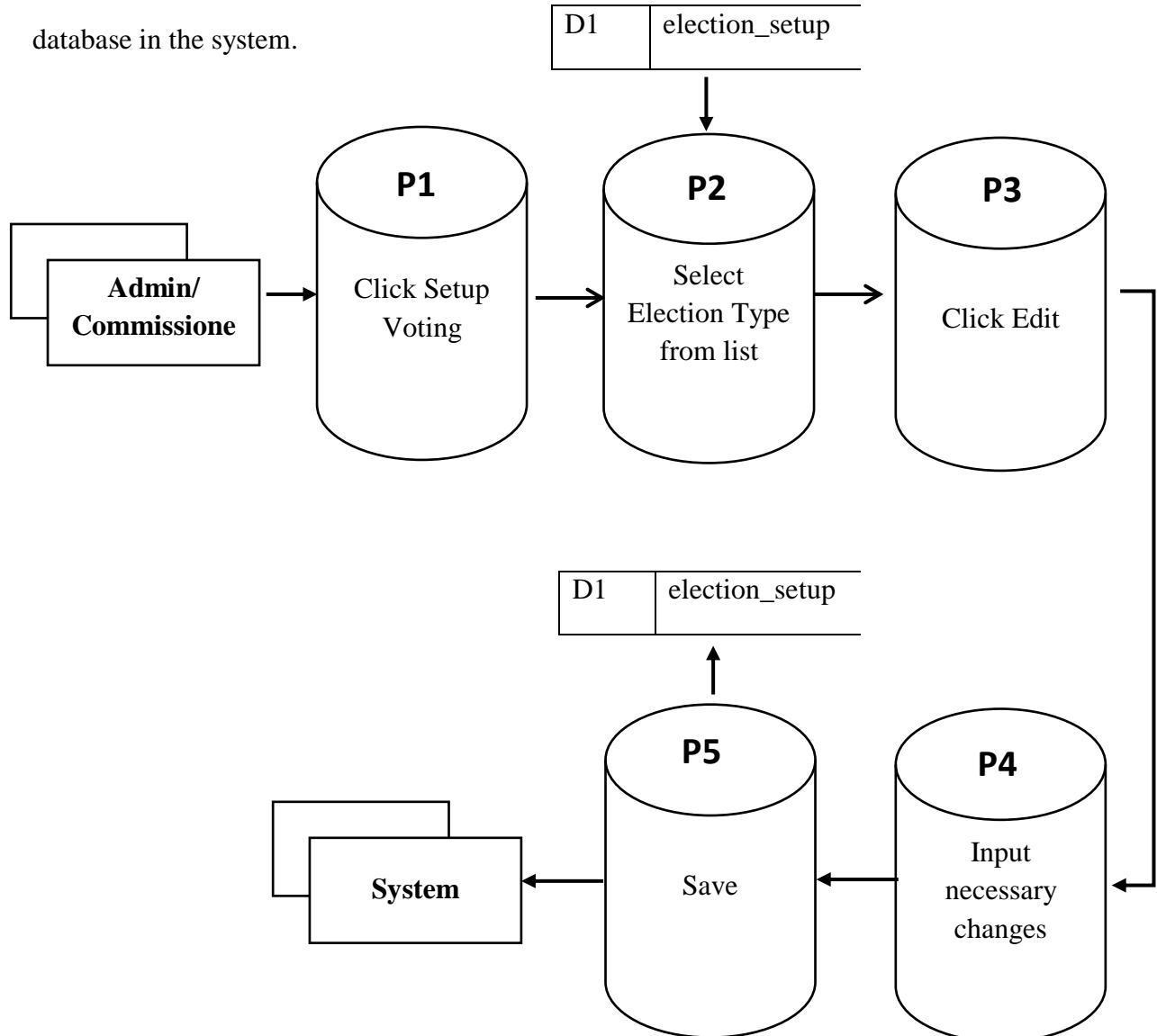


Figure 21: Setup Voting-Edit (Proposed)

Setup Voting (Delete)

Narrative Flow: The Administrator/Commissioner will Click Setup Voting button then select an election type from the list and then click Delete button. A dialog box will appear asking the user to permanently delete the information. Select and Click “Yes” to permanently remove the Election type from the list.

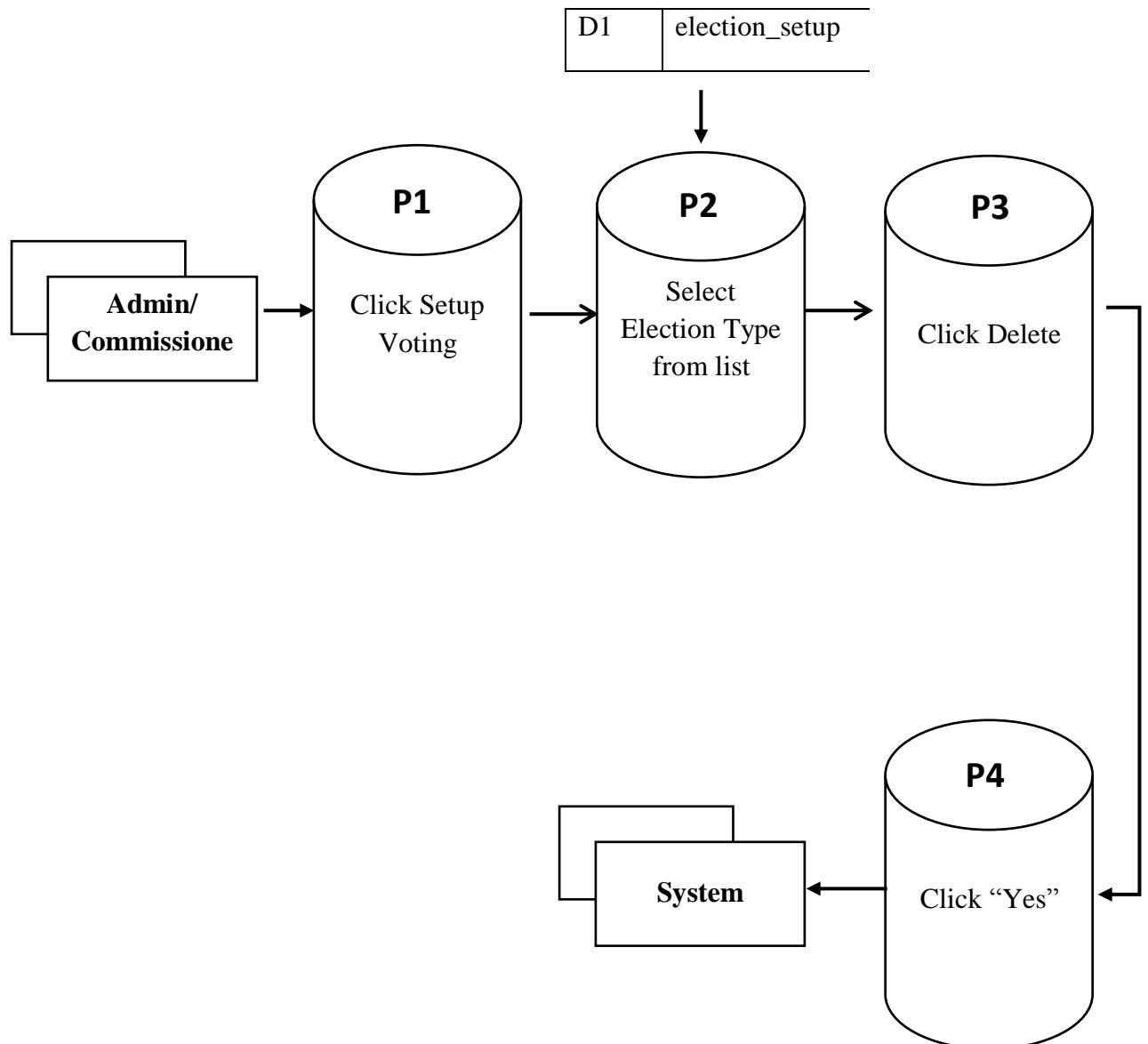


Figure 22: Setup Voting-Delete (Proposed)

Setup Position

Narrative Flow: The Administrator/Commissioner will select Setup Position button then click New button to setup position and may perform update and delete. Browse the election type and input position details. Position description must include the type of position either SRO, SBO or FAST Club then it will be saved to the database in the system.

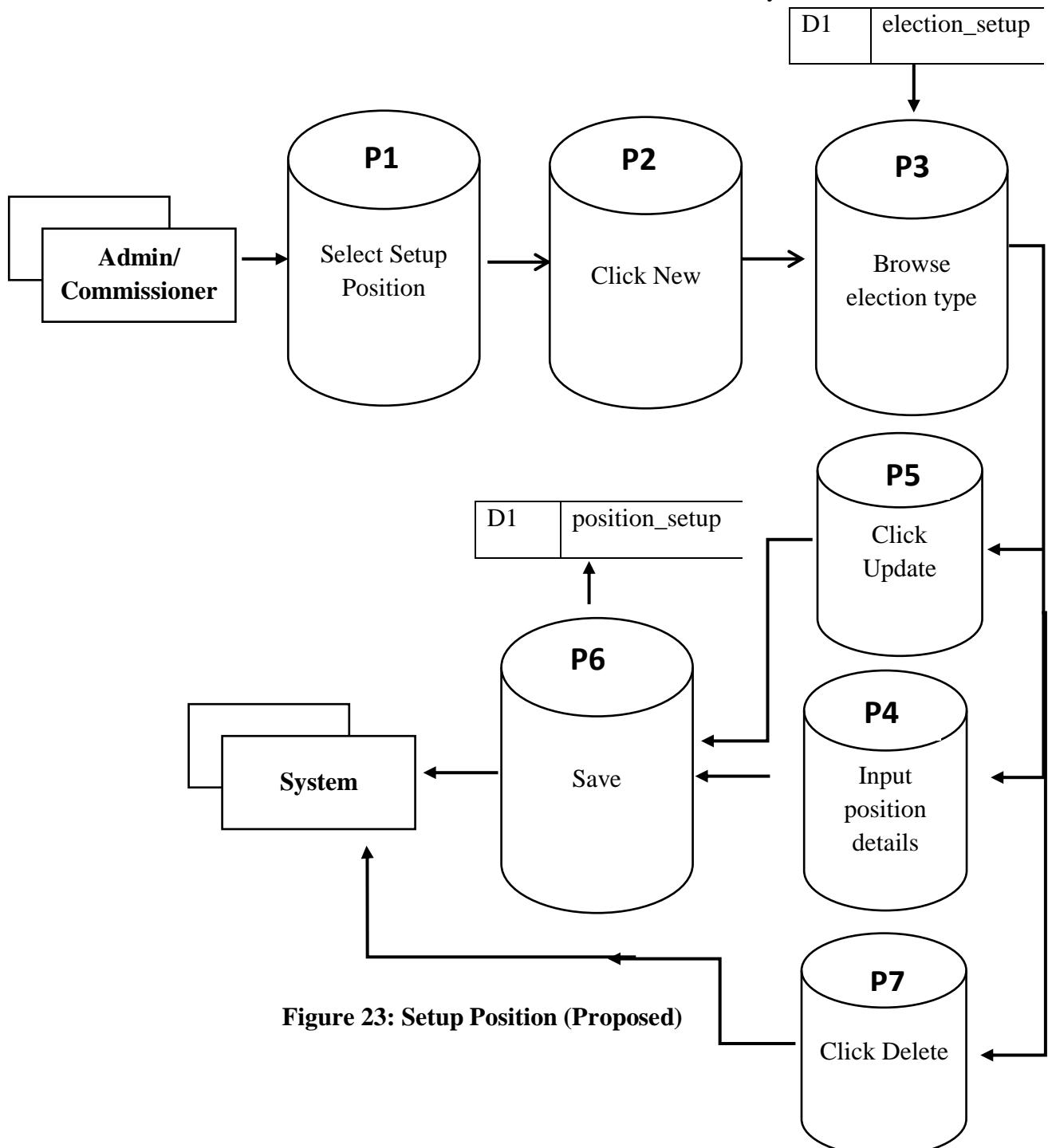


Figure 23: Setup Position (Proposed)

Registration of Unofficial Voters (Add)

Narrative Flow: The Administrator/Commissioner will select Unofficial Registered Voters button then click Add button to add voters. The admin/commissioner will then input the information of the voters then take photo, fingerprint and electronic signature that will be saved to the database in the system.

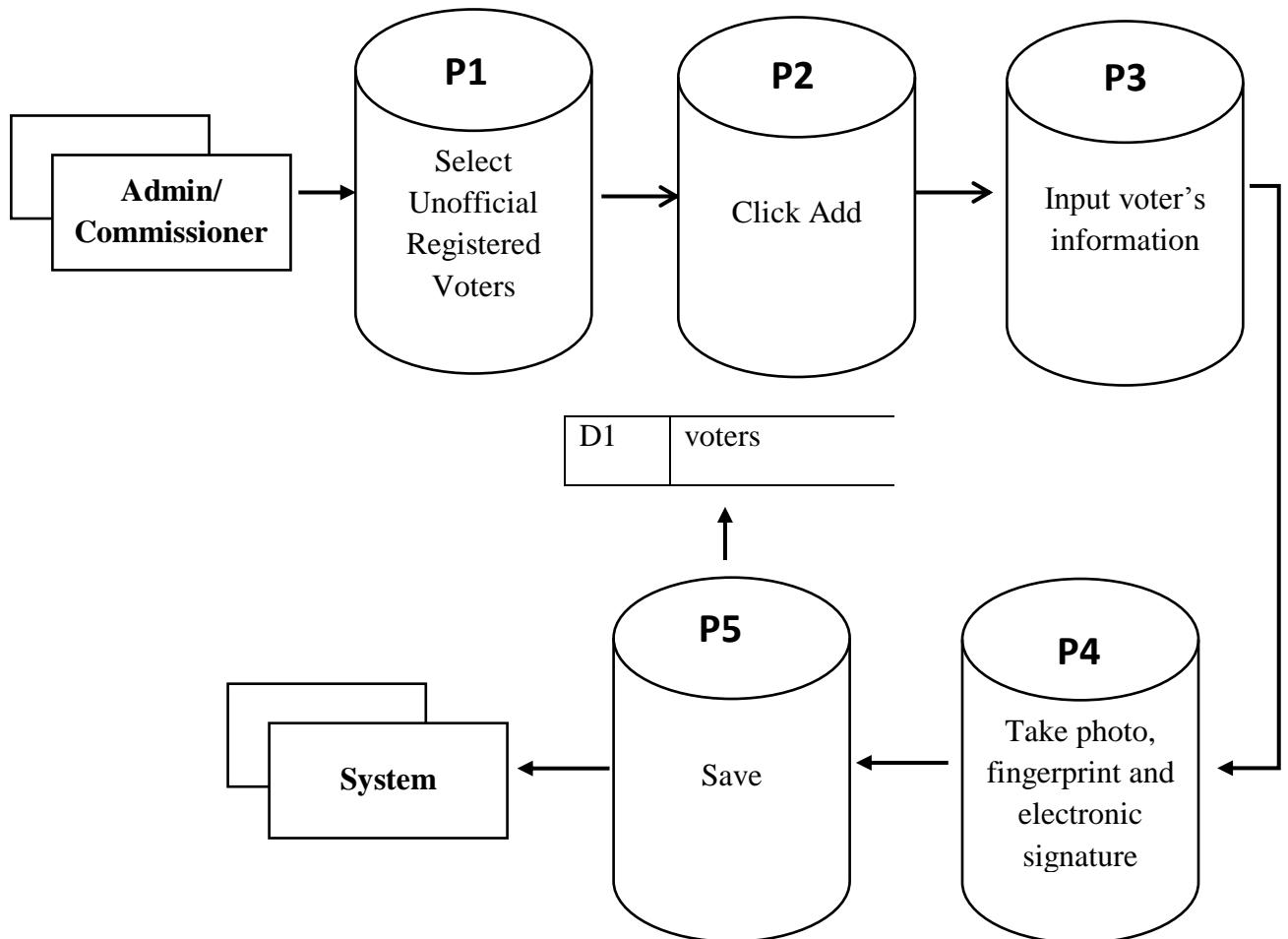


Figure 24: Registration of Unofficial Registered Voters-Add

Registration of Unofficial Voters (Edit)

Narrative Flow: The Administrator/Commissioner will select Unofficial Registered Voters button then select voters from list and click Edit button to do necessary changes that will be saved on the database in the system.

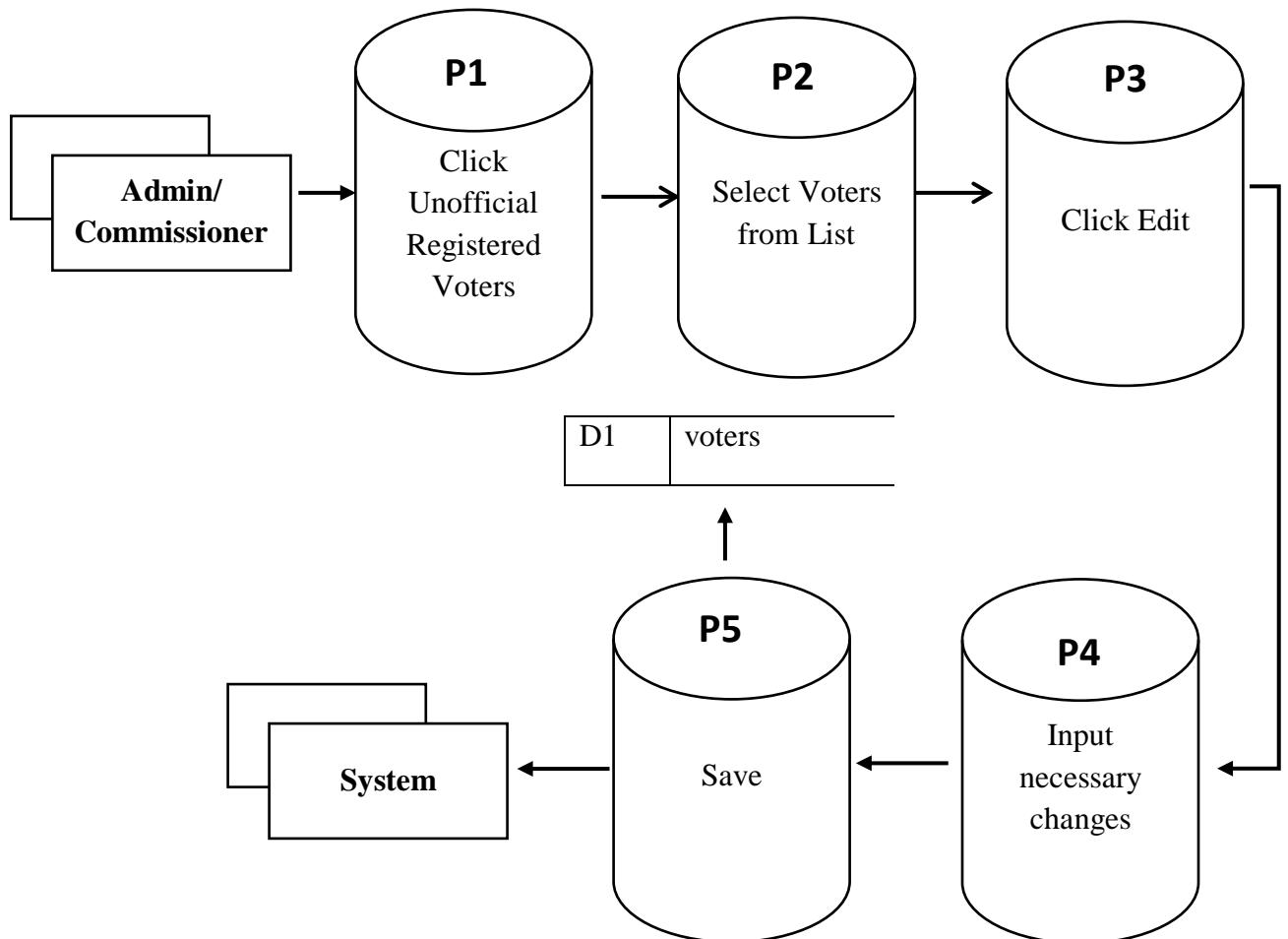


Figure 25: Registration of Unofficial Registered Voters-Edit

Registration of Unofficial Voters (Delete)

Narrative Flow: The Administrator/Commissioner will select Unofficial Registered Voters button then Select voters from list and then click Delete button. A dialog box will appear asking the user to permanently delete the information. Select and Click “Yes” to permanently remove the voters from the list.

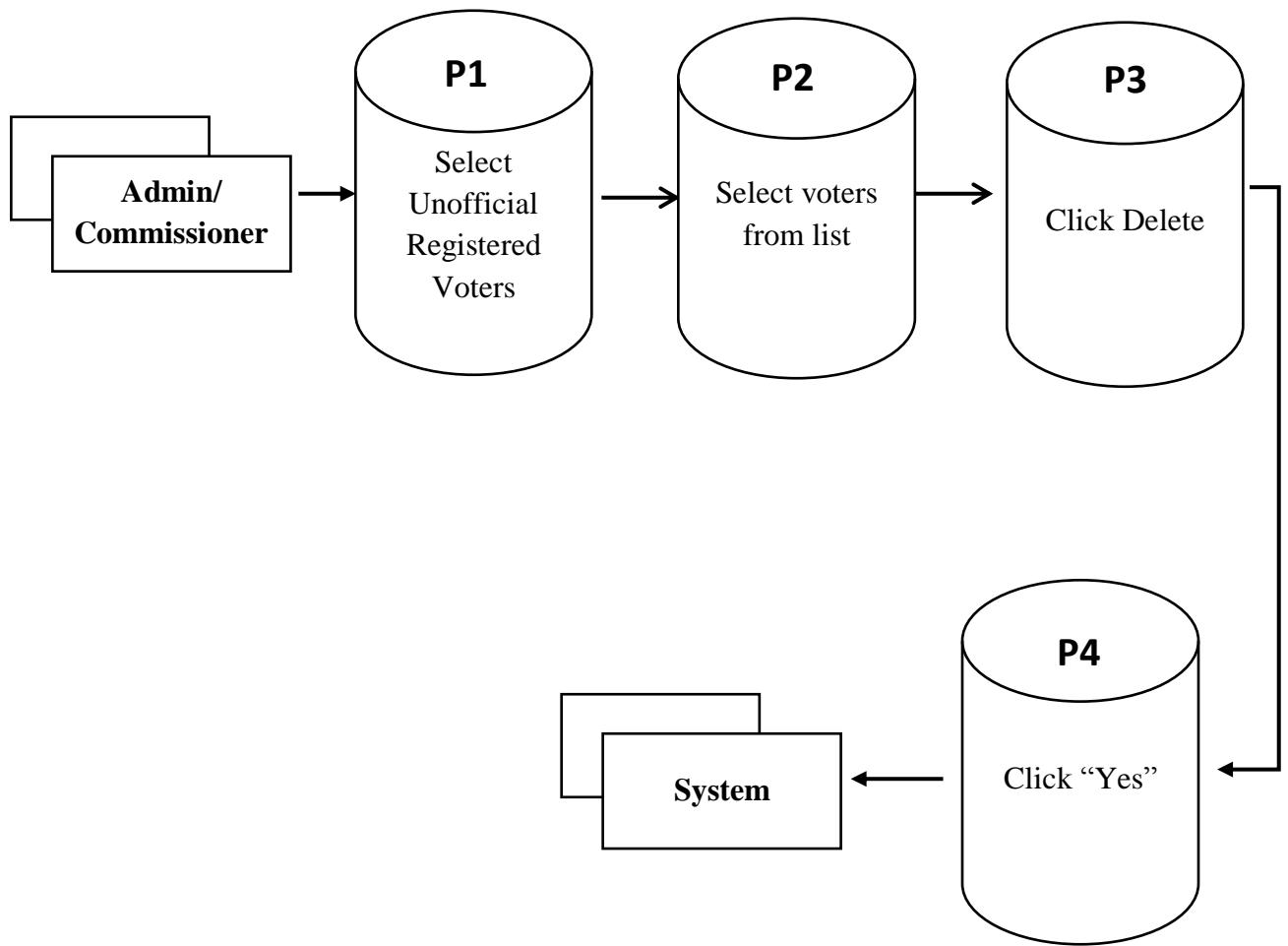


Figure 26: Registration of Unofficial Registered Voters Delete (Proposed)

Registration of an Official Voters (Add)

Narrative Flow: The Administrator/Commissioner will click Setup Voting then perform right click and select Show Registered Voters to display the official voter's list of the default voting process. Right click and select Register Voters from List in order to add new official voters. Select/Highlight voters you want to add then click the Add Voter(s) button to officially register the voters.

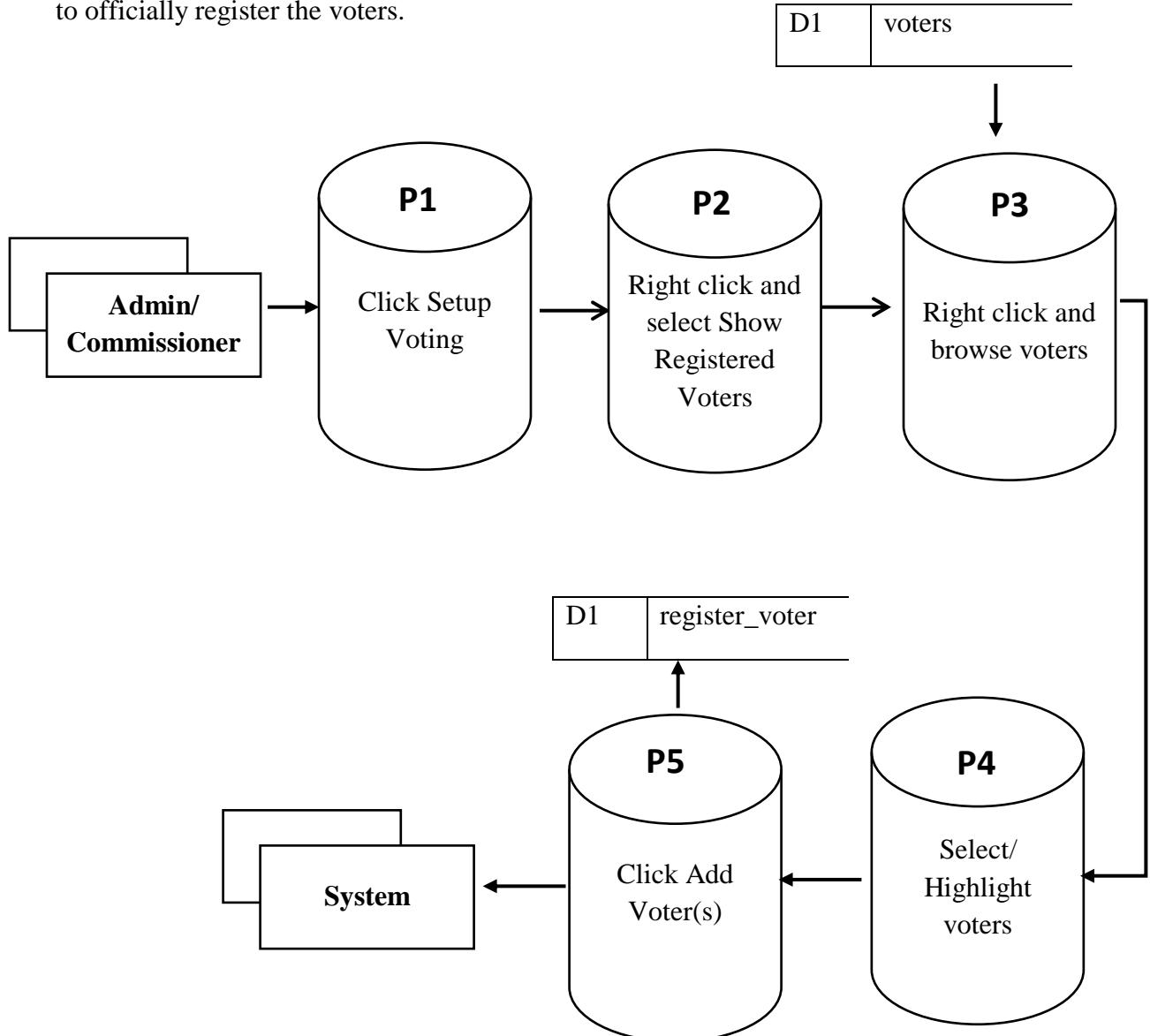


Figure 27: Registration of Official Registered Voters-Add (Proposed)

Registration of an Official Voters (Remove)

Narrative Flow: The Administrator/Commissioner will click Setup Voting then perform right click and select Show Registered Voters to display the official voter's list of the default voting process. Select/Highlight voters you want to remove from list and a dialog box will appear asking you to remove information from list. By clicking “Yes” button, the data will be removed automatically.

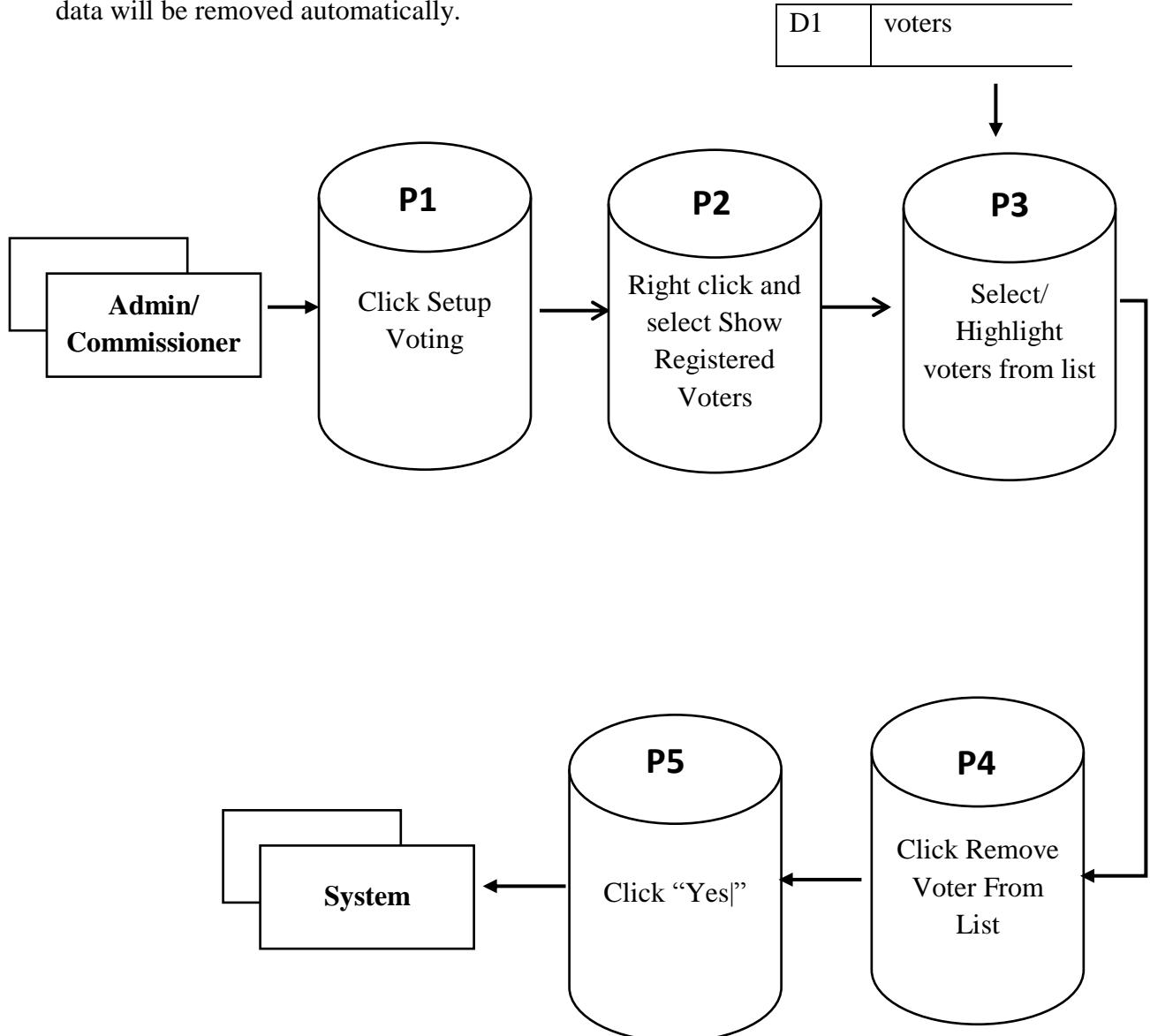


Figure 28: Registration of Official Registered Voters-Remove

Setup Candidate's Requirements (Add)

Narrative Flow: The Administrator/Commissioner will select Setup Candidates Requirements and then right click and select new requirements then select an election type and input requirement on description that will be saved to the database in the system.

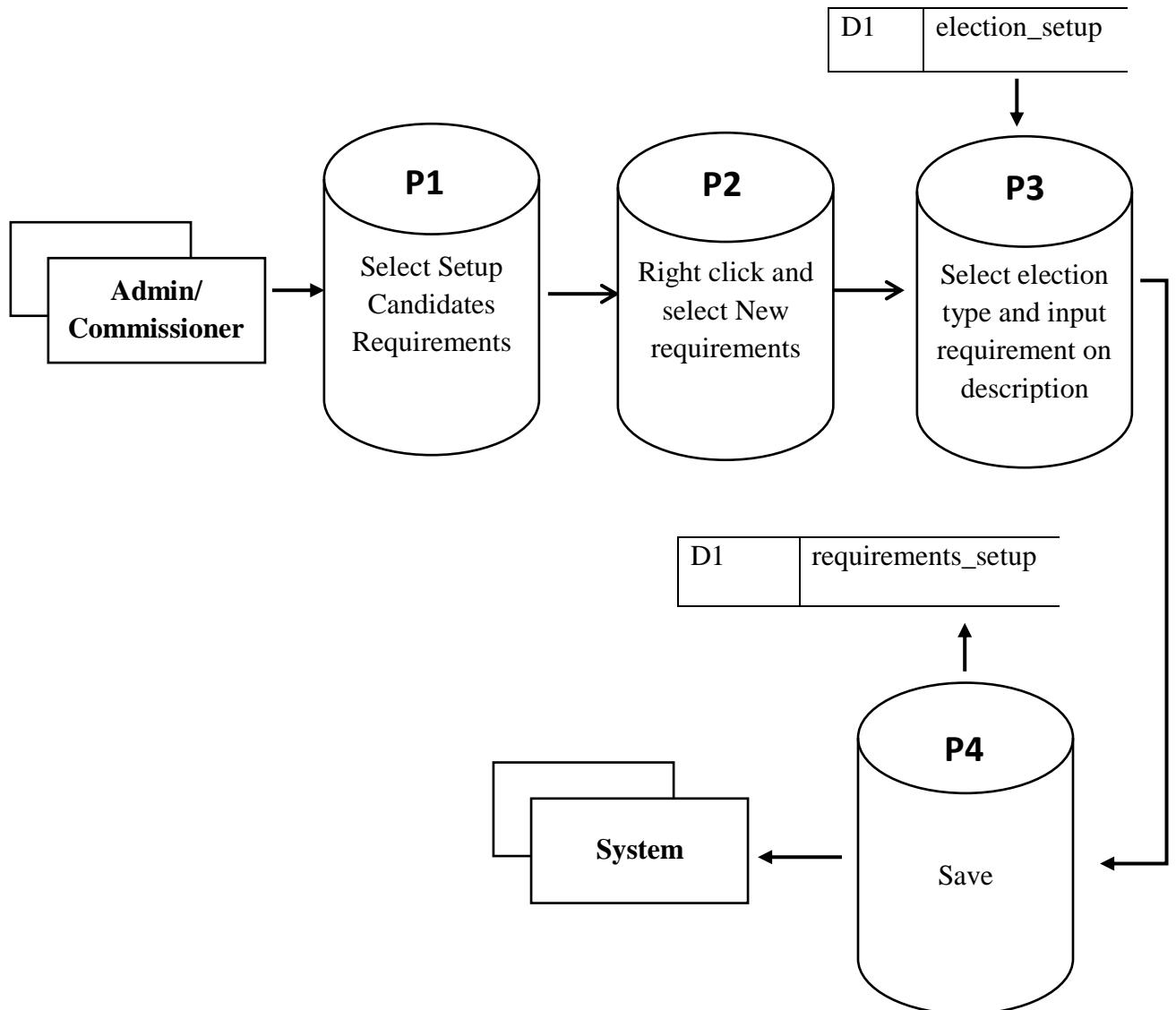


Figure 29: Setup Candidate's Requirements-Add (Proposed)

Setup Candidate's Requirements (Modify Setup)

Narrative Flow: The Administrator/Commissioner will select Setup Candidates Requirements and then select a requirements information from list and click Modify Setup to make any necessary changes and save it to the database in the system.

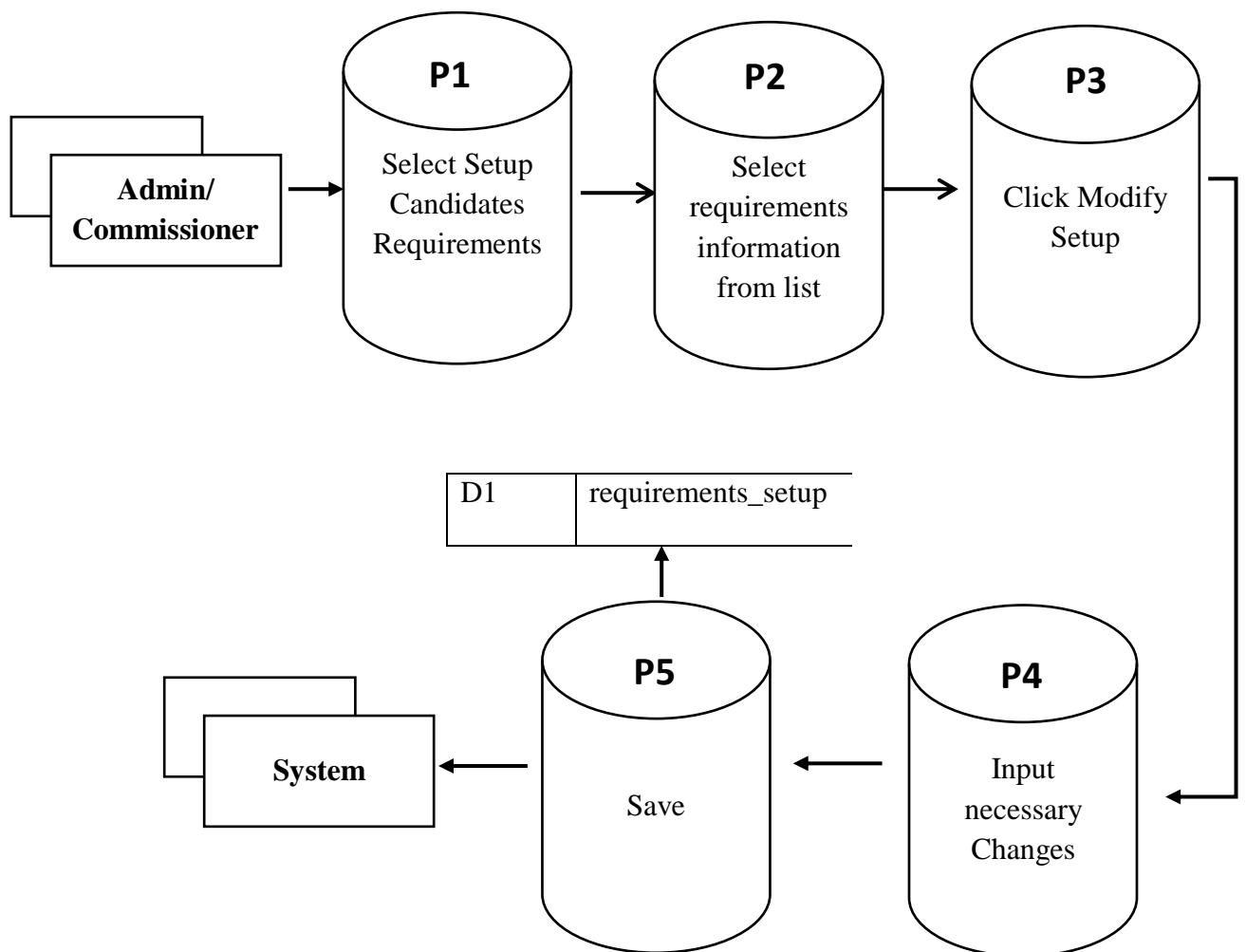


Figure 30: Setup Candidate's Requirements-Modify Setup (Proposed)

Setup Candidate's Requirements (Delete Setup)

Narrative Flow: The Administrator/Commissioner will Click Setup Candidates Requirements and then select a requirements information from list and click Delete Setup remove the candidate's requirements information and a dialog box will appear asking the user to click “Yes” to remove permanently the information.

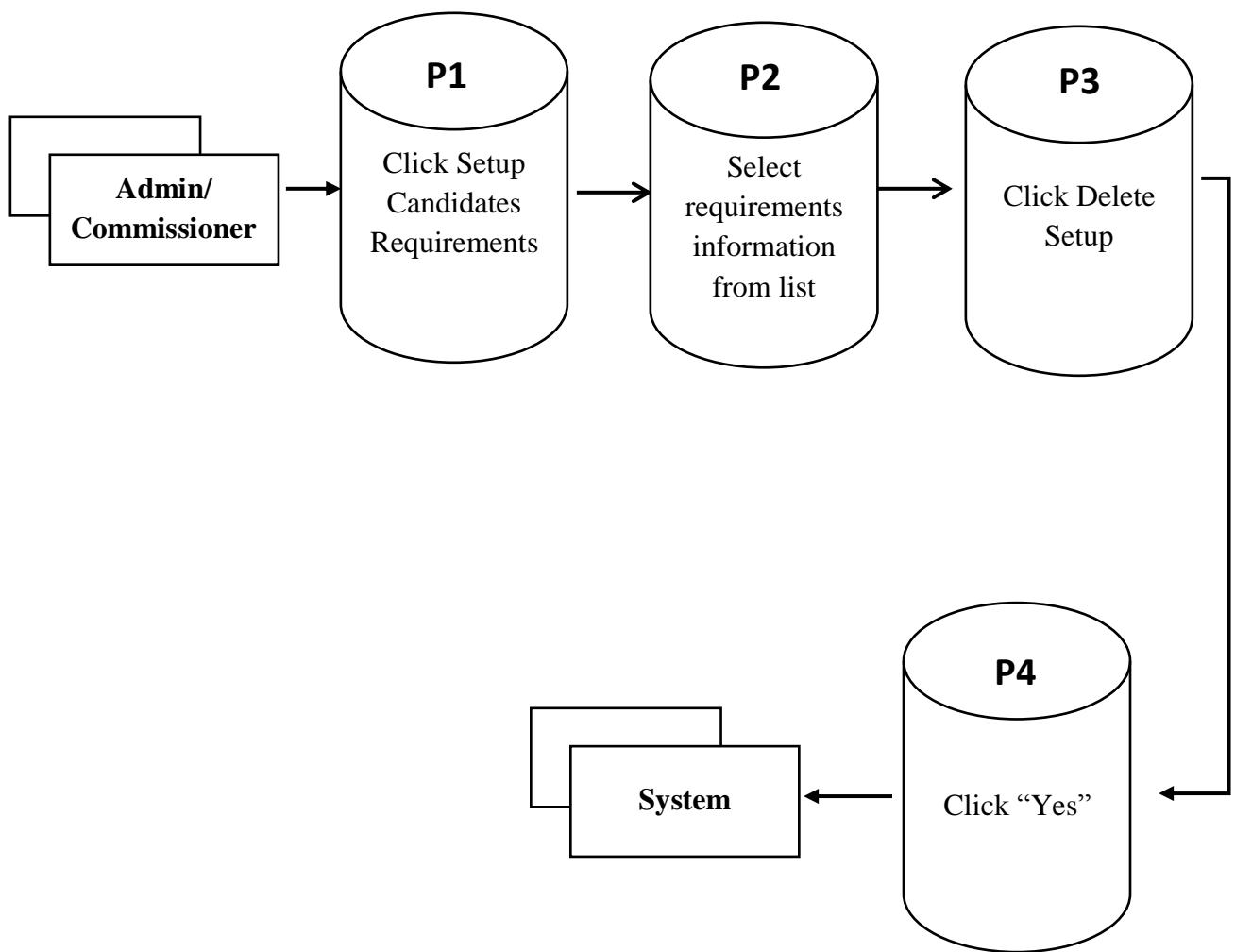


Figure 31: Setup Candidate's Requirements-Delete Setup (Proposed)

Registration (Candidates)

Narrative Flow: The Administrator/Commissioner will select Candidates button then right click and select New Candidate in order to add new running candidate of the default voting. Browse then the available position and also browse the official voter's list then save the selected voters as candidate.

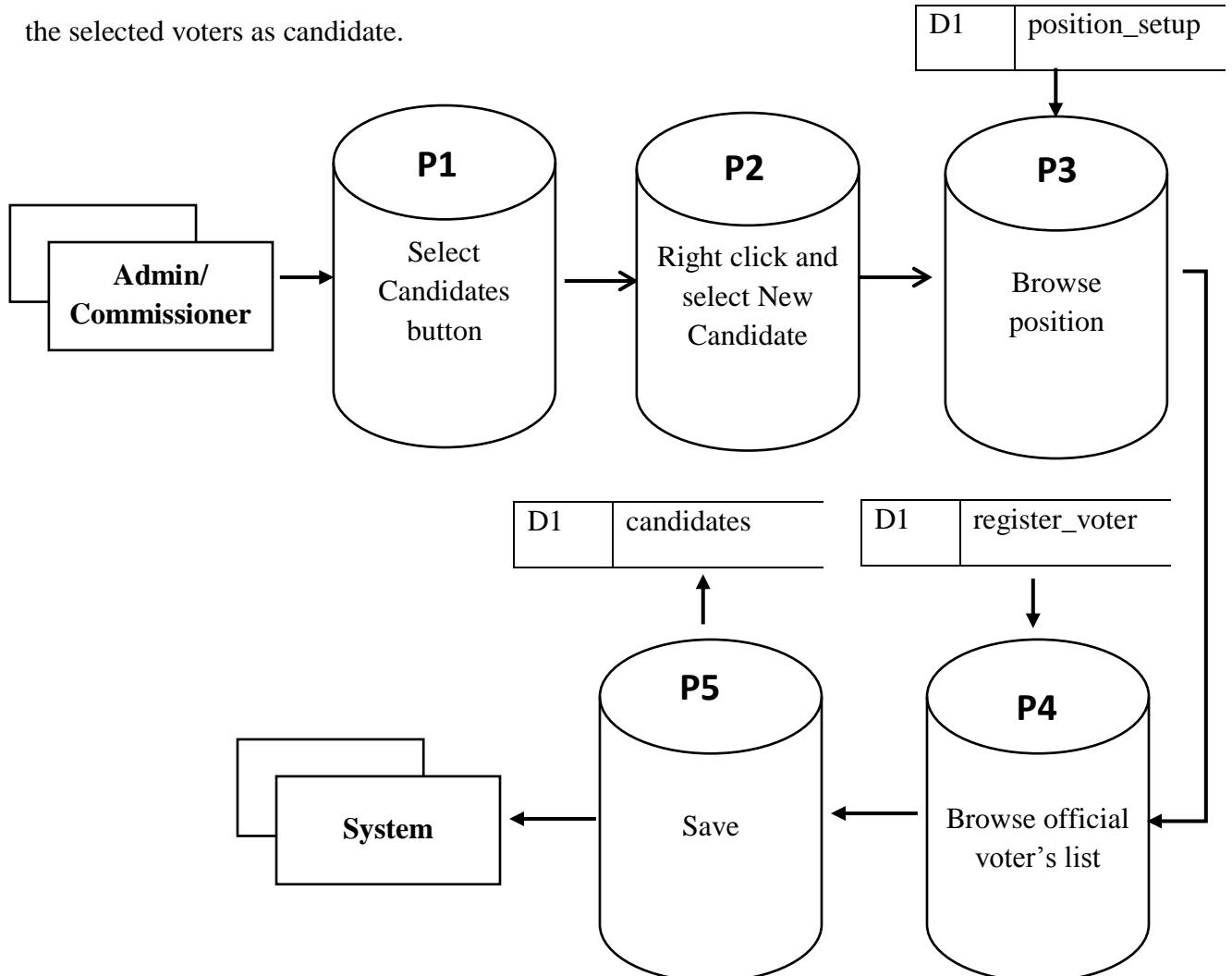


Figure 32: Registration of Candidates (Proposed)

Modify Candidates

Narrative Flow: The Administrator/Commissioner will click Candidates button then select a candidate's information from list and click Modify Candidates to make necessary changes of information and click Save button to Save the data in the system.

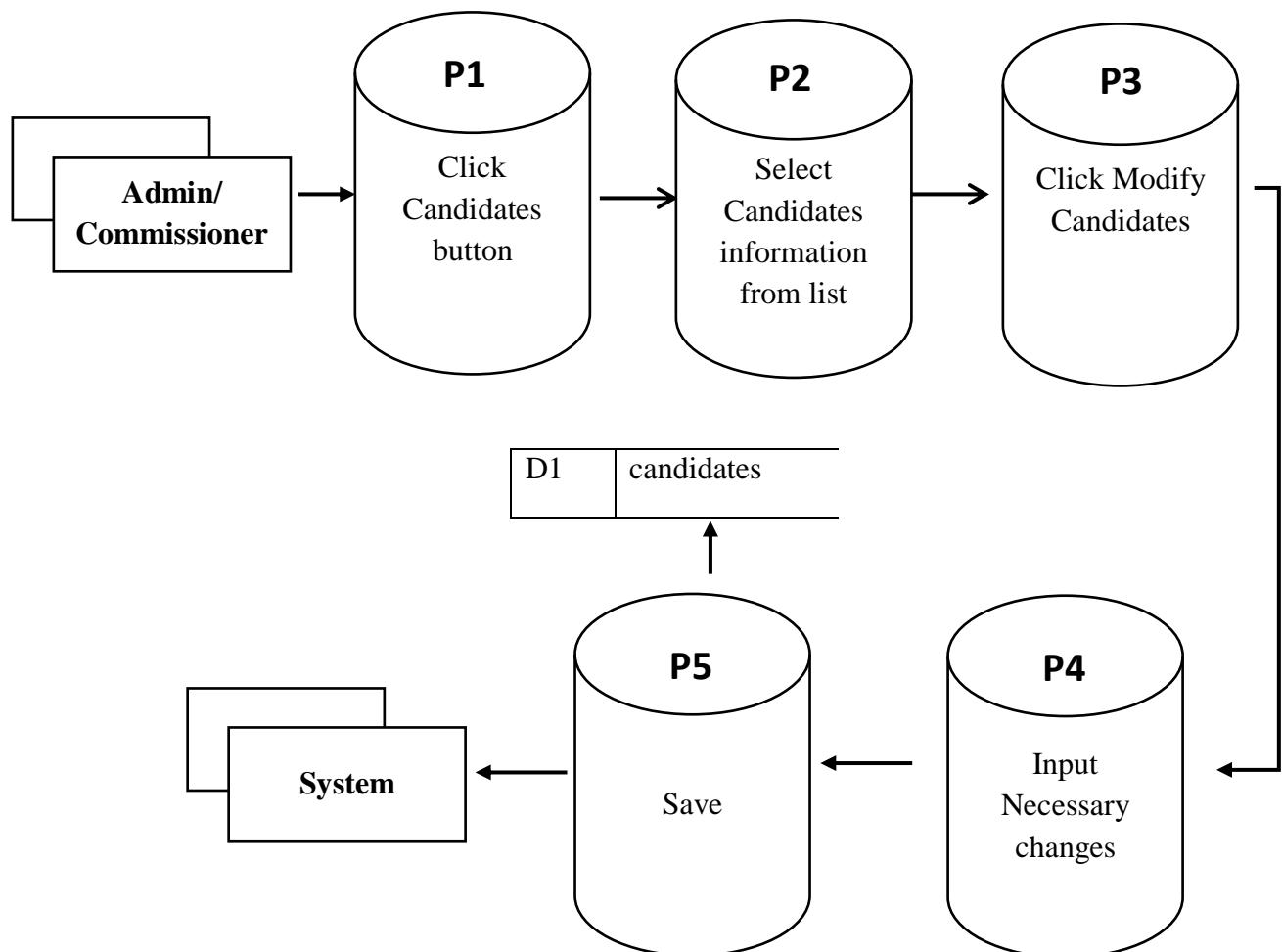


Figure 33: Modify Candidates (Proposed)

Delete Candidates

Narrative Flow: The Administrator/Commissioner will click Candidates button then select a candidate's information from list and click Delete Candidates button remove the information and a dialog box will appear asking the user to click "Yes" to remove permanently the candidate's information in the system.

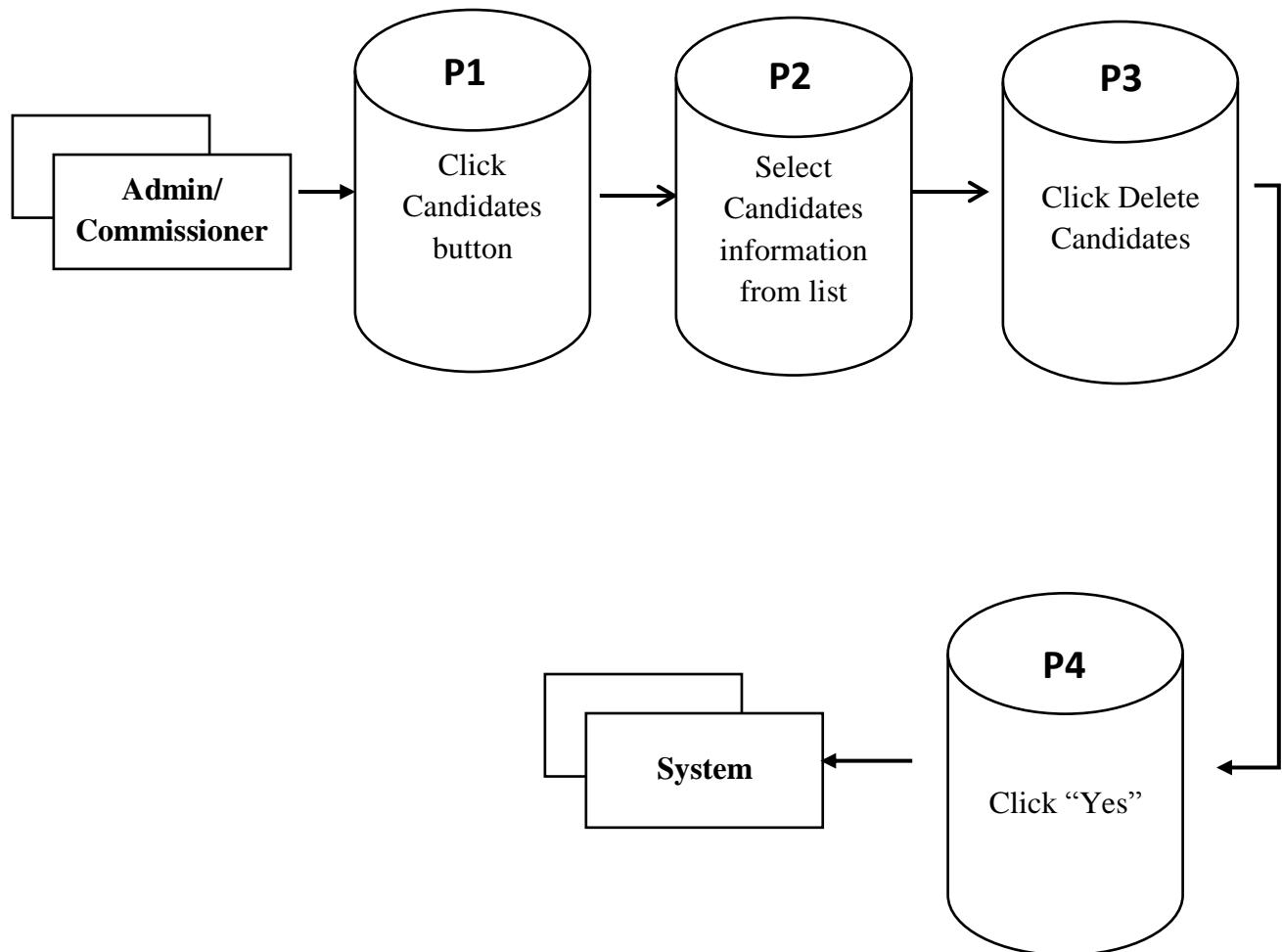


Figure 34: Delete Candidates (Proposed)

Voting Process

Narrative Flow: The voters will connect to wifi or to the network being used by the college. They will then open the Domain host and Login the Username and Password to vote for a candidate and then submit the selected candidates to be saved and calculated by the system.

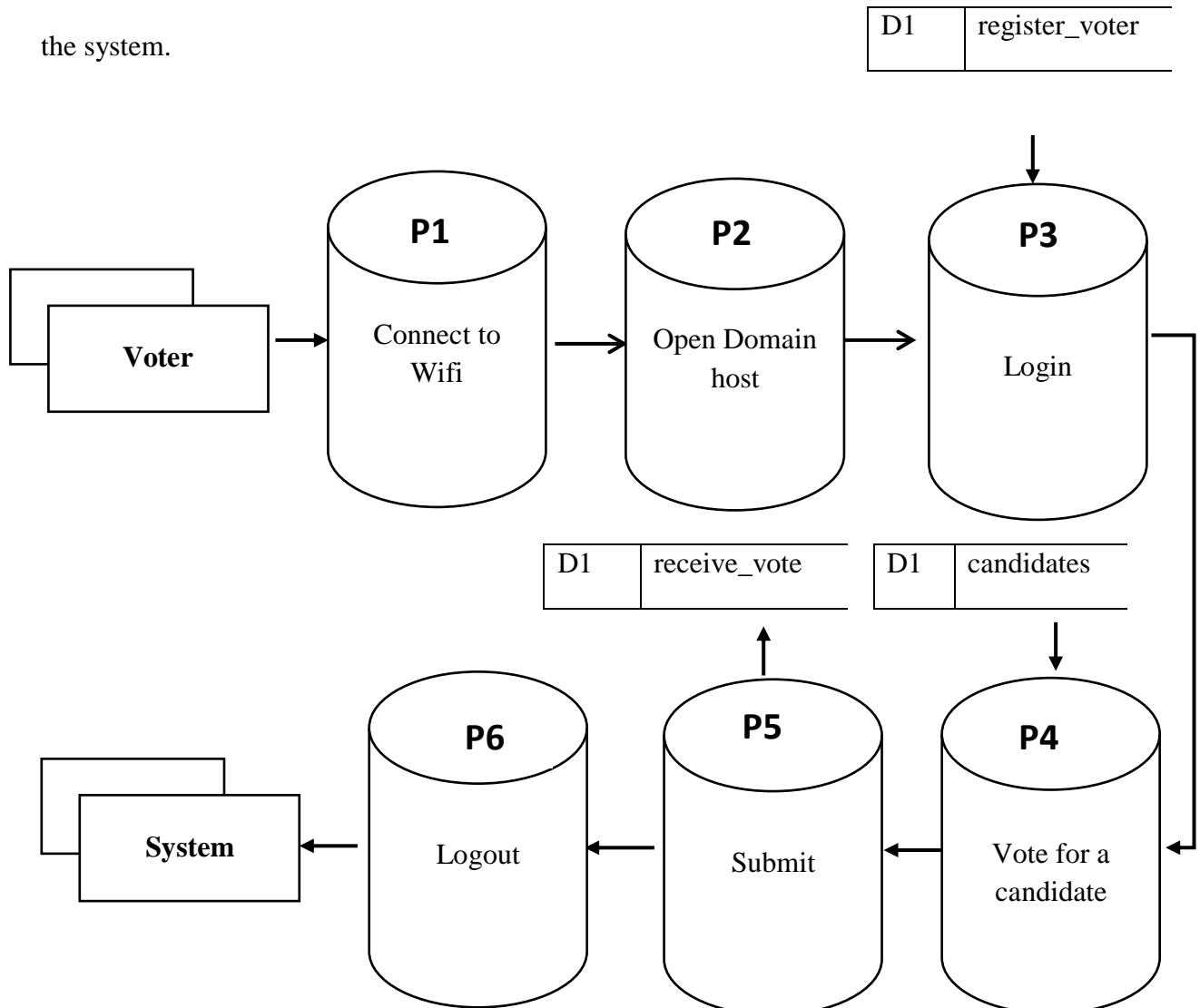


Figure 35: Voting Process (Proposed)

View and Print result

Narrative Flow: The Administrator/Commissioner will click the result after being login.

He/She will then select institute to view the result then click preview and then print result.

The comelec chairman will then affix his/her signature then give the final result to the prefect of the student activities.

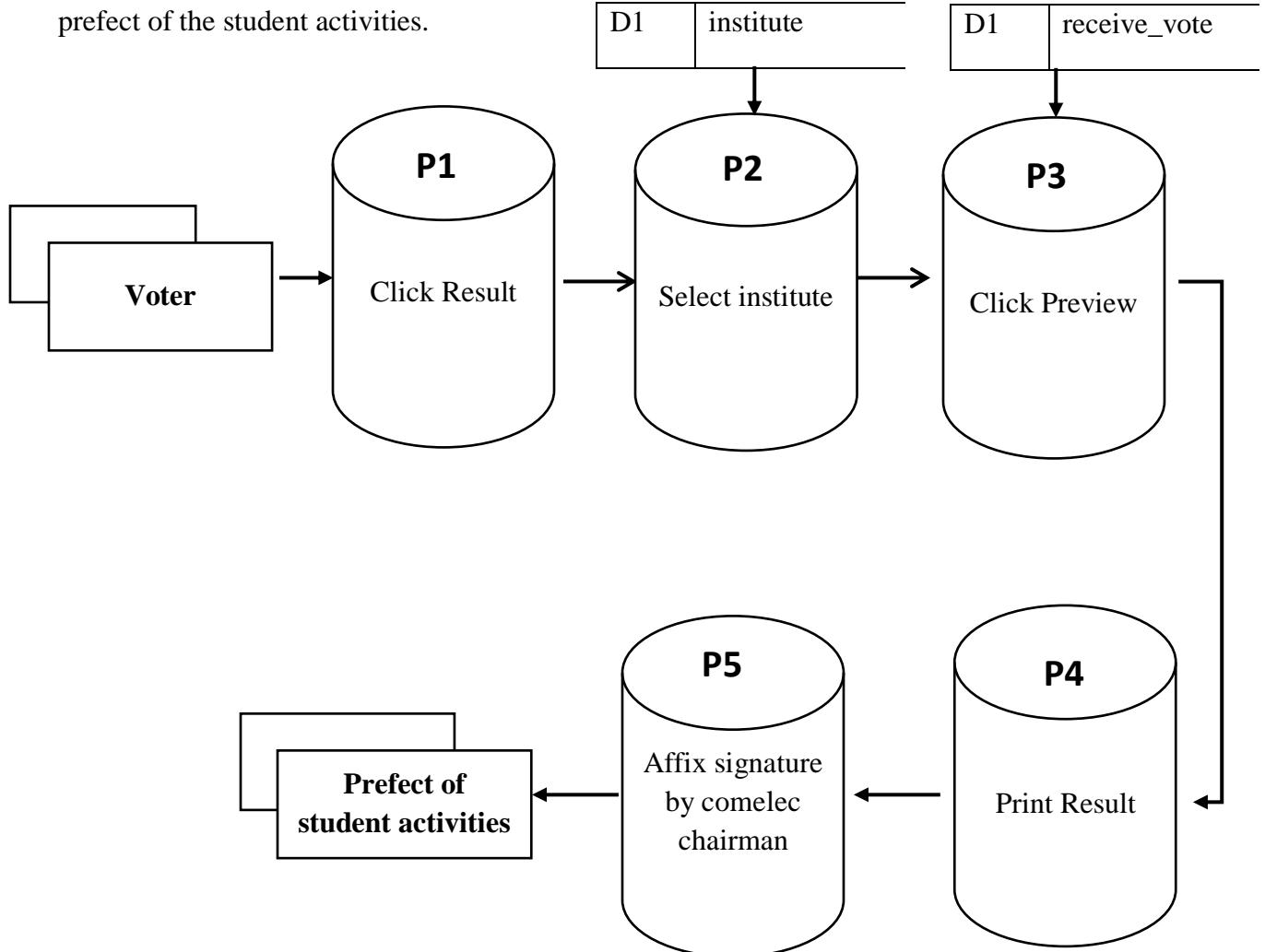


Figure 36: View and Print Result (Proposed)

View and print Voted List

Narrative Flow: The Administrator/Commissioner will click the Report button then select View Voted to view the list of voters who already voted. Then Browse and select election type and browse institute then click search and then print the report.

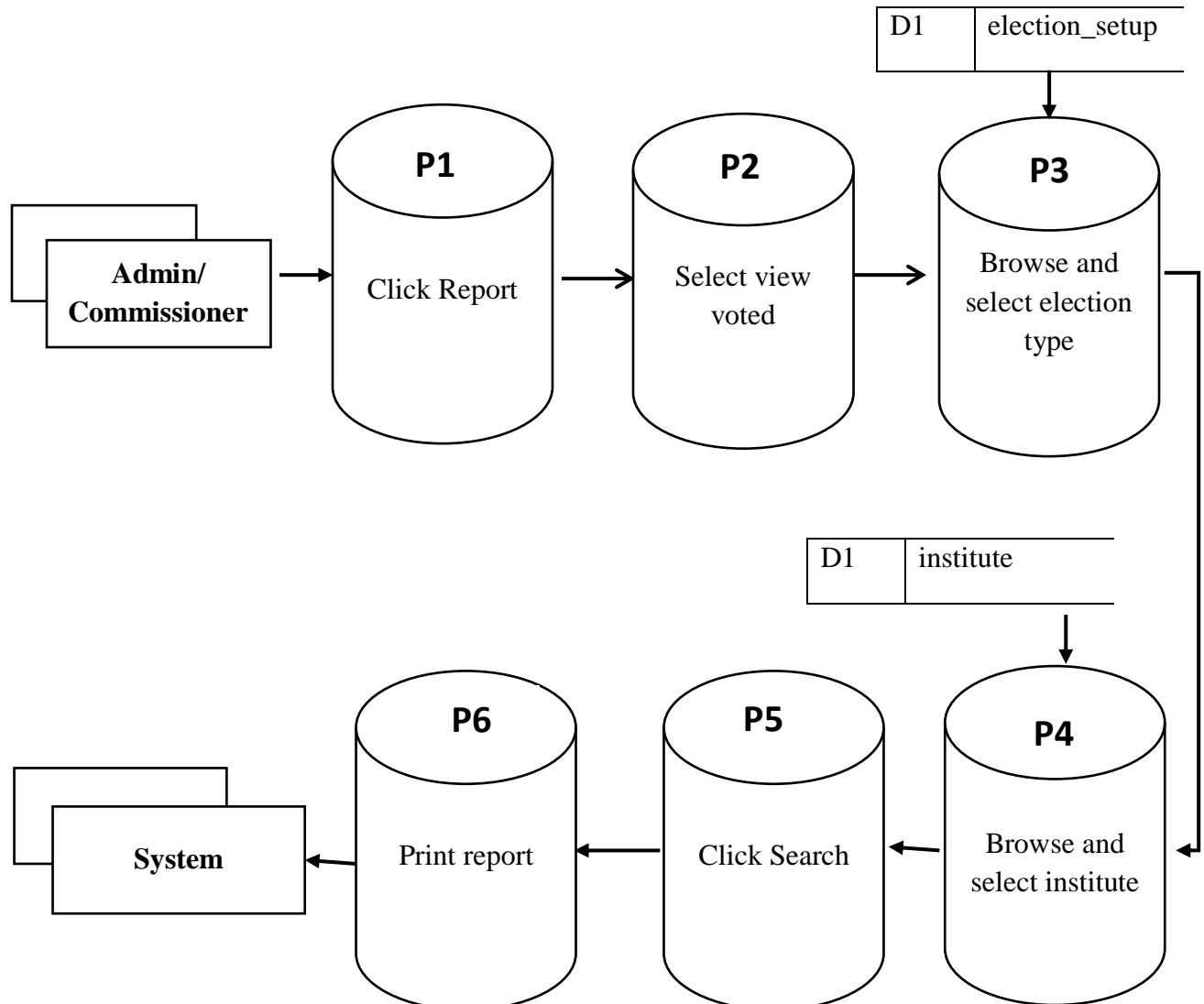


Figure 37: View and Print Voted List (Proposed)

View and print Not Voted List

Narrative Flow: The Administrator/Commissioner will click the Report button then select View Not Voted to view the list of voters who have not voted. Then Browse and election type and browse institute then click search and then print the report.

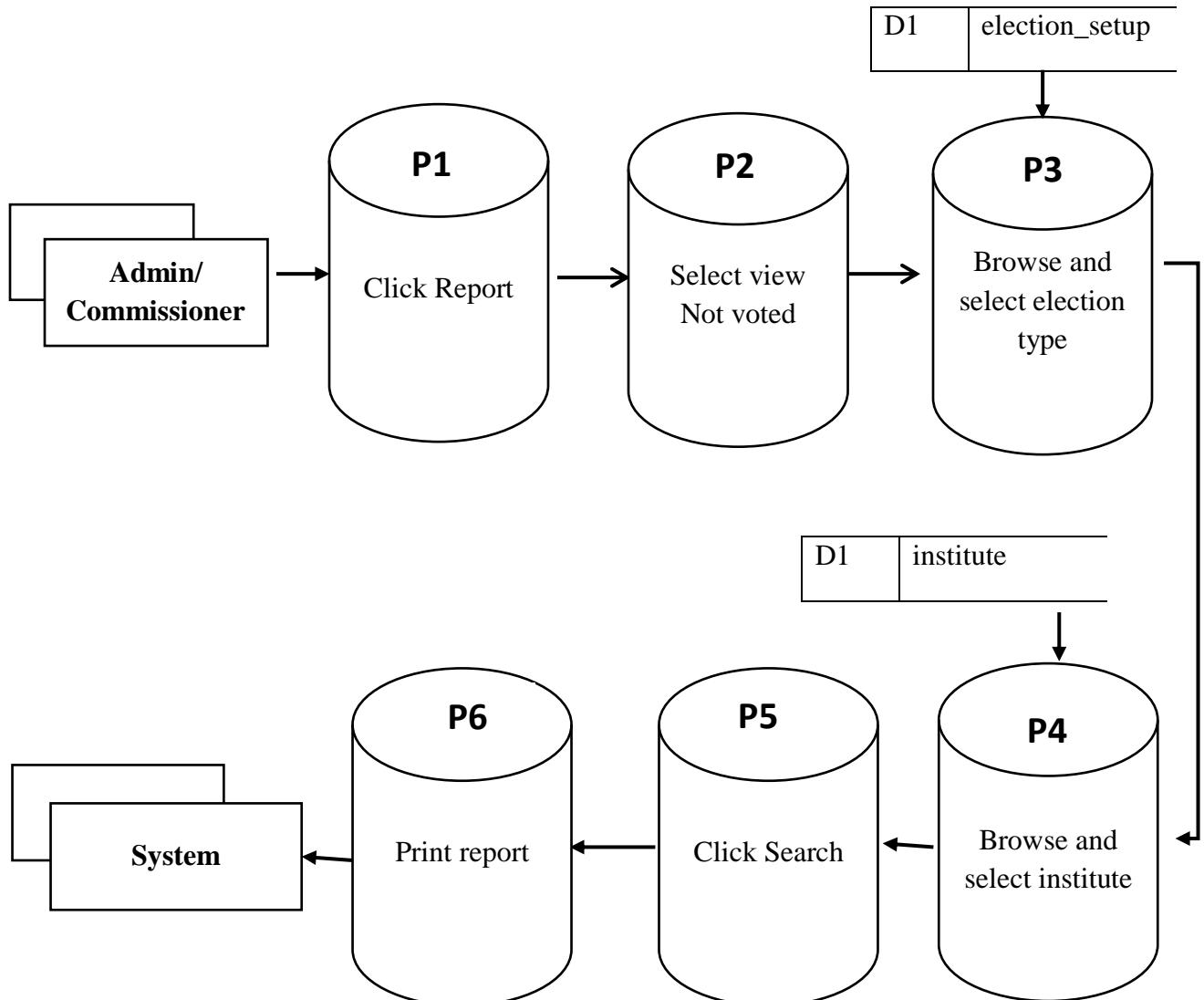


Figure 38: View and Print Not Voted List (Proposed)

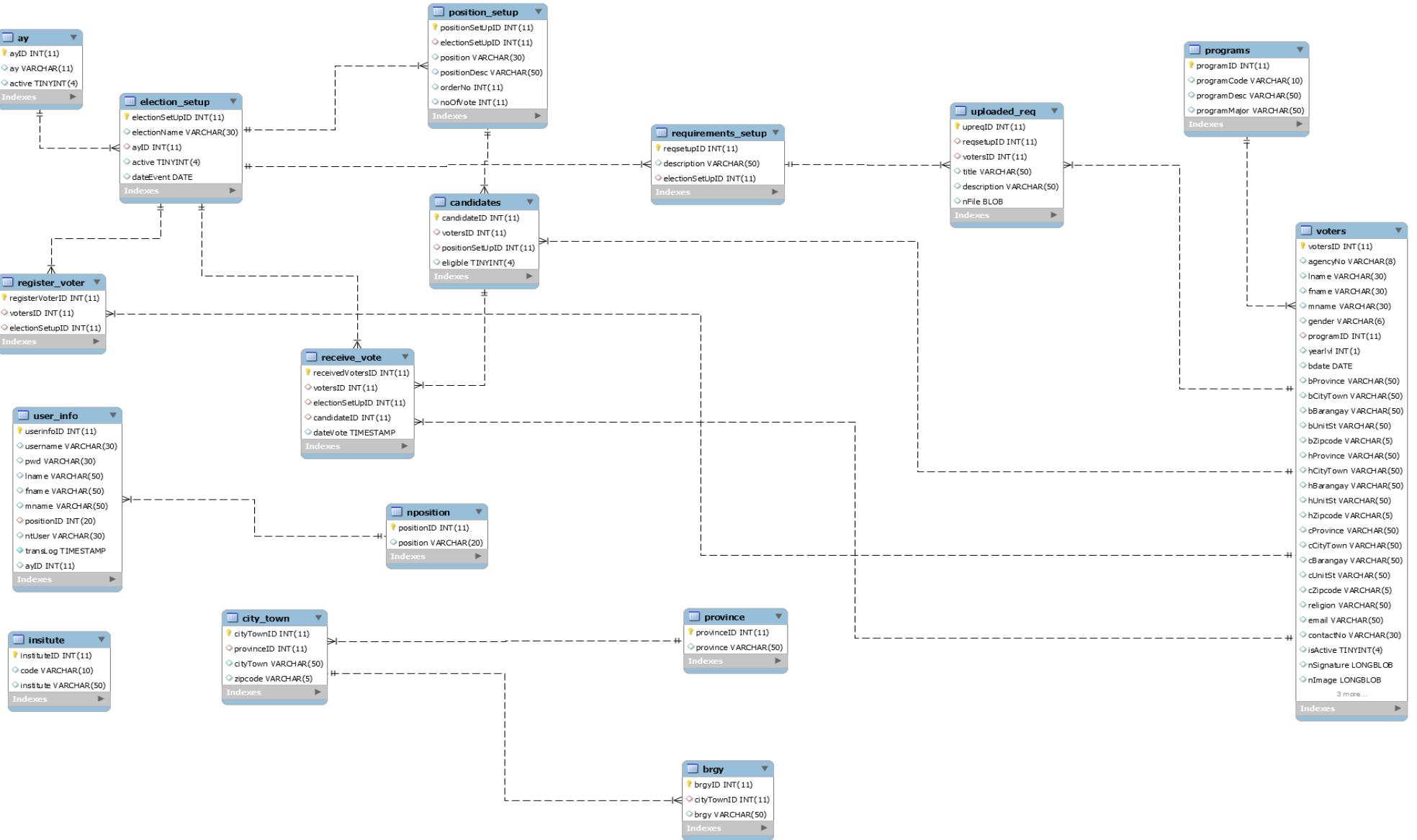


Figure 39: Entity Relationship Diagram (Proposed)

Database Structure

	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	On Update
	candidateID	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	votersID	int	11		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	positionSetUpID	int	11		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	eligible	tinyint	4	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
					<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Table 29: Candidate

	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	On Update
	electionSetUpID	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	electionName	varchar	30	''	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	ayID	int	11		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	active	tinyint	4	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	dateEvent	date			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
					<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Table 30: Election Setup

	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	On Update
	ayID	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	ay	varchar	11	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	active	tinyint	4	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
					<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Table 31: Academic Year

	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	On Update
	positionID	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	position	varchar	20	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
					<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Table 32: Position

	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	On Update
	reqsetupID	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	description	varchar	50		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	electionSetUpID	int	11		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
					<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Table 33: Register Setup

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	On Update
positionSetUpID	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
electionSetUpID	int	11		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
position	varchar	30	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
positionDesc	varchar	50	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
orderNo	int	11		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
noOfVote	int	11		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
alias	varchar	20	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Table 34: Position Setup

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	On Update
registerVoterID	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
votersID	int	11		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
electionSetupID	int	11		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Table 35: Register Voter

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	On Update
upreqID	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
reqsetupID	int	11		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
votersID	int	11		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
title	varchar	50	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
description	varchar	50	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
nFile	blob			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Table 36: Upload Requirements

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	On Update
brgyID	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
cityTownID	int	11		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
brgy	varchar	50	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Table 37: Barangay

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	On Update
cityTownID	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
provinceID	int	11		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
cityTown	varchar	50	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
zipcode	varchar	5	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Table 38: City Town

	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	On Update
	instituteID	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	code	varchar	10	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	institute	varchar	50	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
					<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Table 39: Institute

	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	On Update
	programID	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	programCode	varchar	10	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	programDesc	varchar	50	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	programMajor	varchar	50	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
					<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Table 40: Program

	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	On Update
	provinceID	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	province	varchar	50	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
					<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Table 41: Province

	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	On Update
	userinfoID	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	username	varchar	30	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	pwd	varchar	30	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	lname	varchar	50	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	fname	varchar	50	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	mname	varchar	50	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	positionID	int	20	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	ntUser	varchar	30	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	transLog	timestamp		CURRENT_TIMESTAMP	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	ayID	int	11	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
					<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Table 42: User information

	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	On Update	Comments
	votersID	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	agencyNo	varchar	8	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	lname	varchar	30	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	fname	varchar	30	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	mname	varchar	30	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	gender	varchar	6	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	programID	int	11		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	yearlvl	int	1		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	bdate	date			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	bProvince	varchar	50	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	bCityTown	varchar	50	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	bBarangay	varchar	50	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	bUnitSt	varchar	50	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	bZipcode	varchar	5	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	hProvince	varchar	50	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	hCityTown	varchar	50	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	hBarangay	varchar	50	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	hUnitSt	varchar	50	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	hZipcode	varchar	5	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	cProvince	varchar	50	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	cCityTown	varchar	50	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	cBarangay	varchar	50	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	cUnitSt	varchar	50	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	cZipcode	varchar	5	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	religion	varchar	50	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	email	varchar	50	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	contactNo	varchar	30	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	isActive	tinyint	4	1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	nSignature	longblob			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	nImage	longblob			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	nFingerPrint	longblob			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	user	varchar	30	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	pwd	varchar	30	''	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
					<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Table 43: Voters

	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	On Update	Comments
	receivedVoters	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	electionSetUpID	int	11		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	candidateID	int	11		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	positionSetUpID	int	11		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	dateVotte	date			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
					<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Table 44: Receive Vote

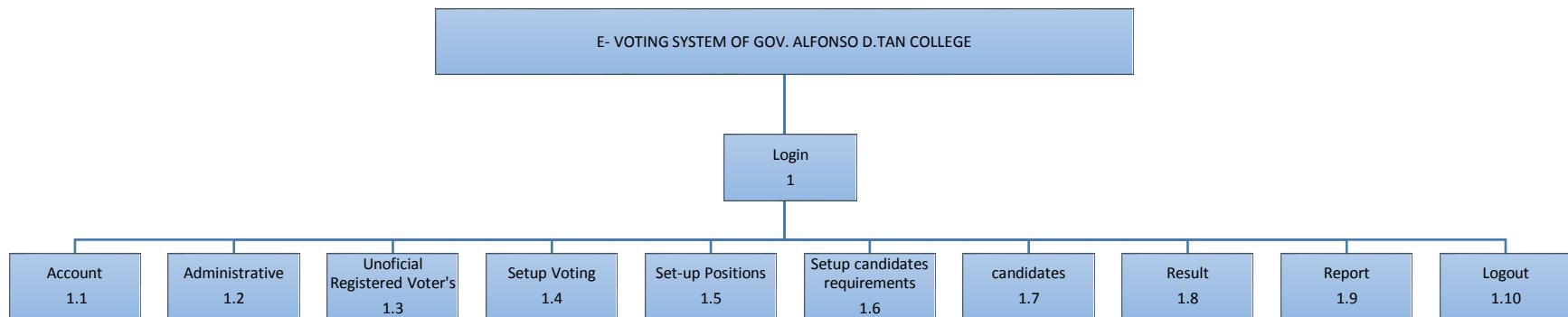


Figure 40: Hierarchical Input Process Output (System administered by the admins)

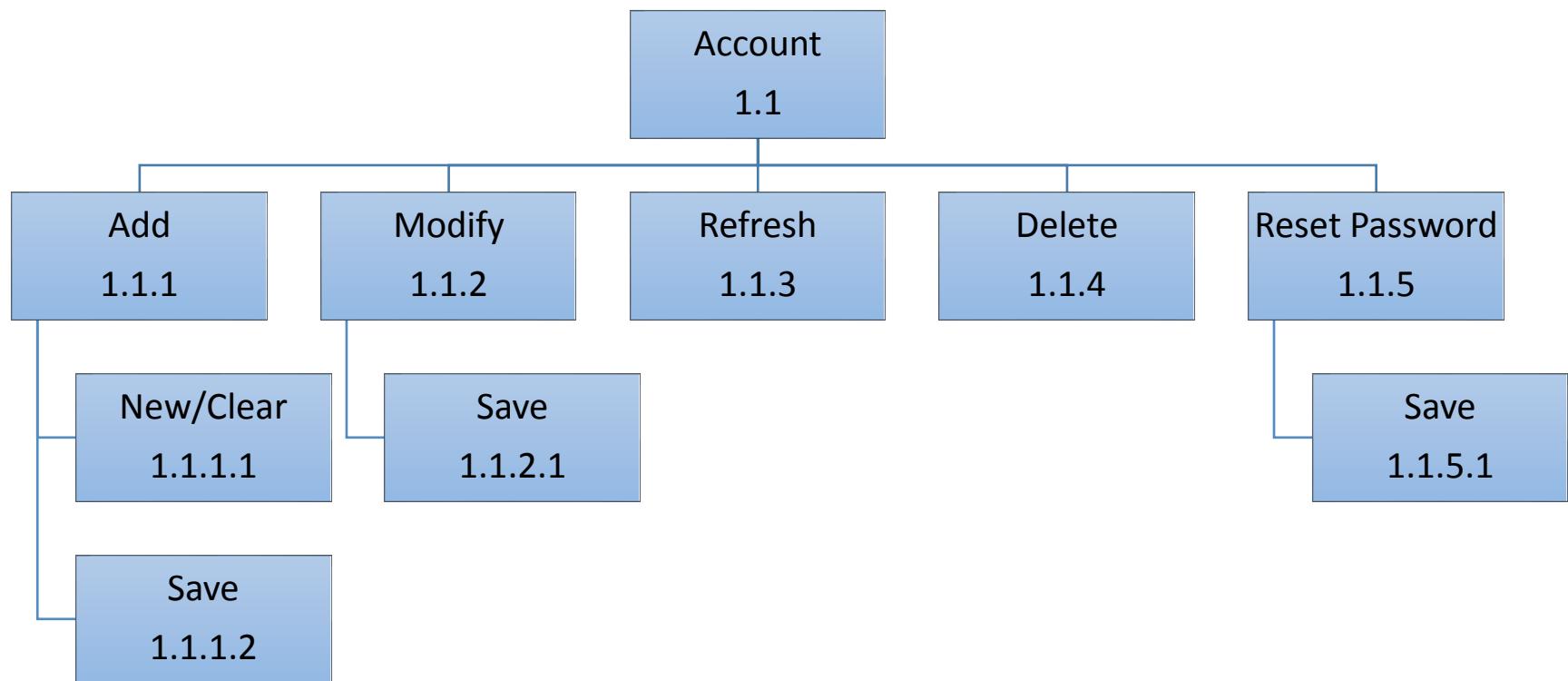


Figure 40: Hierarchical Input Process Output (System administered by the admins)

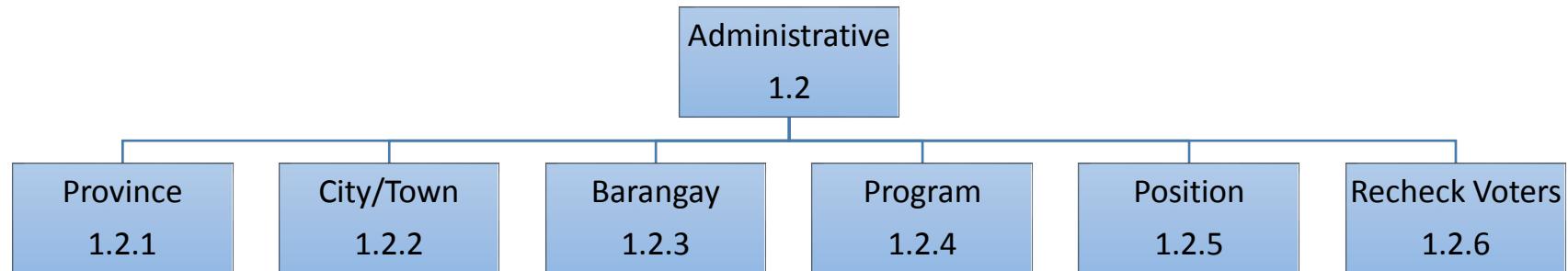


Figure 40: Hierarchical Input Process Output (System administered by the admins)

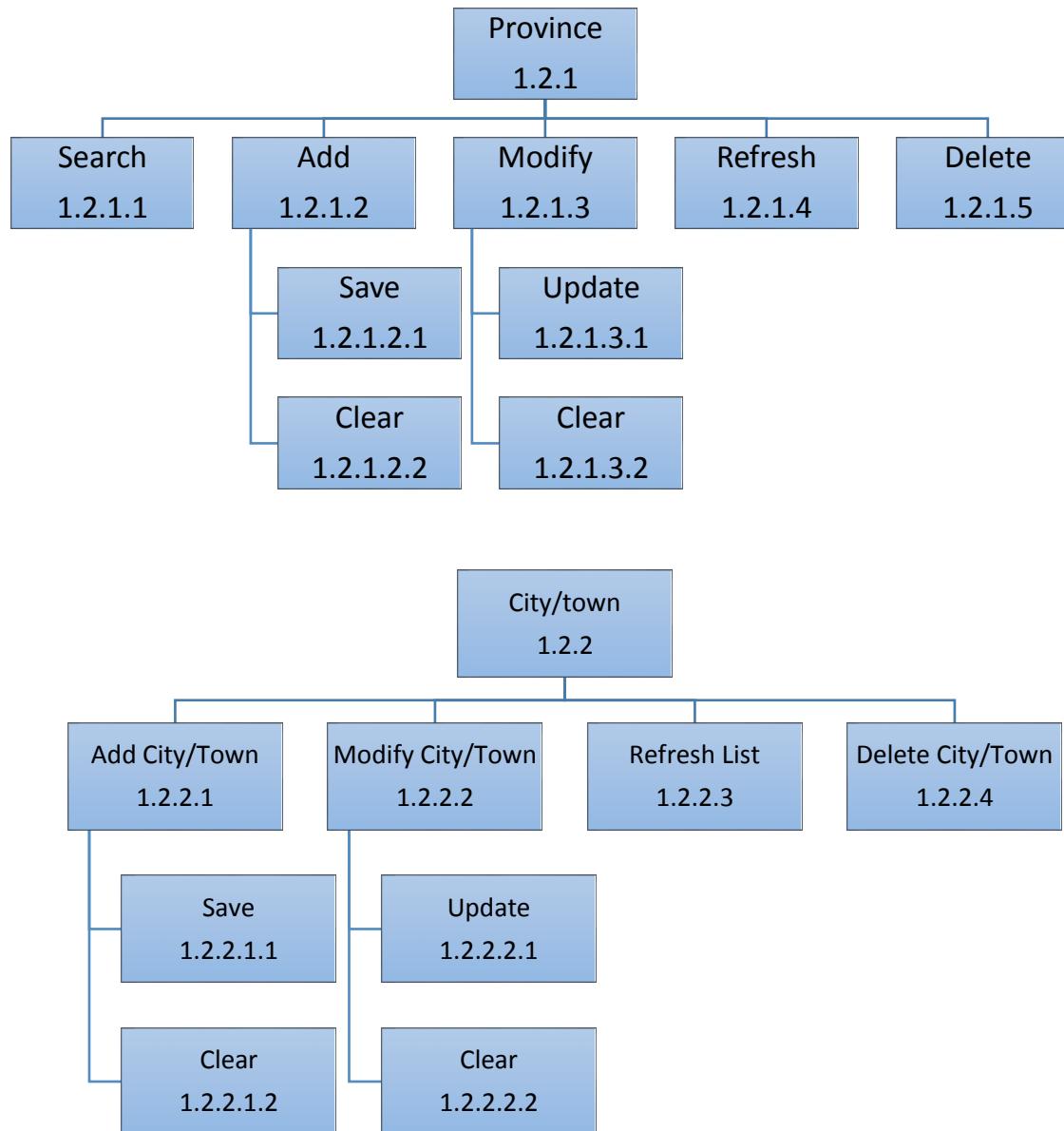


Figure 40: Hierarchical Input Process Output (System administered by the admins)

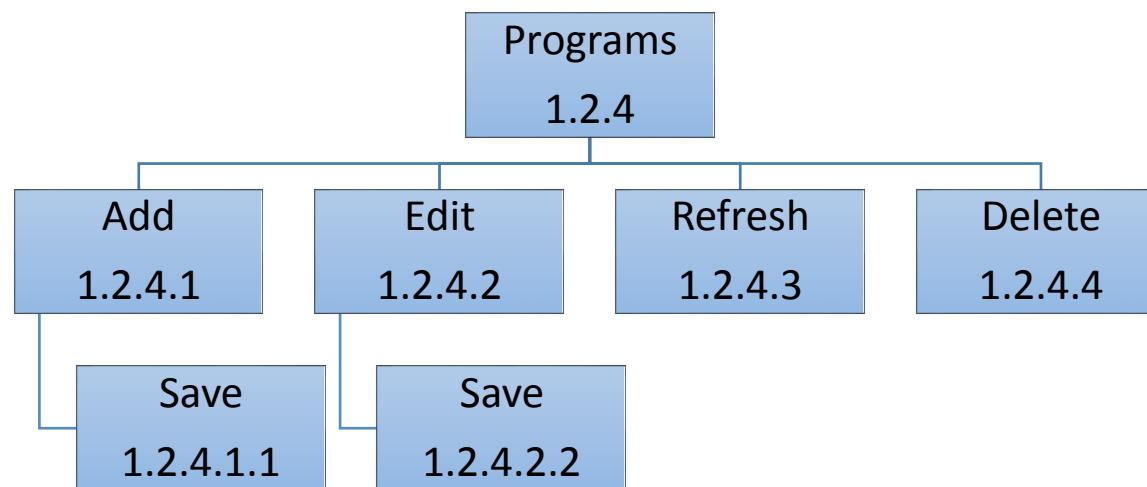
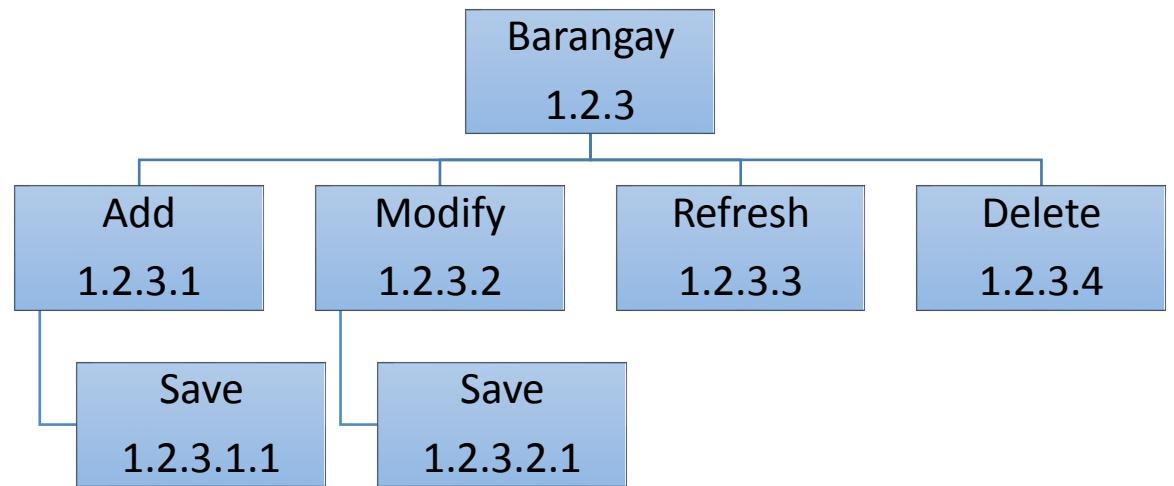


Figure 40: Hierarchical Input Process Output (System administered by the admins)

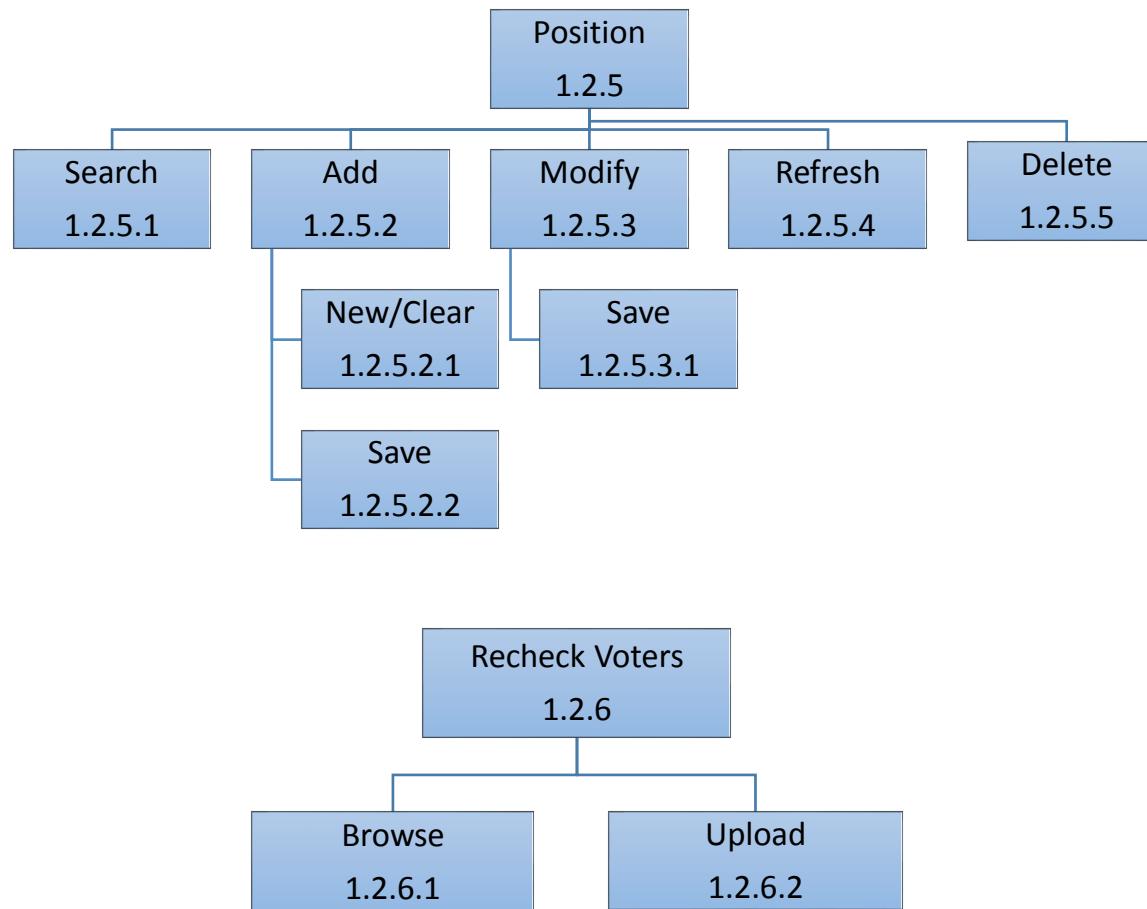


Figure 40: Hierarchical Input Process Output (System administered by the admins)

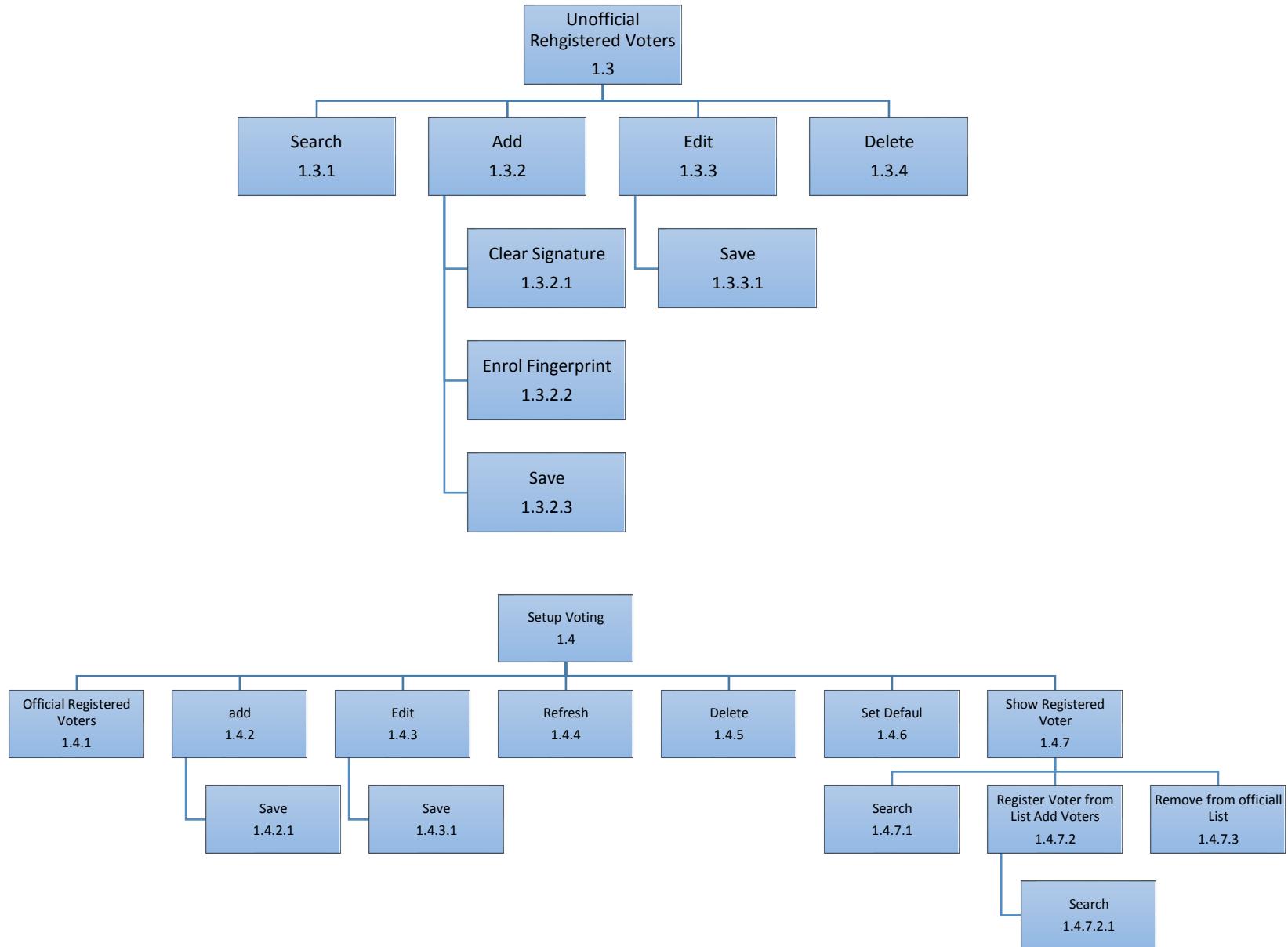


Figure 40: Hierarchical Input Process Output (System administered by the admins)

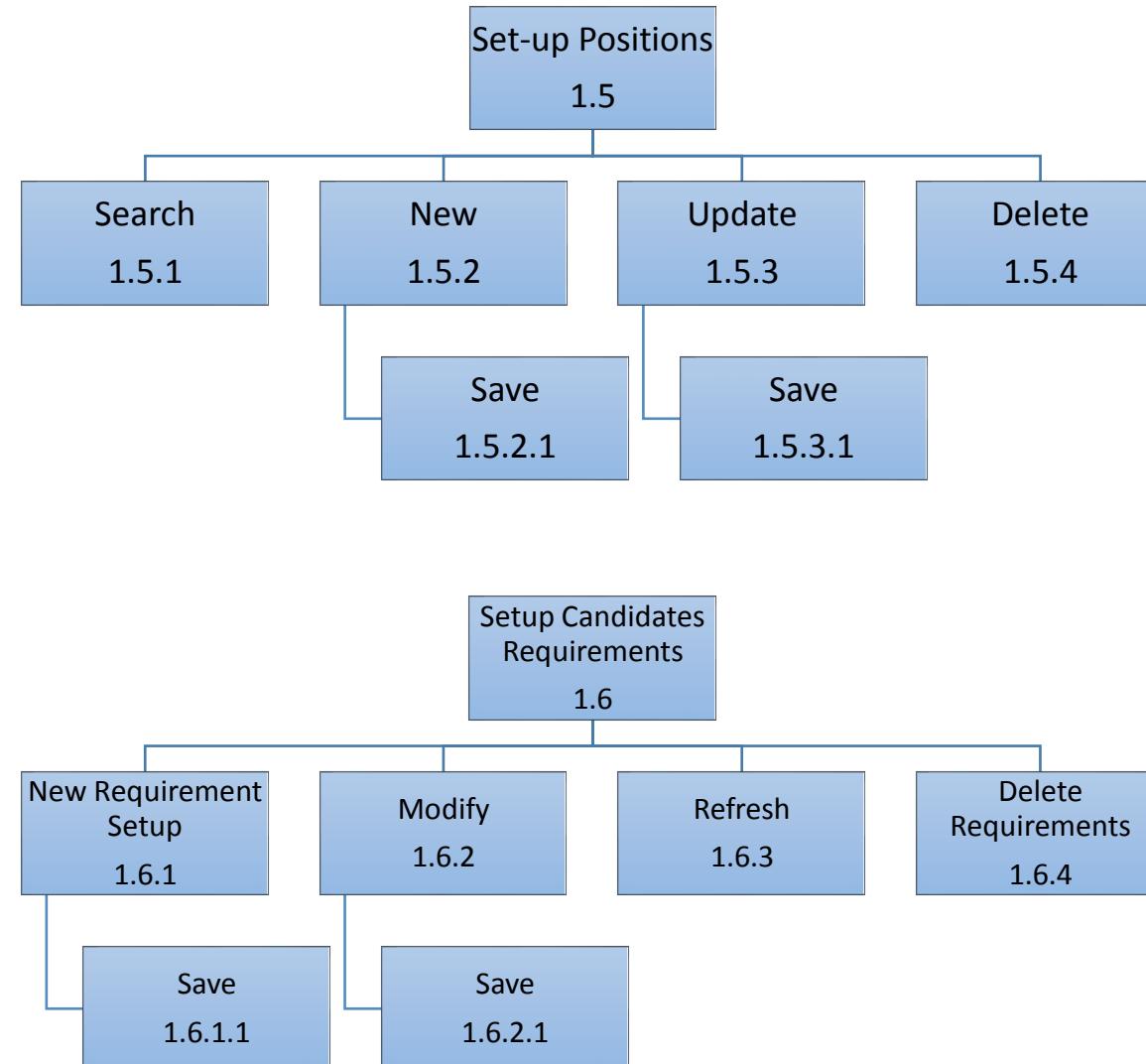


Figure 40: Hierarchical Input Process Output (System administered by the admins)

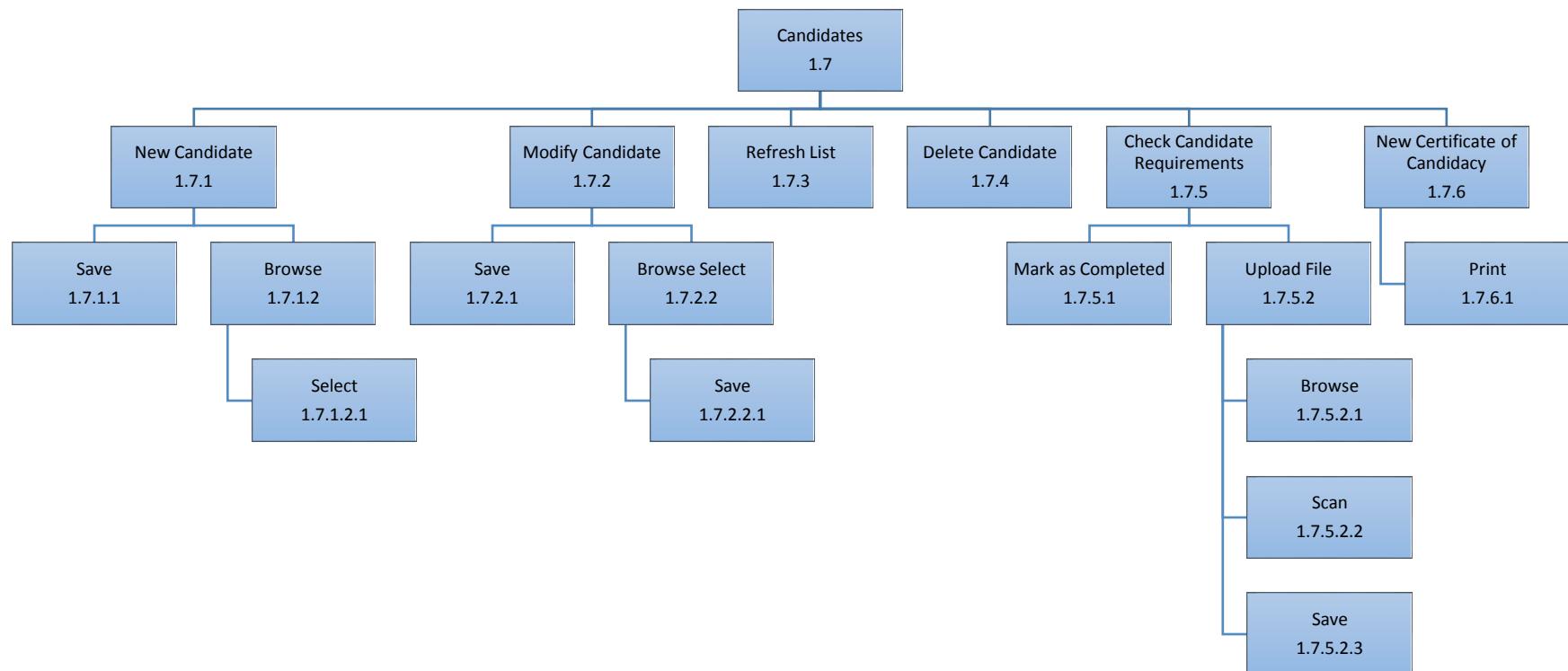


Figure 40: Hierarchical Input Process Output (System administered by the admins)

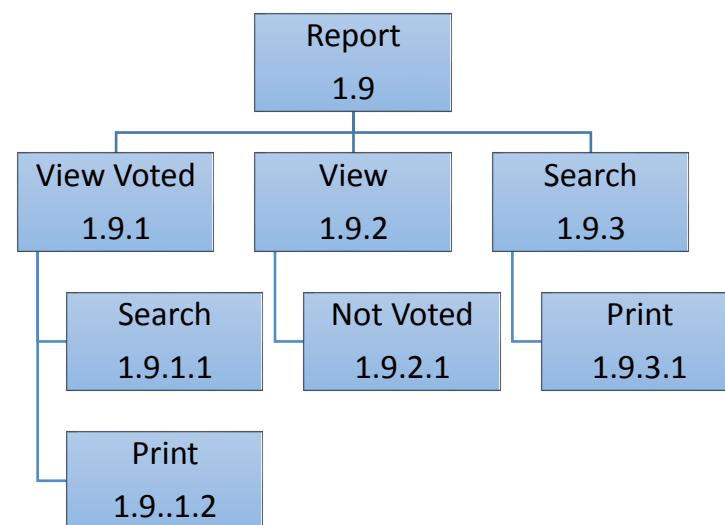
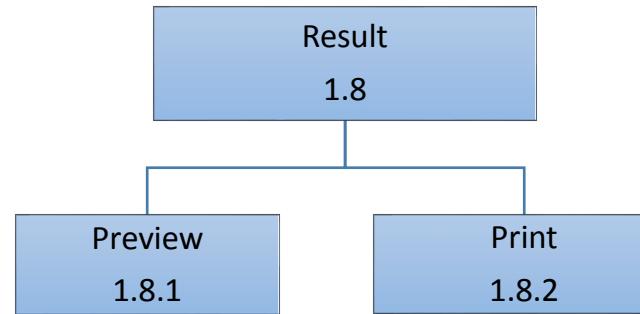


Figure 40: Hierarchical Input Process Output (System administered by the admins)

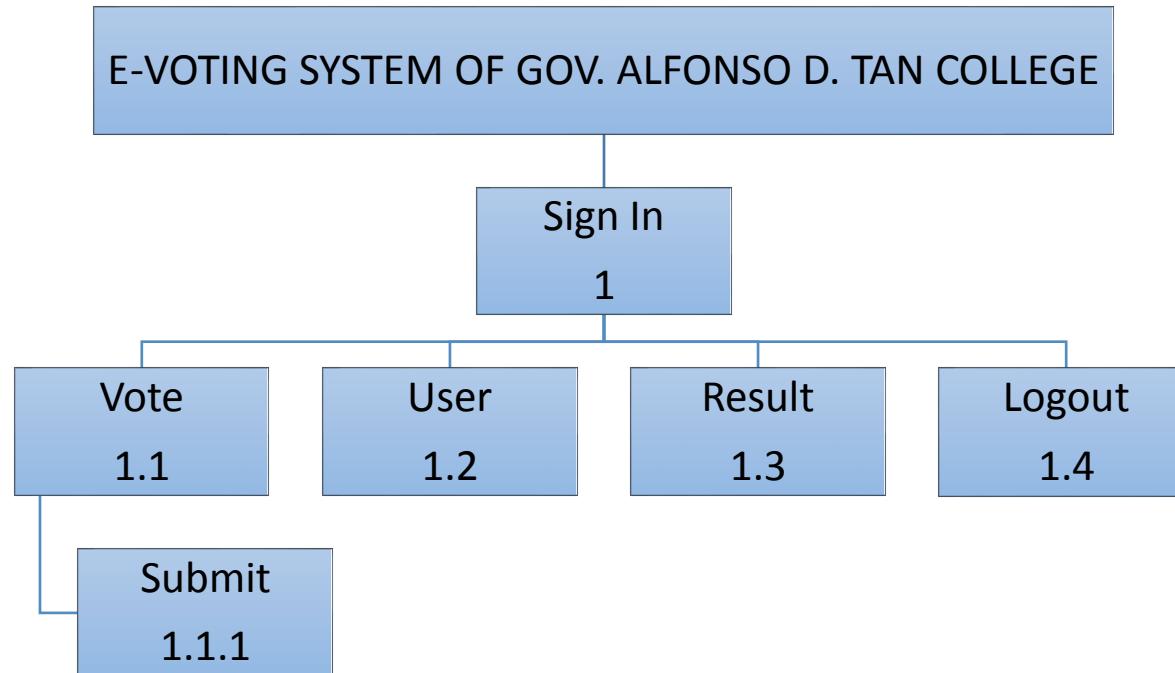


Figure 41: Hierarchical Input Process Output (Voting Process)

Technical Background

Electronic Voting System (EVS) also known as E-voting is a term encompassing election process through mobile application and web hosting embracing electronic means of counting votes.

We utilize photo capturing device, fingerprint management device and electronic signature in registration of the voters and filing of candidacy of the candidates. It needs also at least an Epson Scanner in scanning the requirements of the candidates in filing of candidacy. The designed application does support smartphones and capable of web browsing through wifi hotspot or any computer connected to the Local Area Network which means casting votes is done via website and sends the casted vote to the centralized database. The browser can also be used for viewing of the results.

Implementation of this system uses C# for administrator and MYSQL database. C# is a general-purpose, multi-paradigm programming language encompassing strong typing, lexically scoped, imperative, declarative, functional, generic, programming disciplines. C# is one of the programming languages designed for the common language infrastructure. MYSQL will be used as the back-end database because of its open-source relational database management system that has a big capacity in storing data. A browser will let the user to visit web pages and use web applications. The website can be visited by any browser such as Google Chrome, Opera, Opera Mini, Firefox Browser, Firefox Focus and Microsoft Edge and many more. Furthermore, proposed system must be installed on Windows 7 or higher version operating system.

Chapter 4

RESULTS AND DISCUSSIONS

Results

E-Voting System of Gov. Alfonso D. Tan College is a web-based system that allows the voters to cast vote for a particular candidates using their smartphones or computers. Based on the gathered data from survey in manual voting process, it was found out that the respondents have encountered problems during voting period. It was pointed out that miscounted ballots, vote lost through unclear or invalid marks and a long queuing in voting was the major problem the respondents had ever experienced. It was revealed further that using electronic voting system is more comfortable than paper based voting and it is more interesting than traditional voting system used in GADTC since it can reduce a number of election committee on voting process and can cater any voting type. Moreover, the respondents involved in the process agreed that E-Voting System can replace the inefficient most importantly error prone manual voting and that having this system in the institution would be very timely relevant.

Discussion

In today's generation, everything sets involved with technology. Computers and Mobile phones serve as an important medium in our society, most especially in school premises. It is now marked as one of the leading innovations that educational institution should have. We come up with a study "E-Voting System of Gov. Alfonso D. Tan College" that can provide better satisfaction and convenience to students and also for those election committee who facilitates during election period. This provide faster and accurate counting of votes and eliminates the paper works, proper record handling and time consumed process. To put the study into reality, the researchers chose two hundred (200) respondents in the institution. The respondents were randomly selected with a minimum of students, 28 students per institute. An interview was conducted and a standardized questionnaire was also distributed to the said respondents in order to gather data. The researchers also assumed that the respondents could raise honestly their concerns.

Chapter 5

CONCLUSION AND RECOMMENDATIONS

Conclusion

Based on the result of the study, the researchers concluded that having E-Voting System in GADTC is beneficial in order to resolve problems like long queuing, miscounted ballots, vote lost through unclear or invalid marks during election period. Having this system will make the voting process easier and faster. It will also reduce human effort and can announce the result within a short period of time. Since the cast votes of the voters will be saved to the database and it will then be calculated automatically by the system. Thus, it will always be accurate, secured and available. Indeed, this will greatly make voting process speedy hence, both students and election committee could save more time.

Recommendation

Based on the result of the study, the proponents recommend the following:

1. E-Voting System can be visited online.
2. E-Voting System will register ten (10) fingerprints.
3. E-Voting System will be implemented to smartphones through Android Application. So that voters can login through their fingerprints.
4. Awareness and educating the voters on voting process and its important and relevance to come out with a very accurate and relevant result.

Appendix A

E-Voting System of Gov. Alfonso D. Tan College User's Manual

Admin and Commissioner side

1. Getting Started

To use the E-Voting System of Gov. Alfonso D. Tan College, you need to Login first and input your Username and Password. Then Click “Login” button.

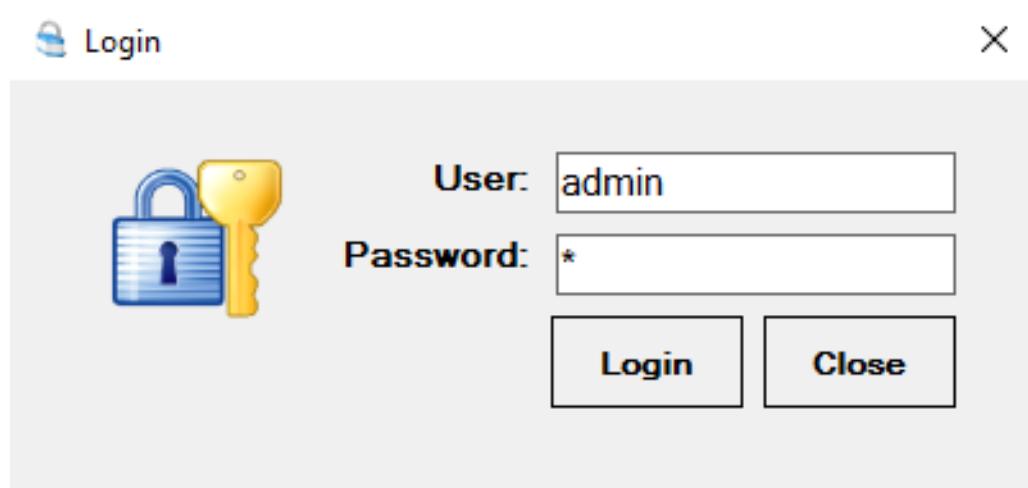


Figure 42: Login

Once you've Login this will be shown:



Figure 43: Homepage

2. Add User's Account

You can add Users through clicking the “Account” button.

NOTE: The Registered administrator can only Add User's Account.

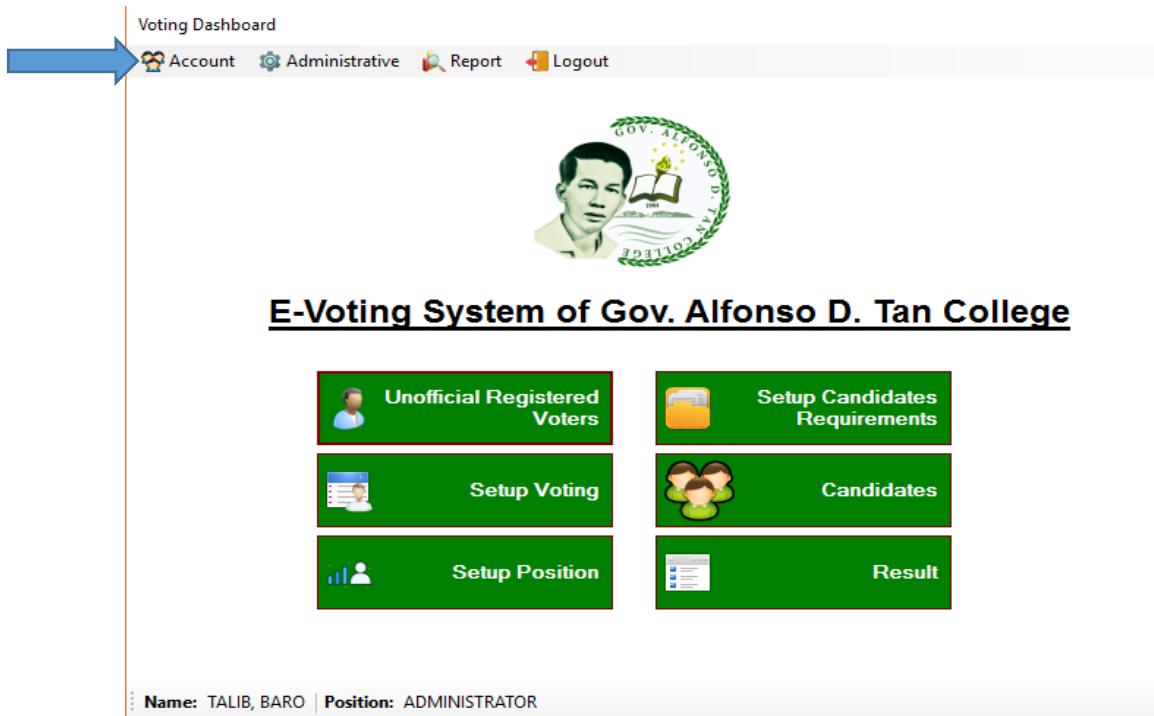


Figure 44: Add User's Account

- The following page will load and the User's Account information will display as shown below then click “Add” button.

User ID	Username	Lastname	Firstname	Middle name	Position
6	admin	TALIB	BARO		ADMINISTRATOR
8	anchie	JAMISOLA	ANNCHIE	CEMAFRANCA	COMMISSIONER
9	joan	DOCY	JOAN	LACIA	COMMISSIONER
10	godfrey	YAMA	GODFREY	MARLO	COMMISSIONER

Figure 45: User's Account List

- Once you've click the "Add" button this will be shown and you must fill out all the necessary data then click "Save" button.

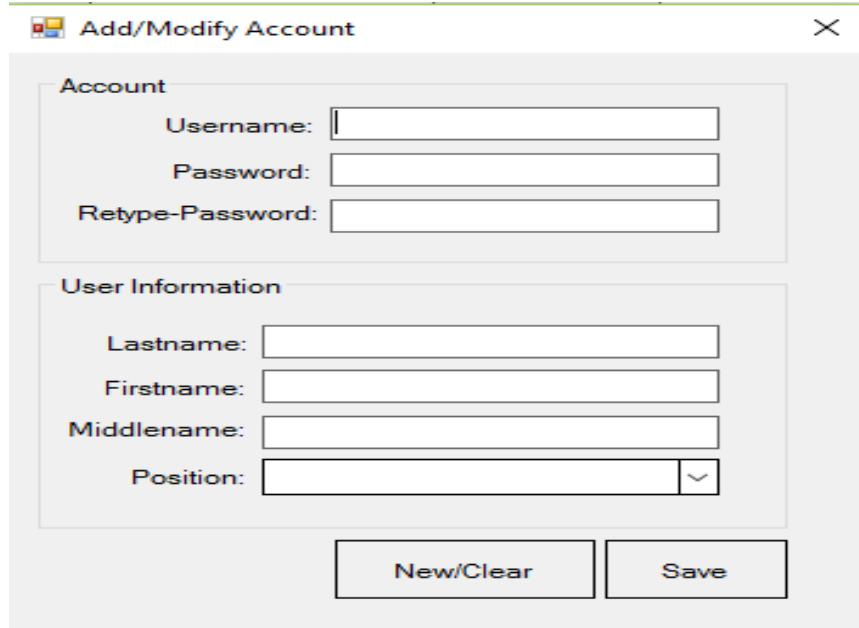


Figure 46: User's Account Information

- It will be automatically saved. The dialog box will appear displaying message that notifying you the success.

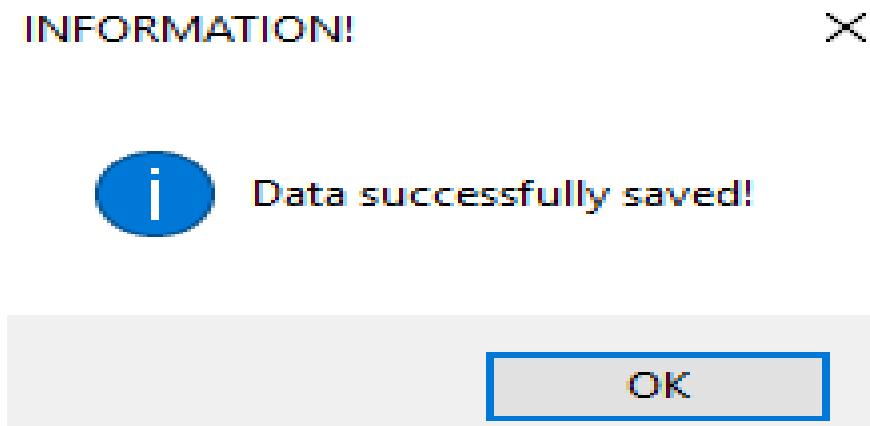


Figure 47: User's Account Information Successfully Saved

3. Add Province

You can add Province through clicking the “Administrative” button. Then select “Province” that shown below.

NOTE: Both Administrator and Commissioner can add Province.

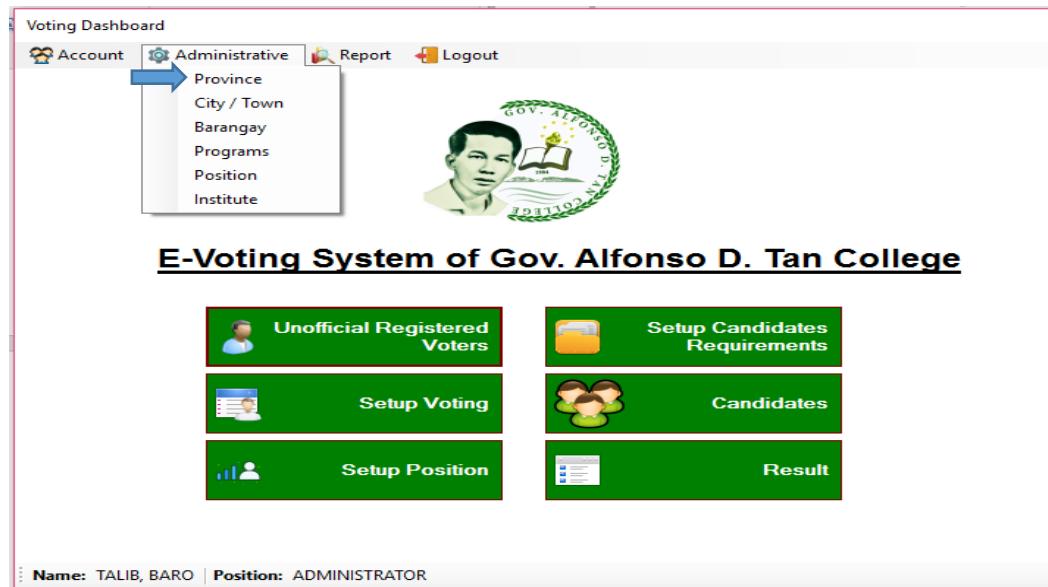


Figure 48: Province

- The Province Main form will display then you must right click to select “Add” button as shown below.

ID	Province
32	DAVAO
2	LANAO DEL NORTE
5	MANILA
3	MISAMIS OCCIDENTAL
4	MISAMIS ORIENTAL
34	TEST
33	ZAMBOANGA DEL SUR

Figure 49: Province List

- Once you've click the "Add" button this will be shown and you must input Province Name then click "Save" button.

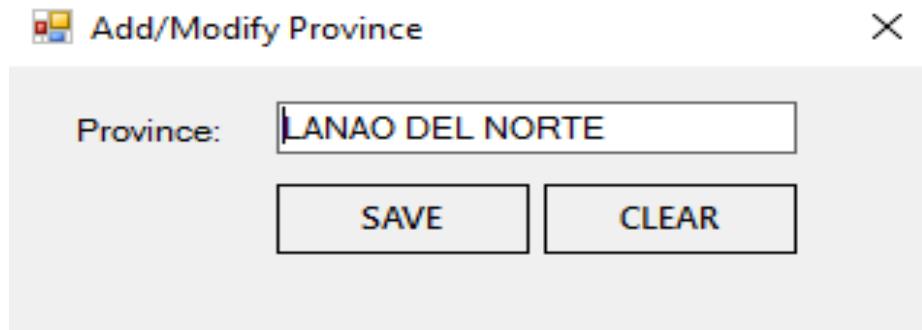


Figure 50: Province Name

- It will be automatically saved. The dialog box will appear displaying message that notifying you the success.

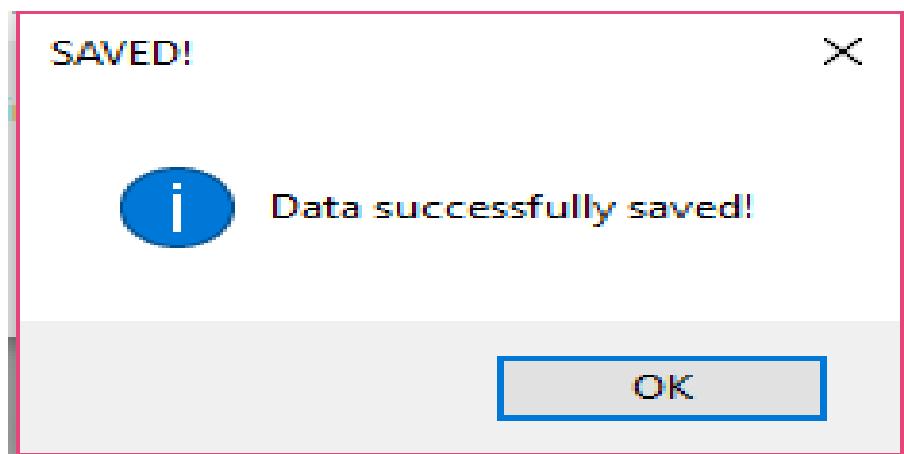


Figure 51: Province Name Successfully Saved

4. Add City/Town

You can add City/Town through clicking the “Administrative” button. Then select “City/Town” that shown below.

NOTE: Both Administrator and Commissioner can add City/Town.

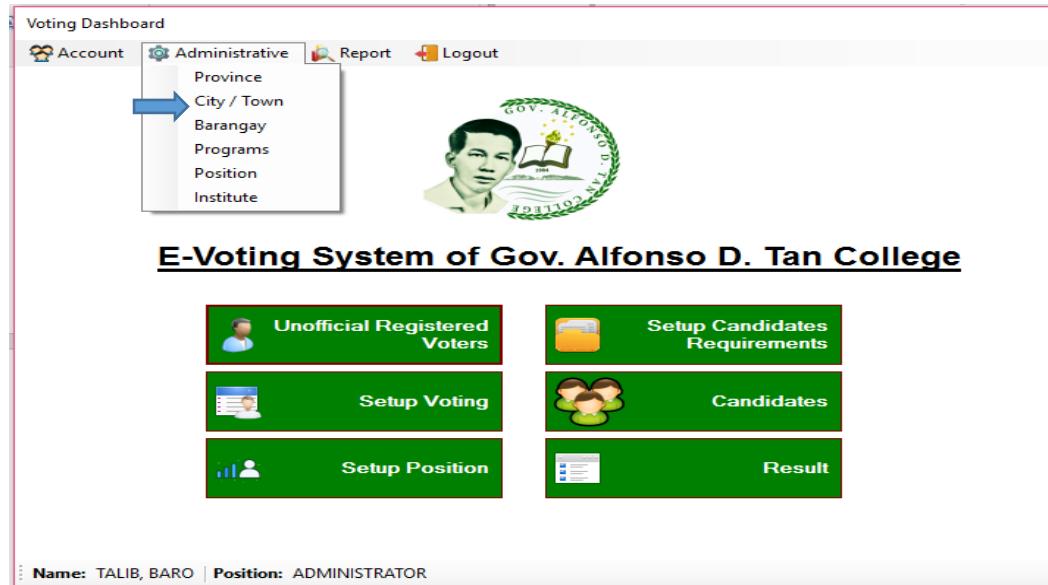


Figure 52: City/Town

- The City/Town Main form will display then you must right click to select “Add City/Town” button as shown below.

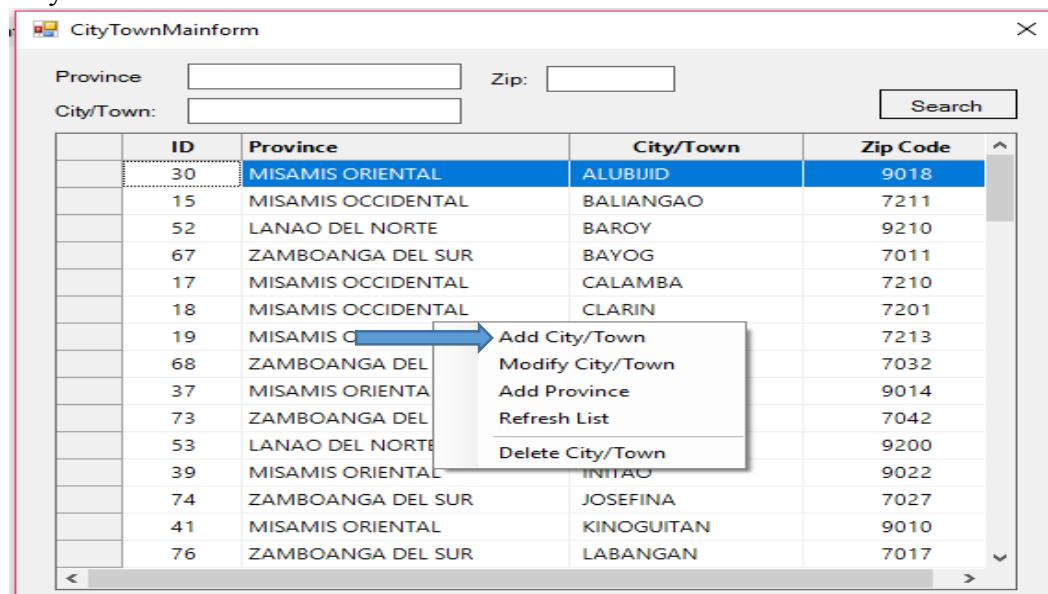


Figure 53: City/Town List

- Once you've click the "Add City/Town" button this will be shown and you must browse Province where the City/Town belong and input City/Town Name then click "Save" button.

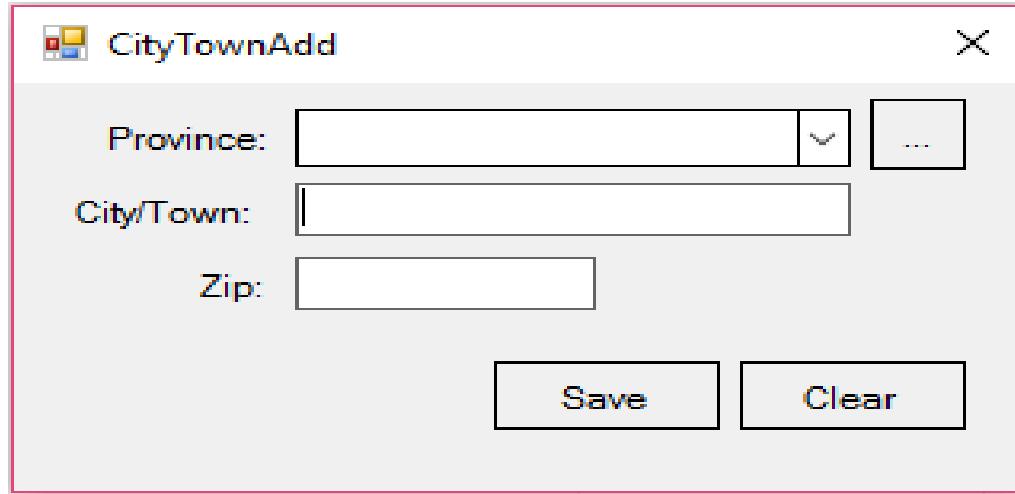


Figure 54: City/Town Name

- It will be automatically saved. The dialog box will appear displaying message that notifying you the success.

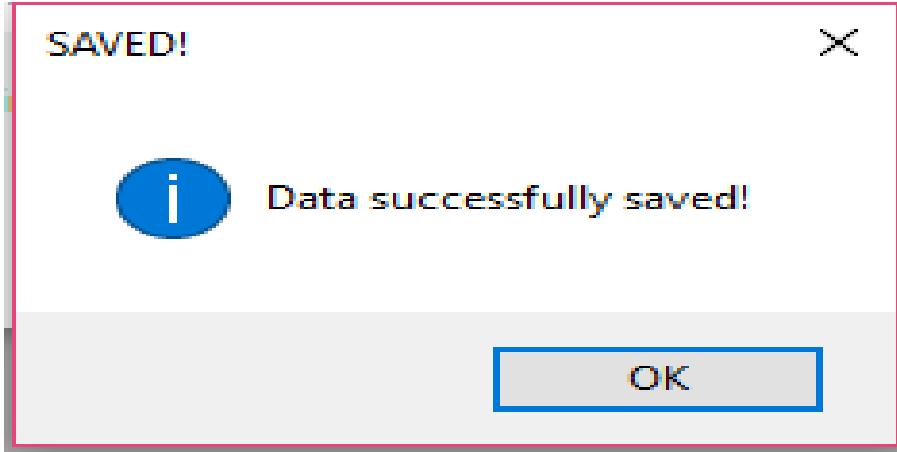


Figure 55 : City/Town Name Successfully Saved

5. Add Barangay

You can add Barangay through clicking the “Administrative” button. Then select “Barangay” that shown below.

NOTE: Both Administrator and Commissioner can add Barangay.

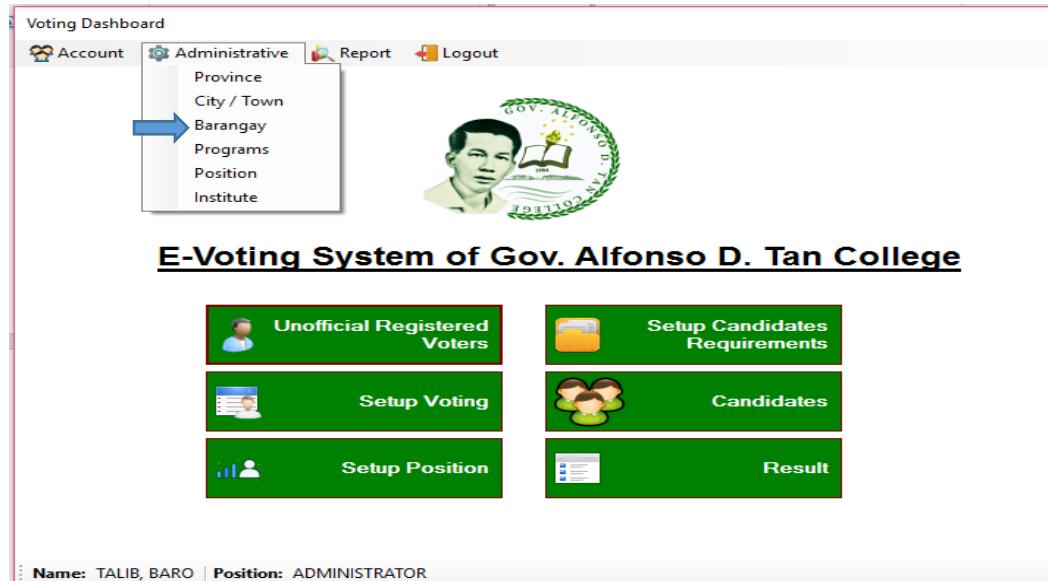


Figure 56: City/Town

- The Barangay Main form will display then you must right click to select “Add” button as shown below.

The screenshot shows the 'BarangayMainform' window. It has a search bar at the top with the placeholder 'Barangay:' and a 'Search' button. Below the search bar is a table with columns: ID, City/Town, Zip Code, and Barangay. The table contains the following data:

	ID	City/Town	Zip Code	Barangay
	1	BAROY	9210	POBLACION
	2	BAROY	9210	BAROY DACU
	3	TANGUB CITY	7214	MALORO
	6	TANGUB CITY	7214	MIGCANAWAY
	7	TANGUB CITY	7214	BARANGAY 1
	8	LALA	9211	MARANDING
	9	KOLAMBUGAN	9207	RIVERSIDE

A blue arrow points from the bottom right towards a context menu that appears over the table. The context menu includes 'Add', 'Modify', 'Refresh', and 'Delete' options.

Figure 57: Barangay List

- Once you've click the "Add" button this will be shown and you must browse Province and City/Town where the Barangay belong and input barangay Name then click "Save" button.

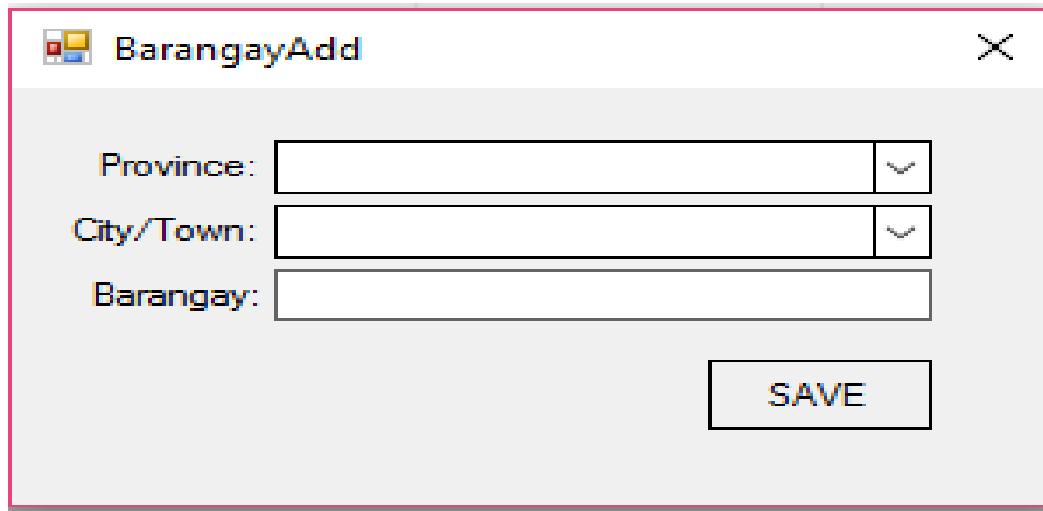


Figure 58: City/Town Name

- It will be automatically saved. The dialog box will appear displaying message that notifying you the success.

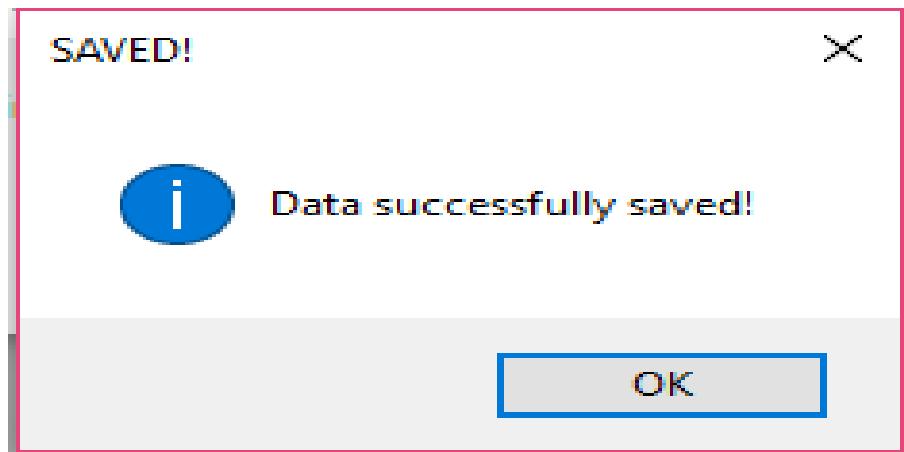


Figure 59: Barangay Name Successfully Saved

6. Add Institute

You can add Institute through clicking the “Administrative” button. Then select “Institute” that shown below.

NOTE: Both Administrator and Commissioner can add Institute.



Figure 60: Institute

- The Institute Main form will display then you must click to select “New” button as shown below.

The screenshot shows the 'InstituteMainform' window. At the top, there's a search bar with 'Search:' and a 'Search' button. Below the search bar is a table with three columns: 'ID', 'Institute Code', and 'Description'. The table contains four rows of data:

ID	Institute Code	Description
4	IAS	INSTITUTE OF ARTS AND SCIENCES
2	ICJE	INSTITUTE OF CRIMINAL JUSTICE EDUCATION
1	ICS	INSTITUTE OF COMPUTER STUDIES
3	IOM	INSTITUTE OF MIDWIFERY

At the bottom of the form, there are three buttons: 'New' (highlighted with a blue arrow), 'Modify', and 'Delete'.

Figure 61: Institute List

- Once you've click the "New" button this will be shown and you must input Institute Code and description then click "Save" button.

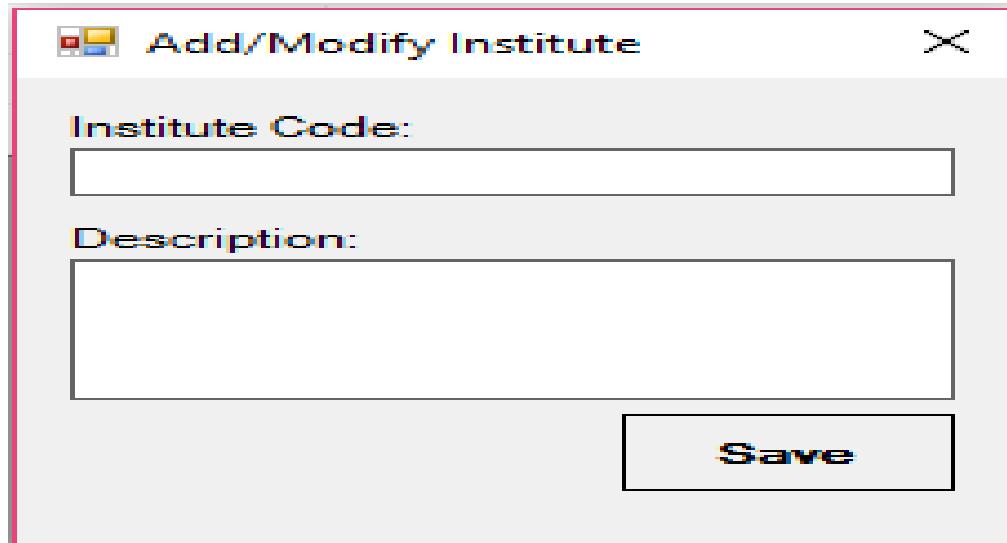


Figure 62: Institute Name

- It will be automatically saved. The dialog box will appear displaying message that notifying you the success.

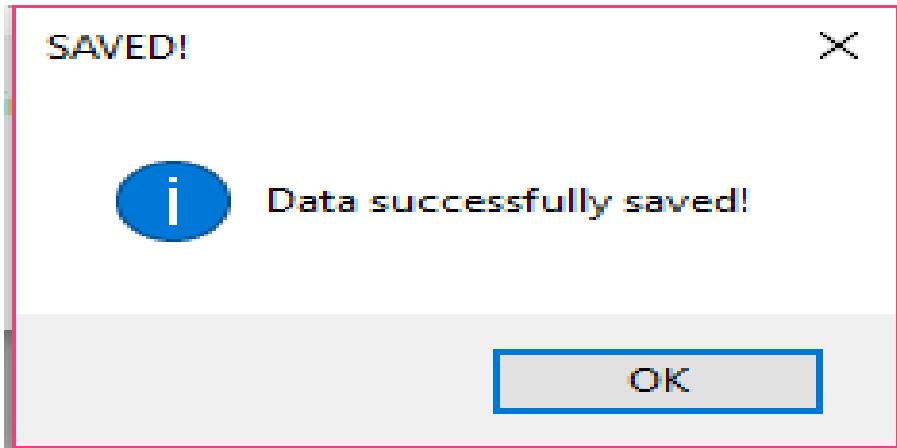


Figure 63: Province Name Successfully Saved

7. Add Programs

You can add Programs through clicking the “Administrative” button. Then select “Programs” that shown below.

NOTE: Both Administrator and Commissioner can add Programs.

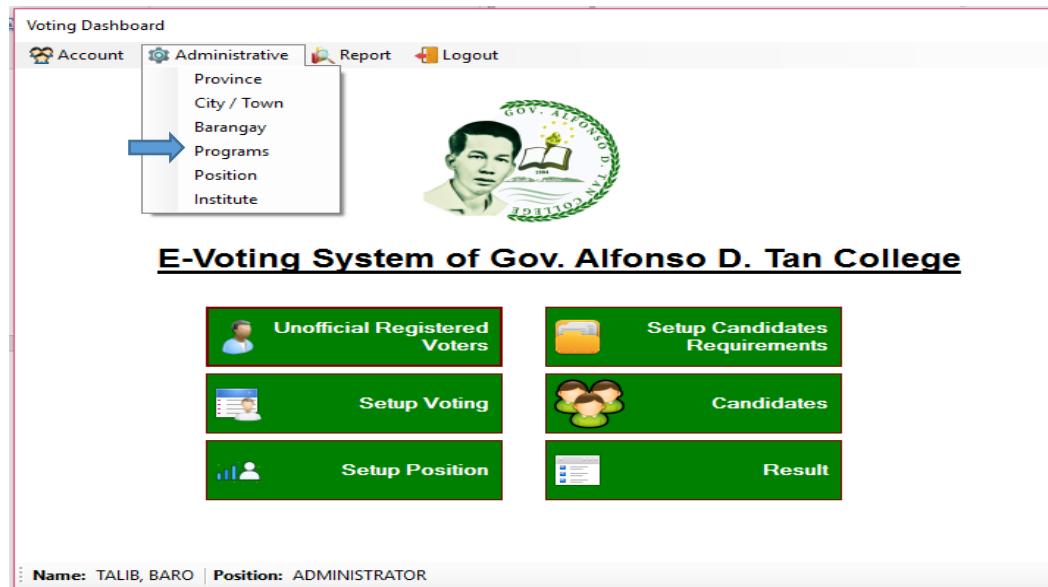


Figure 64: programs

- The Programs Main form will display then you must right click to select “Add” button as shown below.

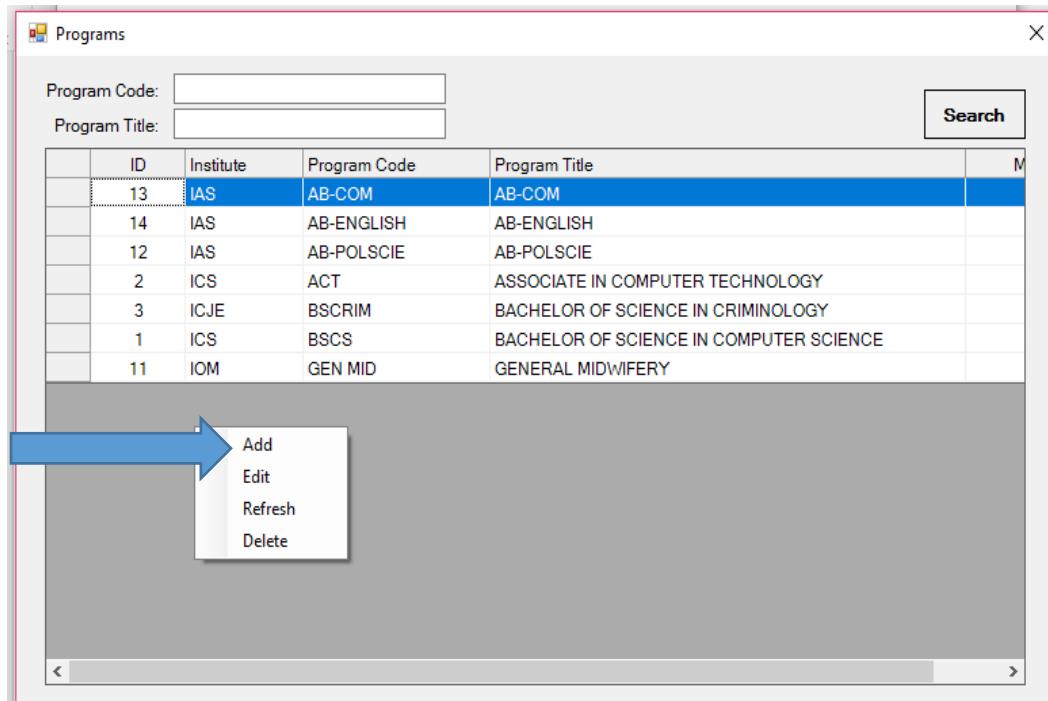


Figure 65: Programs List

- Once you've click the "Add" button this will be shown and you must browse and select institute where the Programs belong and input Program information then click "Save" button.

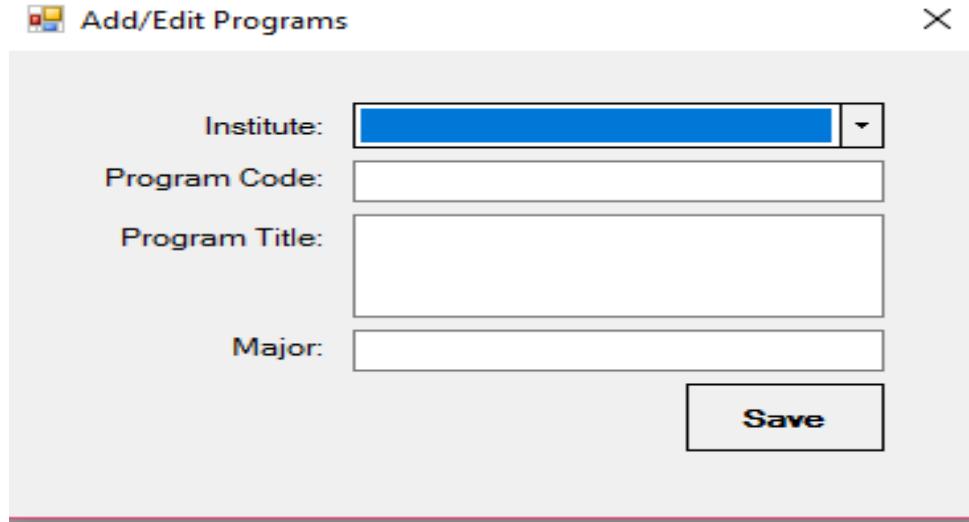


Figure 66: Program Name

- It will be automatically saved. The dialog box will appear displaying message that notifying you the success.

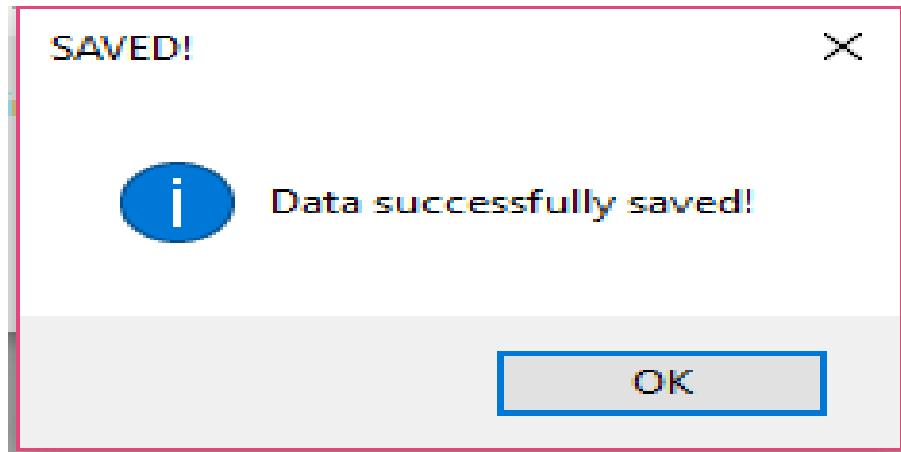


Figure 67: Program Name Successfully Saved

8. Add User's Position

You can add User's Position through clicking the "Administrative" button. Then select "Position" that shown below.

NOTE: Both Administrator and Commissioner can add User's Position.

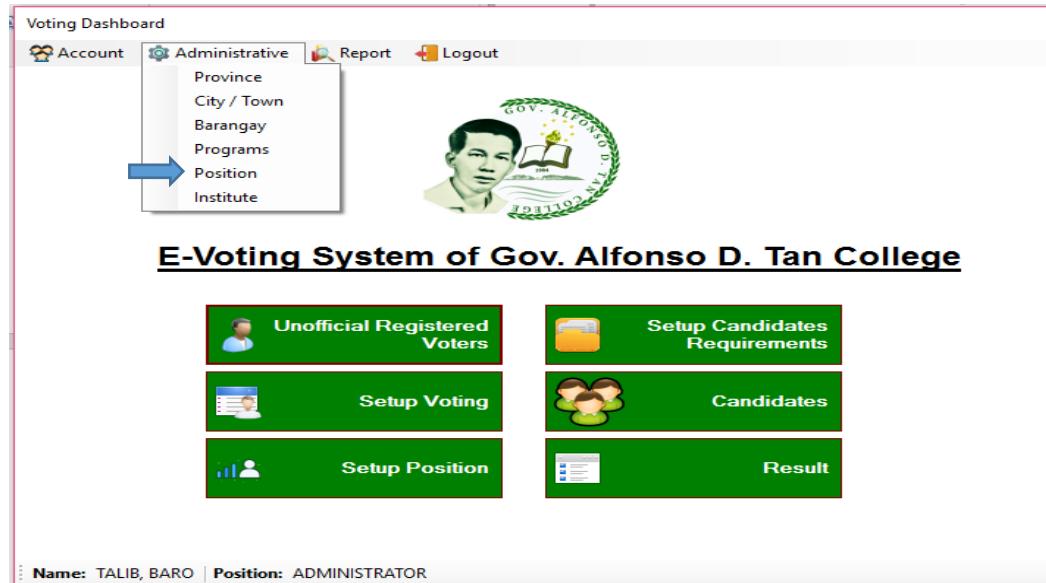


Figure 68: Position

- The Position List will display then you must click "Add" button as shown below.

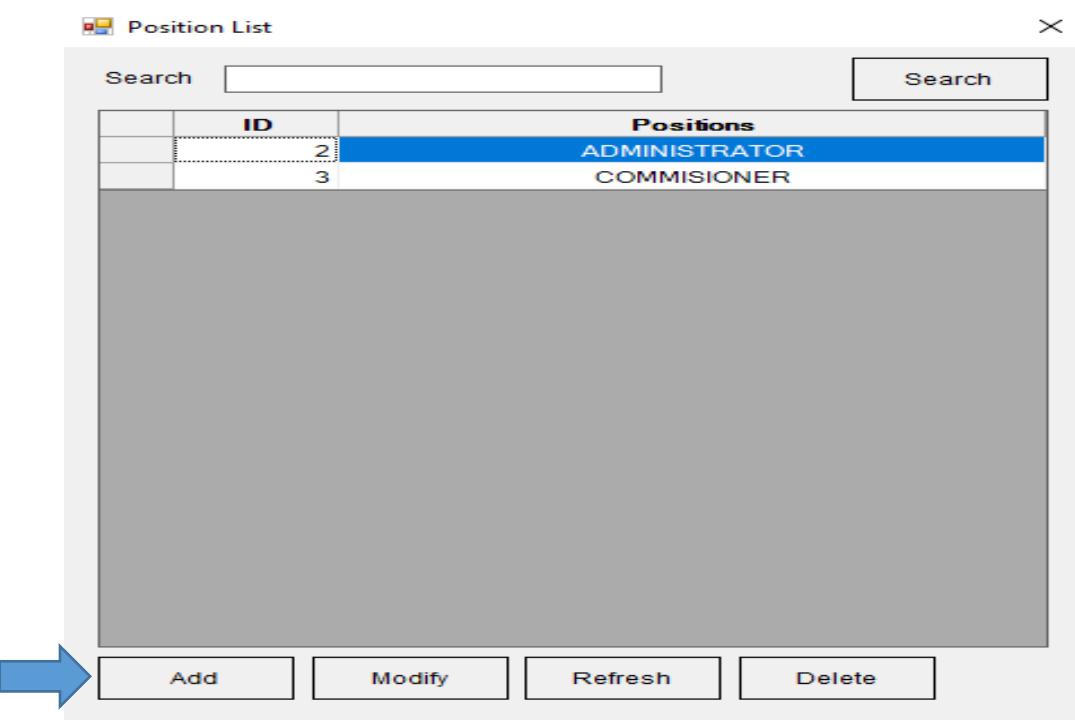


Figure 69: Position List

- Once you've click the "Add" button this will be shown and input position Name then click "Save" button.

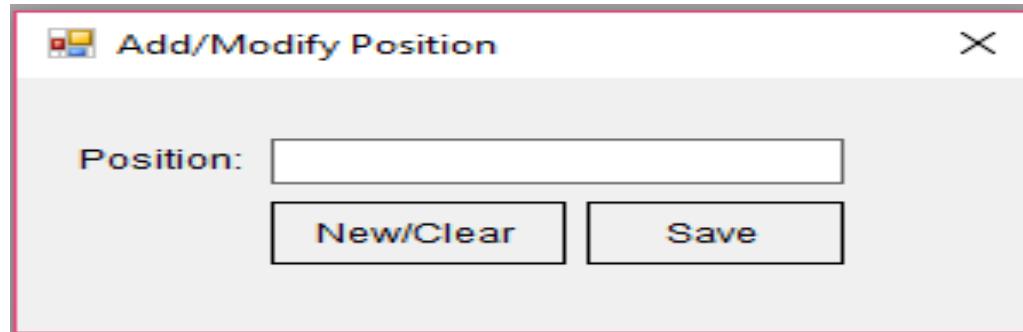


Figure 70: Position Name

- It will be automatically saved. The dialog box will appear displaying message that notifying you the success.

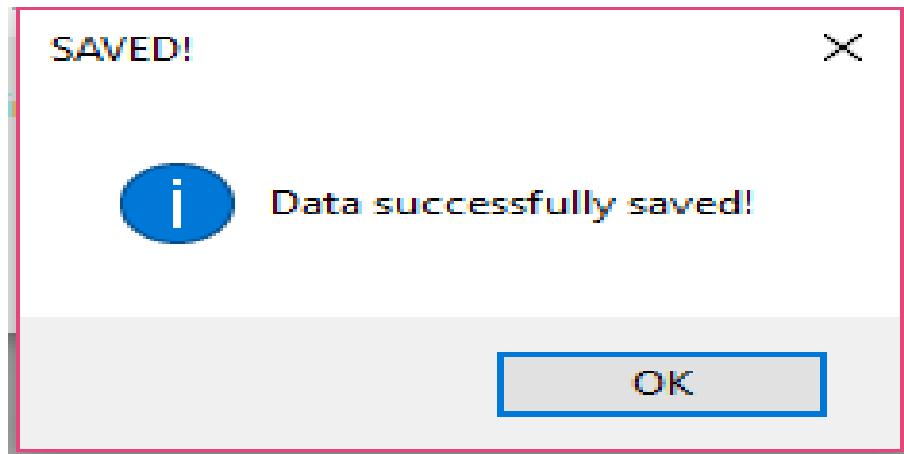


Figure 71: Position Name Successfully Saved

9. Election Setup

By Clicking “Setup Voting” button, you can setup a new type of voting.

NOTE: Both Administrator and Commissioner can Setup Election.

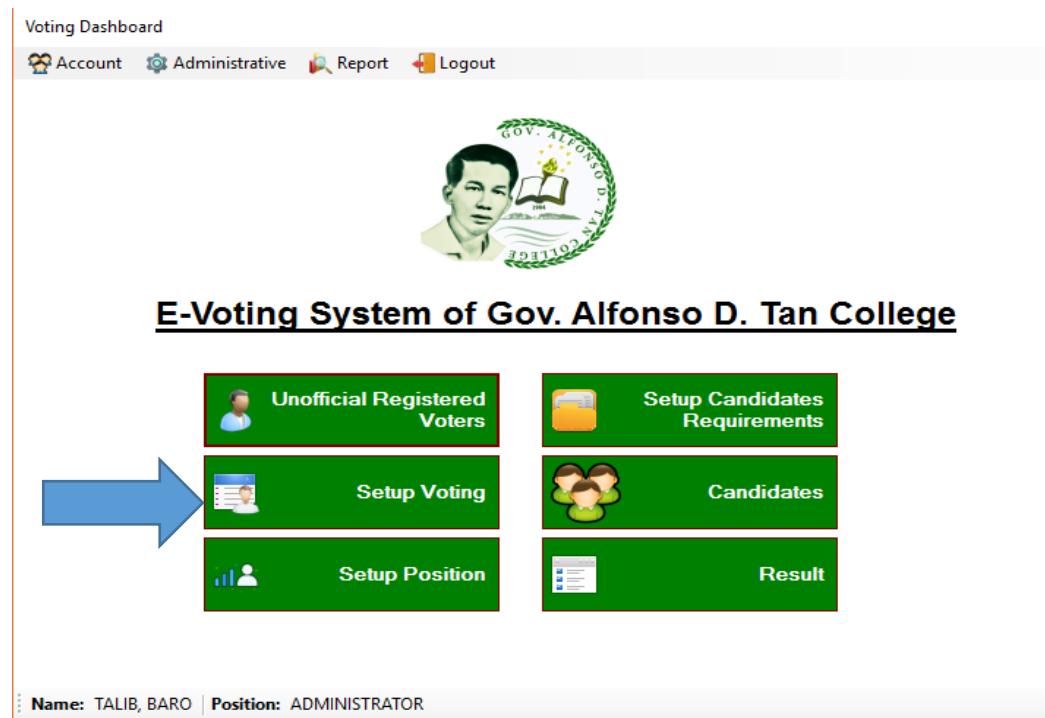


Figure 72: Setup New type of Voting

- The Election Type List will display then you click “Add” button as shown below.

The screenshot shows the 'ELECTION SETUP' window. It has a header with 'ELECTION SETUP' and a sub-header 'Election Name:'. Below this is a table with columns 'ID', 'Election Name', 'Academic Year', and 'Default'. There is one row with ID 1, Election Name 'SSB ELECTION 2019', Academic Year '2018-2019', and Default checked. At the bottom, there is a message 'Number of Row(s) : 1' and buttons for 'Set Default', 'Add', 'Edit', 'Refresh', and 'Delete'.

Figure 73: Election Type List

- Once you've click the "Add" button this will be shown and you must select an Academic Year and input Election Name then click "Save" button.

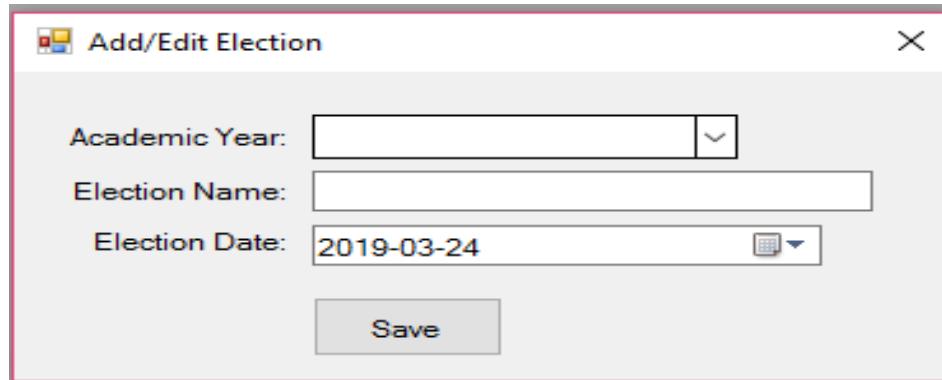


Figure 74: Election Name

- It will be automatically saved. The dialog box will appear displaying message that notifying you the success.

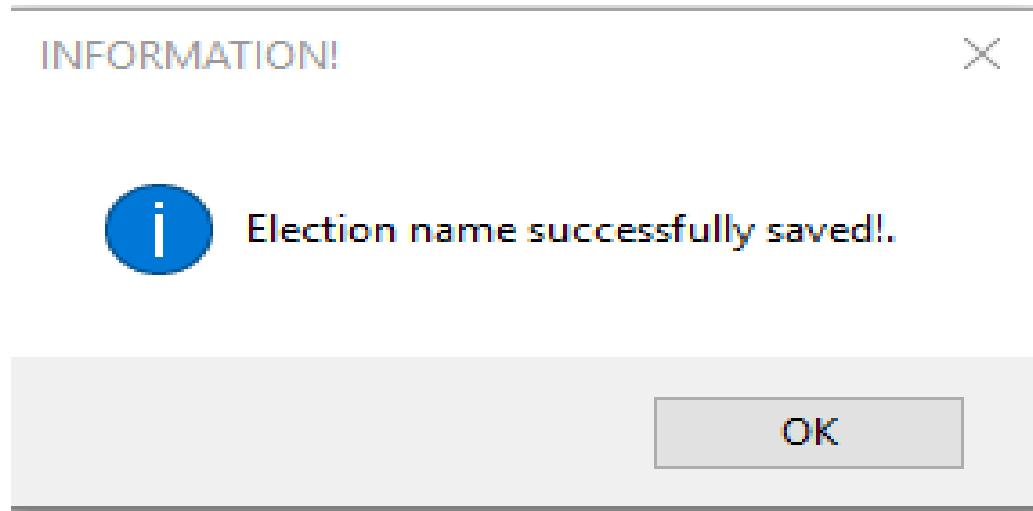


Figure 75: Election Name Successfully Saved

- You can Set default election by selecting a Type of election from election type list and then click Set Default button.

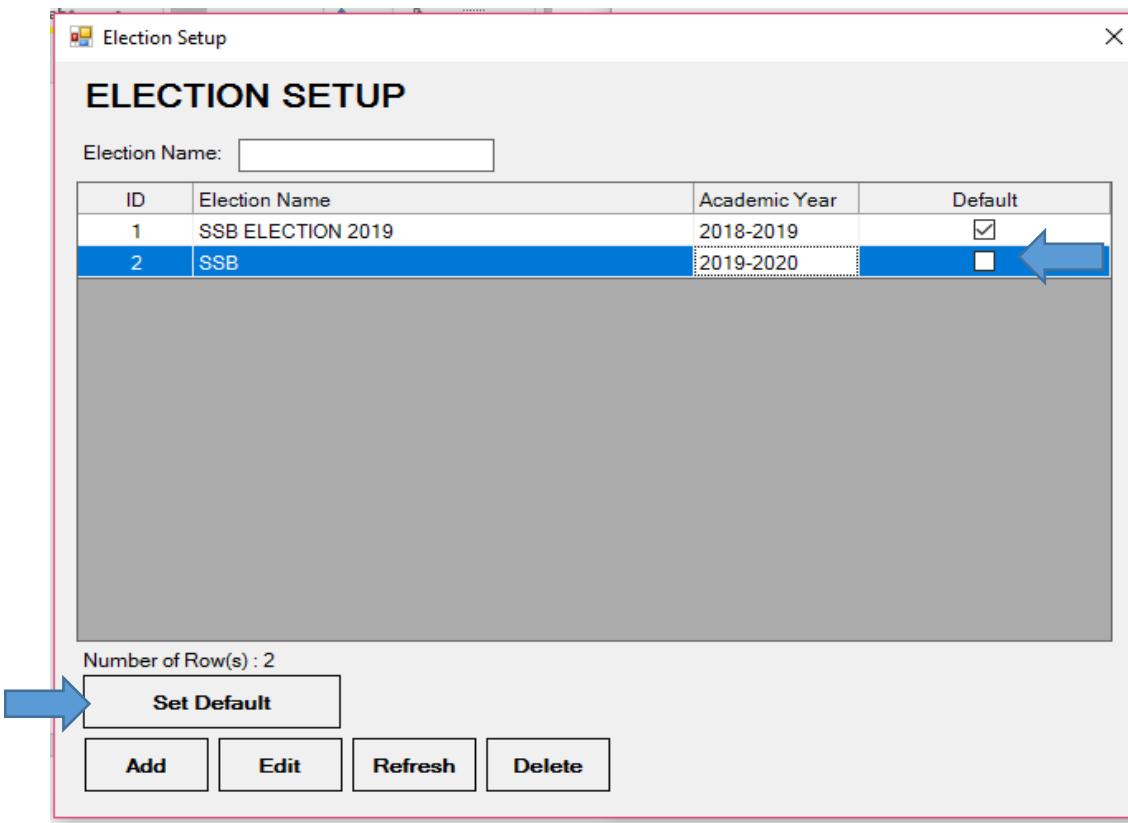


Figure 76: Set Default Election Type

Note: Once an Election Type set as default then it will marked as Check.

10. Setup Position

By Clicking “Setup Position” button, you can setup a new Position of the default election type.

NOTE: Both Administrator and Commissioner can Setup Position.



Figure 77: Setup New Position

- The Position List will display then you click “New” button as shown below.

The screenshot shows the "Position List" window. It has search fields for "Position" and "Election", and a "Search" button. Below is a table with columns: Position, Description, Election Name, Academic Year, and Order No. The table contains the following data:

Position	Description	Election Name	Academic Year	Order No
GOVERNOR	GOVERNOR	SSB ELECTION 2019	2018-2019	1
CEO	CEO	SSB ELECTION 2019	2018-2019	2
VICE CEO	VICE CEO	SSB ELECTION 2019	2018-2019	3
SECRETARY	SECRETARY	SSB ELECTION 2019	2018-2019	4
TREASURER	TREASURER	SSB ELECTION 2019	2018-2019	5
AUDITOR		SSB ELECTION 2019	2018-2019	6
EXTERNAL AFFAI	EXTERNAL AFFAIRS OFFICER	SSB ELECTION 2019	2018-2019	7
SOCIO CULTURAL	SOCIO CULTURAL OFFICER	SSB ELECTION 2019	2018-2019	8
SPORTS DEVELO	SPORTS DEVELOPMENT OFFICI	SSB ELECTION 2019	2018-2019	9
ADMINISTRATOR	ADMINISTRATOR	SSB ELECTION 2019	2018-2019	10

At the bottom, there are three buttons: "New" (with a plus sign), "Update" (with a pencil), and "Delete" (with a minus sign).

Figure 78: Position List

- Once you've click the "New" button this will be shown and you must Browse and select an election type and then input necessary information then click "Save" button.

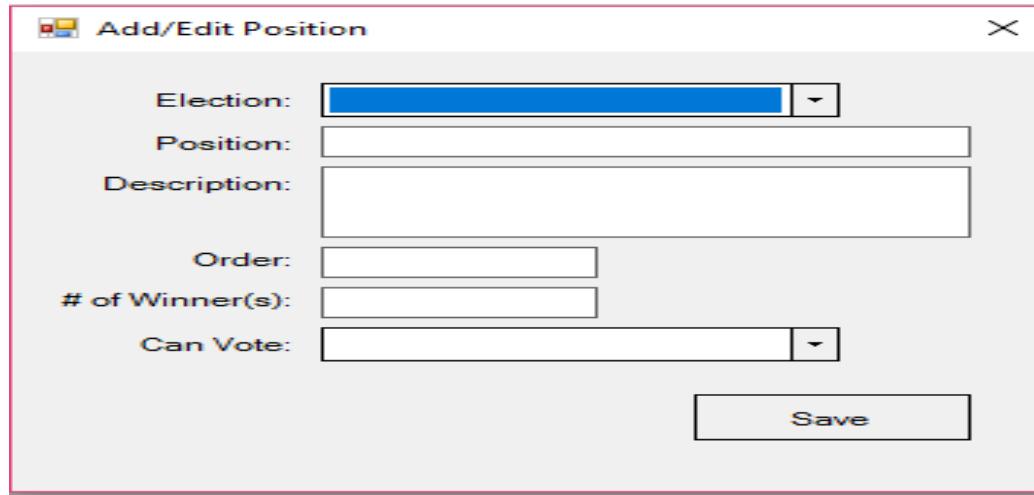


Figure 79: Position Name

- It will be automatically saved. The dialog box will appear displaying message that notifying you the success.

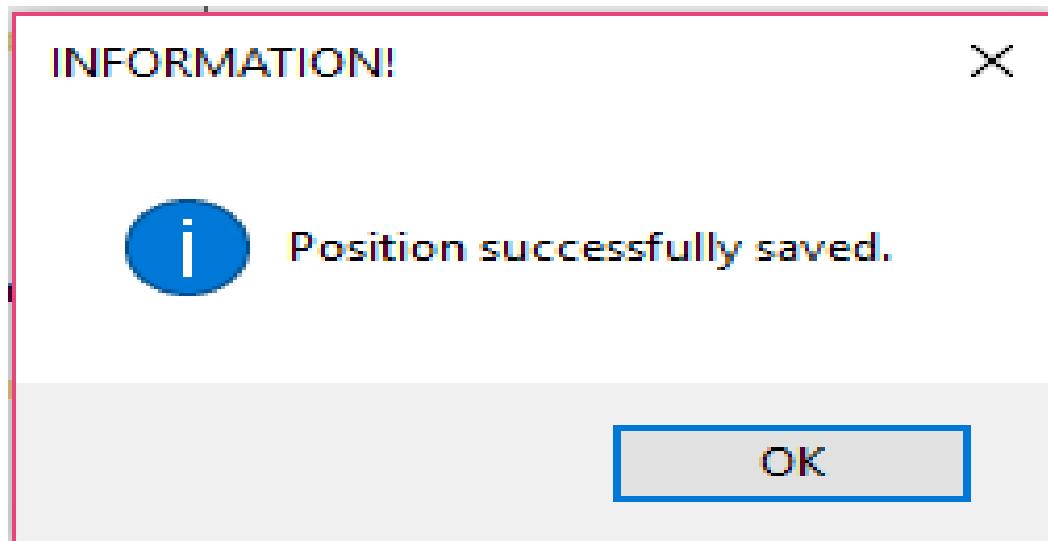


Figure 80: Position Name Successfully Saved

11. Registration of Unofficial Voters

By Clicking “Unofficial registered Voters” button, you can register a new voter to a default election type.

NOTE: Both Administrator and Commissioner can Register Unofficial Voters.



Figure 81: Registration of Unofficial Voters

- The Voter's List will display then you click “Add” button as shown below.

The screenshot shows the "Voters List" window. At the top, there is a search bar with fields for ID No., Lastname,Firstname, Program, and a Search button. Below the search bar is a table displaying voter records. The table has columns: Agency No., Lastname, Firstname, Middlename, Program, Year Level, and Birthdate. The data in the table is as follows:

Agency No.	Lastname	Firstname	Middlename	Program	Year Level	Birthdate
159319	ADONA	CHRYPTON	RENZ	AB-POLSCIE	3	3/11/2019
160950	ALAB	MARVILOWE	TORAJA	BSCS	3	3/11/2019
181334	ALCALA	JACQUELINE	TULO	BSCS	1	3/12/2019
144648	ALDUHESA	FEEL MARY	CORDOVA	BSCS	4	3/11/2019
170375	ALEGRE	LEE BAVERLY	ENCLONAR	BSCS	2	3/11/2019
185520	ALGADIPE	HEZZLE	BARINAN	BSCS	1	3/12/2019
184343	ALIA	ALBERT	BORJA	BSCS	1	3/11/2019
188199	ALJAS	ROSELYN	MACASARTE	ACT	1	3/13/2019
080310	AMPARADO	ETIENNE WAYNE	NAMOCATCATG	BSCS	4	3/13/2019
188053	ASON	GENELIZA	CATULAG	BSCS	1	3/12/2019
182566	AUXTERO	ZYRA JEAN	BONGCALES	BSCS	1	3/12/2019
185914	BALAYONG	JUSTINE JED	ASARI	BSCS	1	3/13/2019
183460	BANAYBANAY	JONA	APLACA	BSCS	1	3/14/2019
188170	BANO	ANNIE JANE	MAGLANGIT	BSCS	2	3/11/2019
183373	BARHIYO	JOHNA	ANSAO	BSCS	1	3/12/2019
168216	BAYAWA	JATE CHRISTIAN	SINOY	BSCS	3	3/13/2019
154269	BETACURA	RUSTOM		BSCS	3	3/11/2019
152372	BETFE	PRETTY ANGEL	SARTF	BSCS	1	3/13/2019

At the bottom of the window, there are three buttons: **Add**, **Edit**, and **Delete**. A blue arrow points to the **Add** button.

Figure 82: Voter's List

- Once you've click the “Add” button, the following will be shown and you must Enroll Fingerprint, take photo, electronic signature and input necessary information of the voter then click “Save” button.

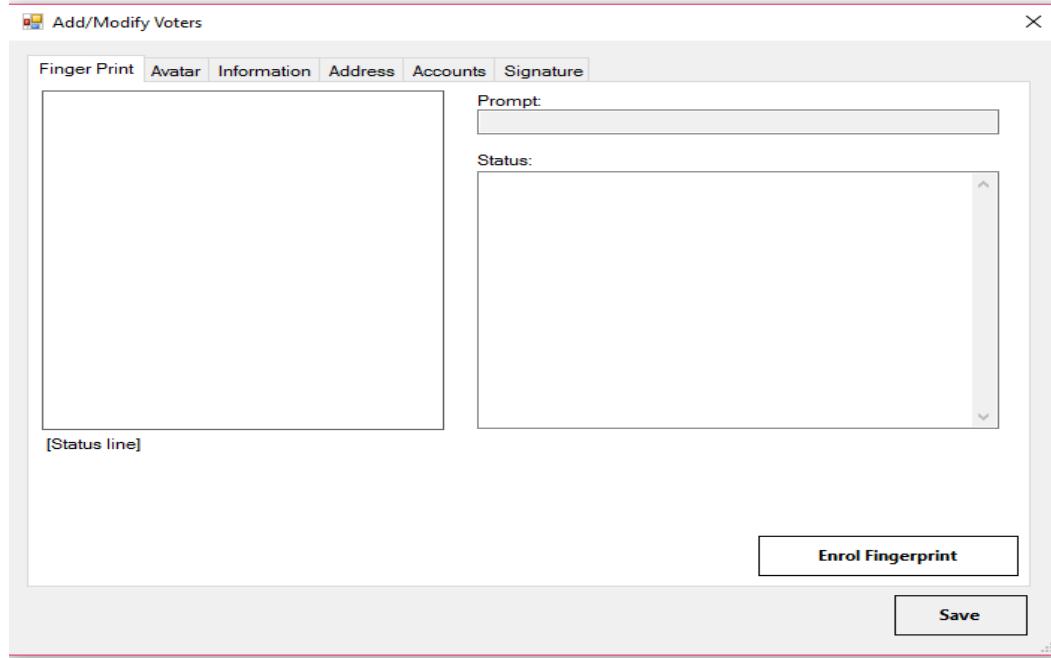


Figure 83: Enroll Finger Print

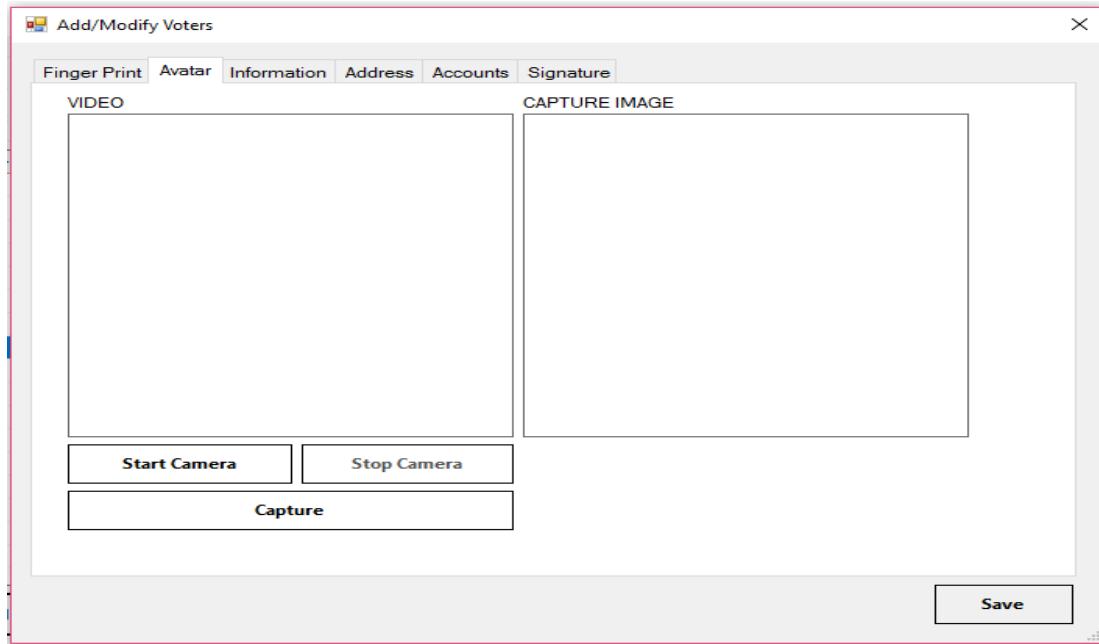


Figure 84: take Photo

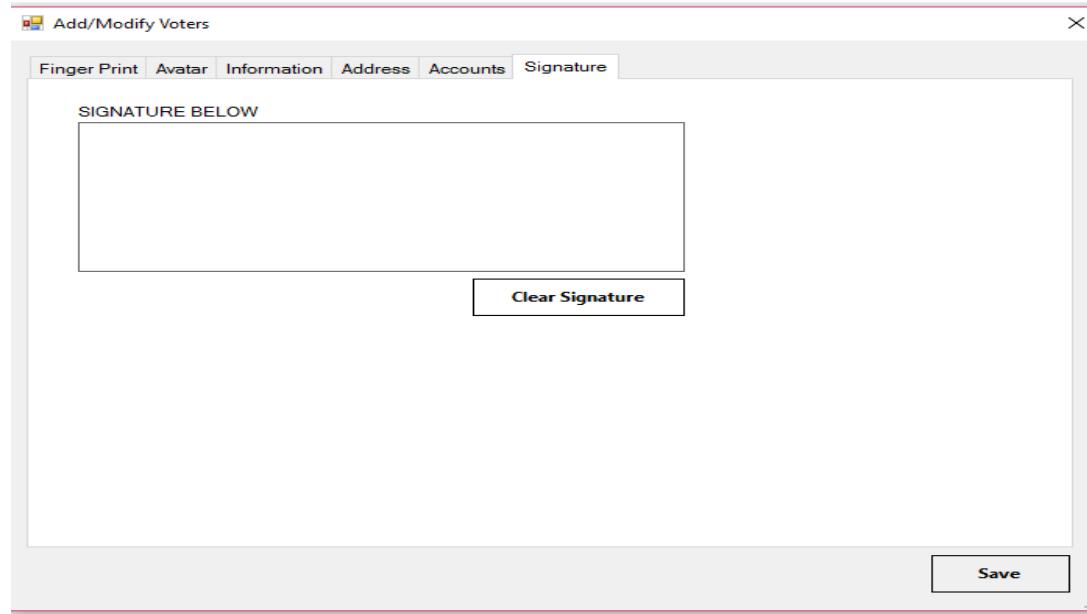


Figure 85: Signature

The screenshot shows the "Information" tab of the "Add/Modify Voters" application. The form includes the following fields:

- ID No.: [Text Input]
- Lastname: [Text Input]
- Firstname: [Text Input]
- Middlename: [Text Input]
- Institute: [Text Input]
- Birthdate: [Text Input] (containing "2019-03-24") with a calendar icon to its right.
- Gender: [Text Input]
- Religion: [Text Input]
- Email: [Text Input]
- Contact No.: [Text Input]
- Program/Position: [Text Input]
- Year Level: [Text Input] with a numeric value "0" and up/down arrows.
- Birth Place:
 - Province: [Text Input]
 - Barangay: [Text Input]
 - Zip Code: [Text Input]
- City/Town: [Text Input]
- Unit/Street: [Text Input]

A "Save" button is located in the bottom right corner of the form area.

Figure 86: Voter's Information

- It will be automatically saved. The dialog box will appear displaying message that notifying you the success.

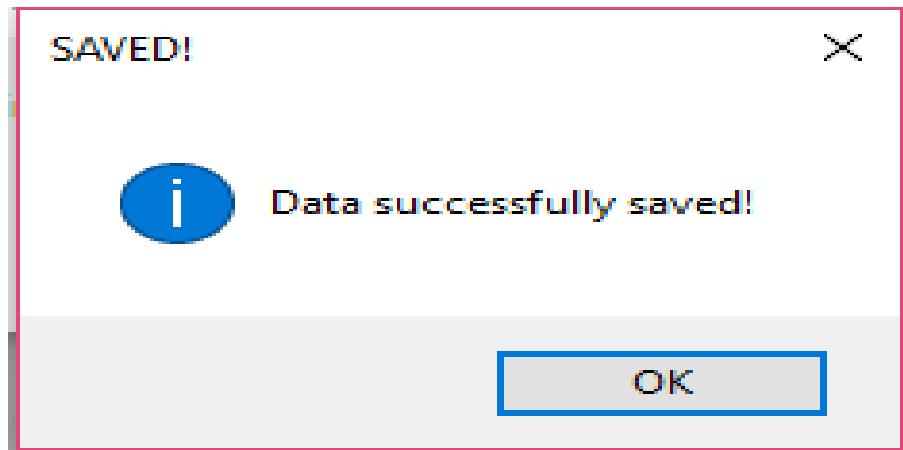


Figure 87: Voter Successfully Saved

12. Registration of Official Voters

To register Official Voters you click “Setup Voting” button.

NOTE: Both Administrator and Commissioner can Register Official Voters.



Figure 88: Registration of Official Voters

- The Election Type List will display then you right click “Show Registered Voters” button as shown below.

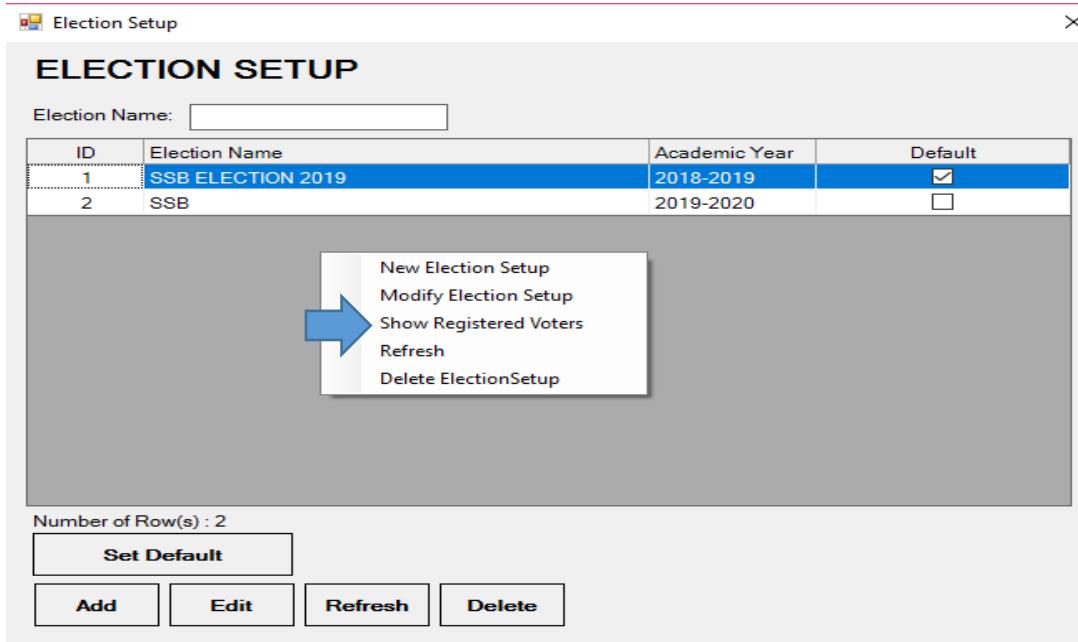


Figure 89: Show Registered voters

- Once you click “Show registered Voters” button, Official registered Voters List will display and you can register a new voter by clicking ‘Register Voter From List’ button as shown below:

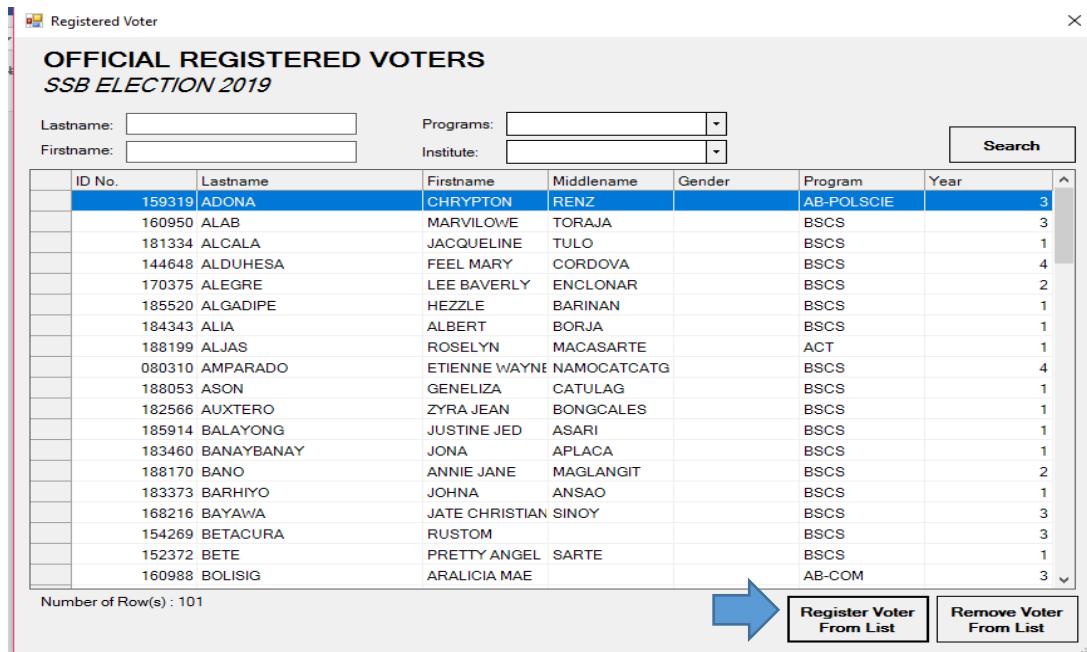


Figure 90: Official Registered Voters List

- Once you click “Register Voter From List” button, unofficial registered Voters List will display and you can register a new voter by clicking ‘Add Voter(s)” button as shown below:

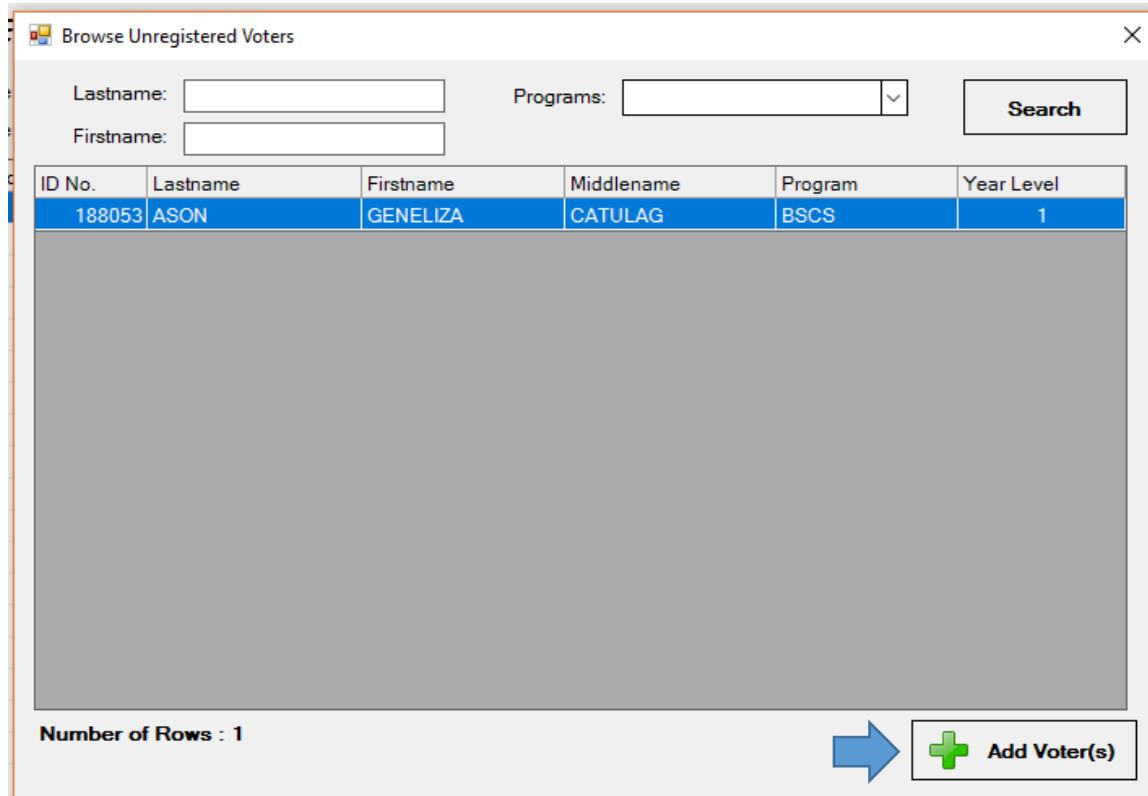


Figure 91: Add Official Voters

- It will be automatically saved. The dialog box will appear displaying message that notifying you the success.

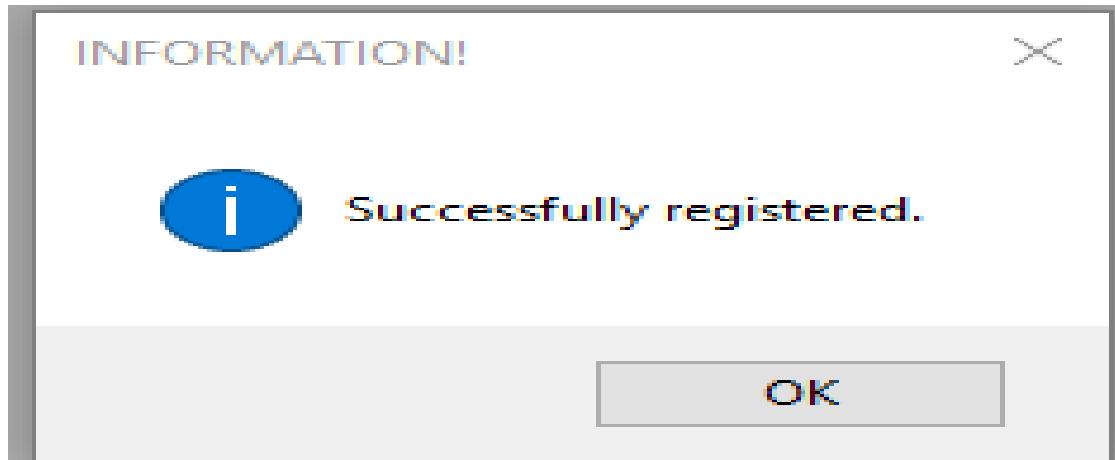


Figure 92: Voter Officially registered

13. Registration of Candidates

To register Candidates, you click “Candidates” button.

NOTE: Both Administrator and Commissioner can Register candidates.



Figure 93: candidates

- The Candidates will display then you click “New Candidate” button as shown below.

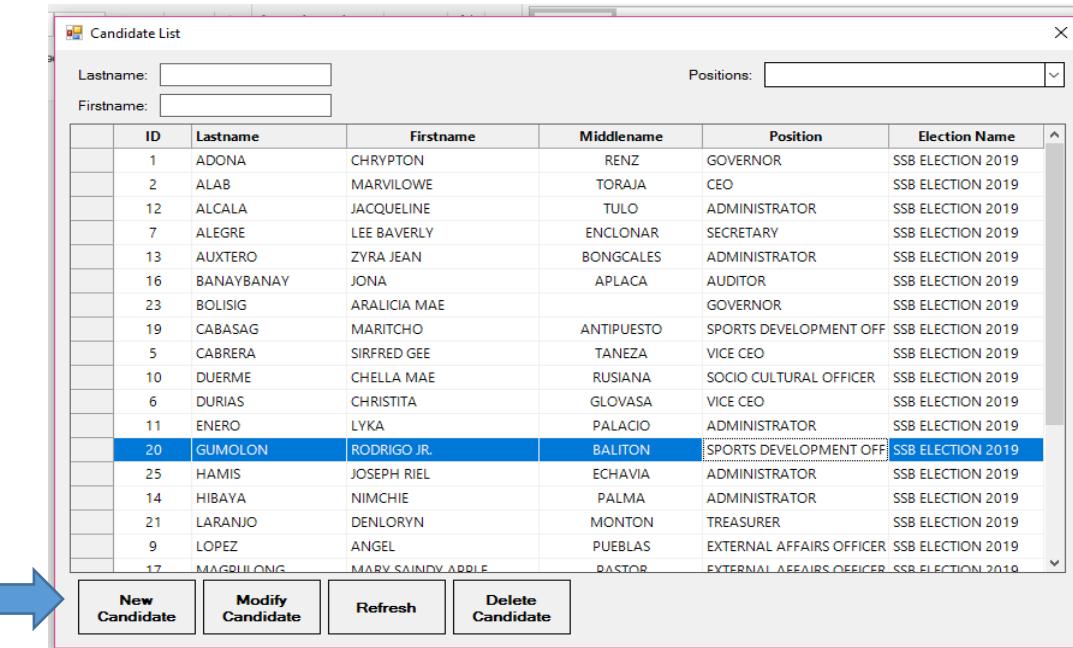


Figure 94: new candidate's registration

- After you click the “ New candidate” button, this will display :

The screenshot shows a Windows application window titled "Add/Modify Candidates". It contains several input fields: "Position" (dropdown), "Description" (text area), "Election Type" (text area), "Voters ID" (text box with browse button), "Lastname" (text box), "Firstname" (text box), and "Middlename" (text box). A blue arrow points to the "Save" button at the bottom right.

Figure 95: Add candidates

NOTE: Only the Officially Registered Voters can add as a candidate.

- You will browse then the registered voters then add as a candidate then click “Save” button and this will display again the candidates list.
- It will be automatically saved. The dialog box will appear displaying message that notifying you the success.

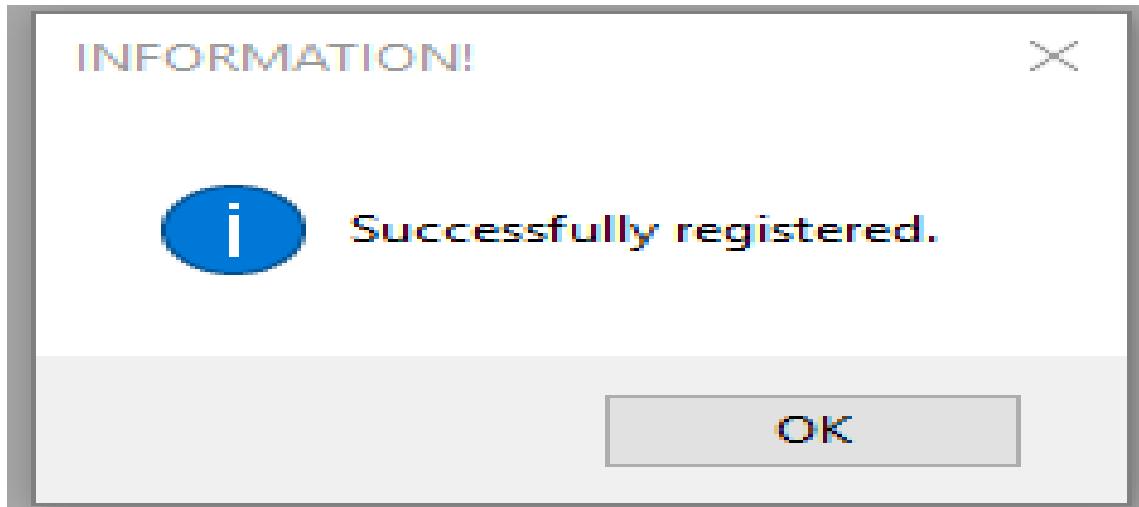


Figure 96: Official candidates

14. Setup Candidates Requirements

By Clicking “Setup Candidates requirements” button, you can setup a new requirements of a candidates.

NOTE: Both Administrator and Commissioner can Setup Candidates Requirements.



Figure 97: Setup New Candidates Requirements

- The Candidates Requirements List will display then you click “New Setup” button as shown below.

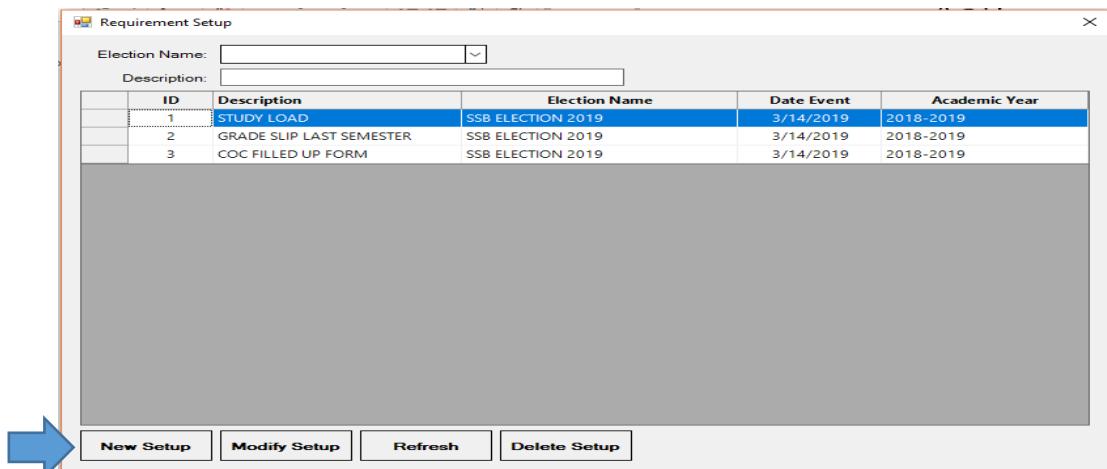


Figure 98: Election Type List

- Once you've click the "New Setup" button this will be shown and you must select an Election Type and input Description then click "Save" button.

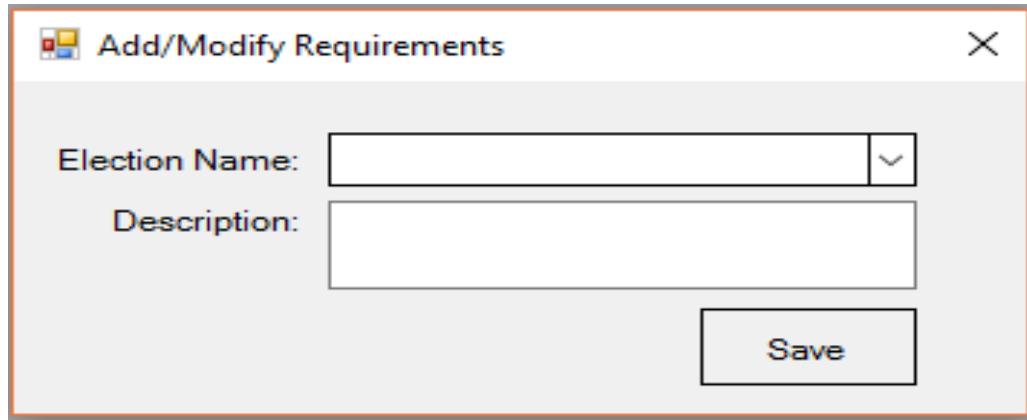


Figure 99: Requirements Description

- It will be automatically saved. The dialog box will appear displaying message that notifying you the success.

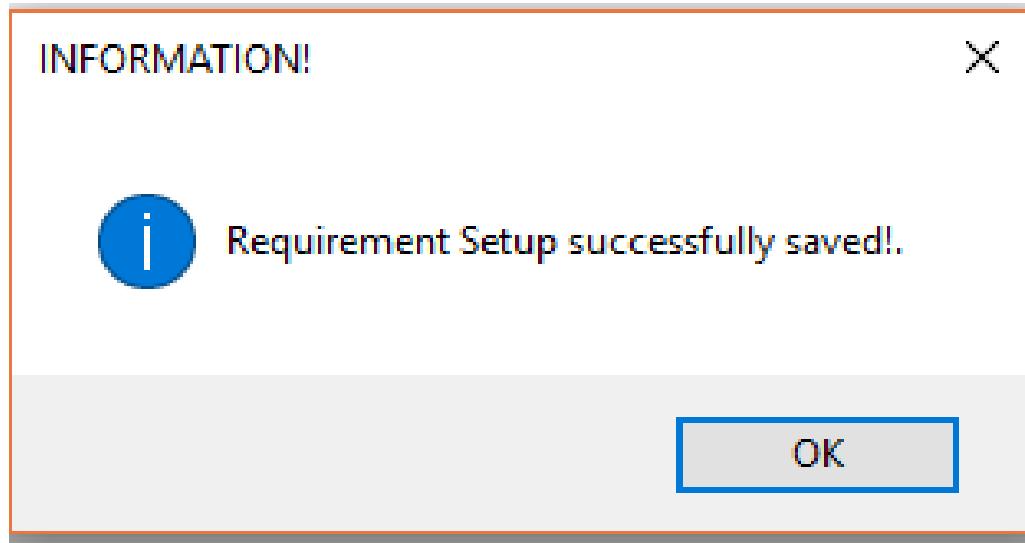


Figure 100: Requirement Successfully Saved

15. Result

To view and print the result of the election just click “Result” button as shown in the figure below.

NOTE: Both Administrator and Commissioner can View and print the result.



Figure 101: Result

- Next step is you will browse and select an institute then click “Preview” button to view the result and then click the print icon to print.

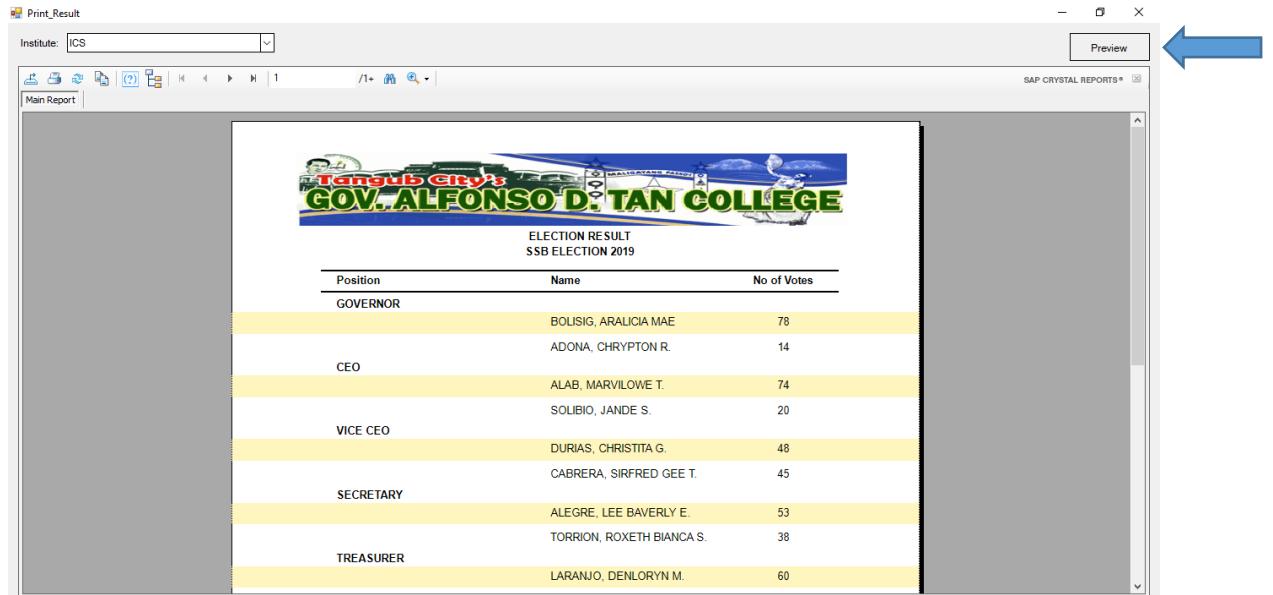


Figure 102: Election Result

NOTE: Candidates with highlighted color are the winners who got the highest votes during the election.

16. Report

To view and print the list of voters who voted during the election, click “Report” button as shown in the figure below.

NOTE: Both Administrator and Commissioner can View and print the Voted list.

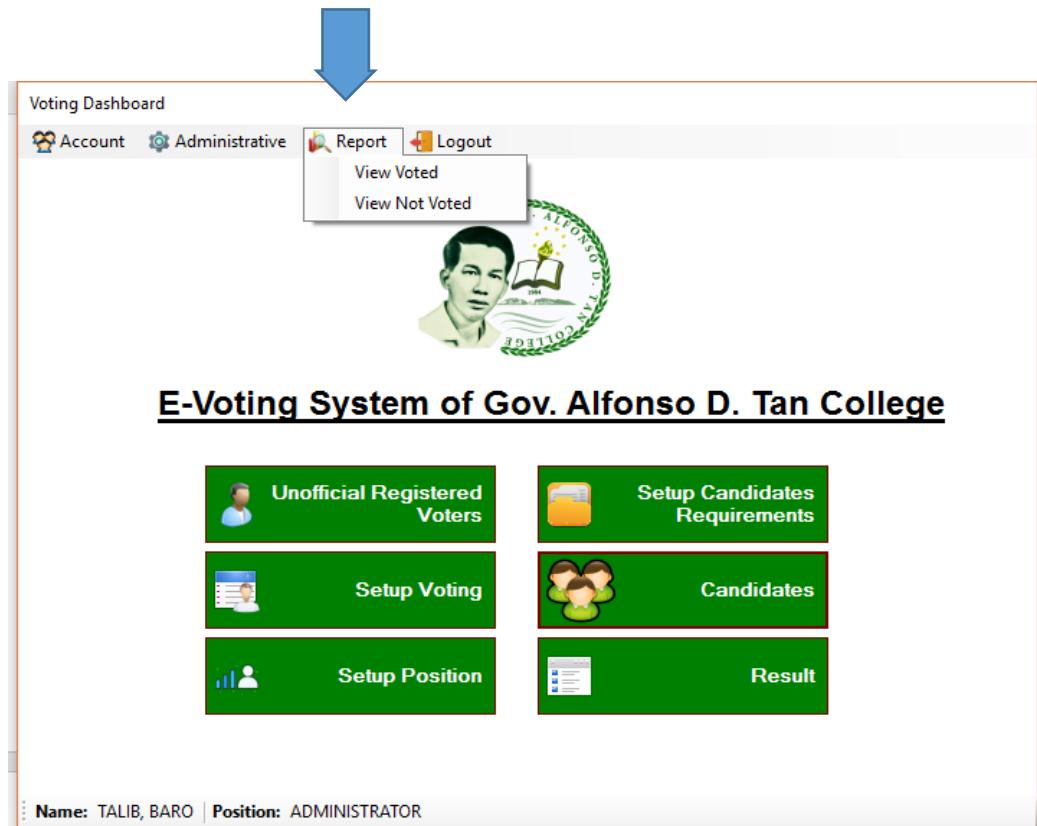


Figure 103: Report voted list

- Next step is you will browse and select an election type and institute then click “Search” button to view the Voted list and then click the print icon to print.

The screenshot shows a SAP Crystal Reports interface titled "Print_VotedList". At the top, there are dropdown menus for "Election Type" (set to "SSB ELECTION 2019") and "Institute" (set to "ICS"). To the right of these is a "Search" button. Below the header is a toolbar with various icons. The main report area displays a banner for "Tangub City's GOV. ALFONSO D. TAN COLLEGE" followed by a table titled "VOTED LIST SSB ELECTION 2019". The table has columns for "Voters ID", "Lastname", "Firstname", "Middlename", and "Institute". The data in the table is as follows:

Voters ID	Lastname	Firstname	Middlename	Institute
26	ALAB	MARVILOWE	TORAJA	ICS
75	ALCALA	JACQUELINE	TULO	ICS
38	ALDUHESA	FEEL MARY	CORDOVA	ICS
19	ALEGRE	LEE BAVERLY	ENCLONAR	ICS
61	ALGADIPE	HEZZLE	BARINAN	ICS
10	ALIA	ALBERT	BORIA	ICS
93	ALJAS	ROSELYN	MACASARTE	ICS
44	ASON	GENELIZA	CATULAG	ICS
74	AUXTERO	ZYRA JEAN	BONGCALES	ICS
89	BALAYONG	JUSTINE JED	ASARI	ICS
77	BANAYBANAY	JONA	APLACA	ICS
11	BANO	ANNIE JANE	MAGLANGIT	ICS
51	BARHIYO	JOHNA	ANSAO	ICS

At the bottom of the report, it says "Current Page No.: 1" and "Total Page No.: 1+", with a "Zoom Factor: 100%" option. A blue arrow points from the right side of the search button towards the print icon.

Figure 104: Voted list

To view and print the list of voters who not voted during the election, click “Report” button as shown in the figure below.

NOTE: Both Administrator and Commissioner can View and print the Not Voted list.

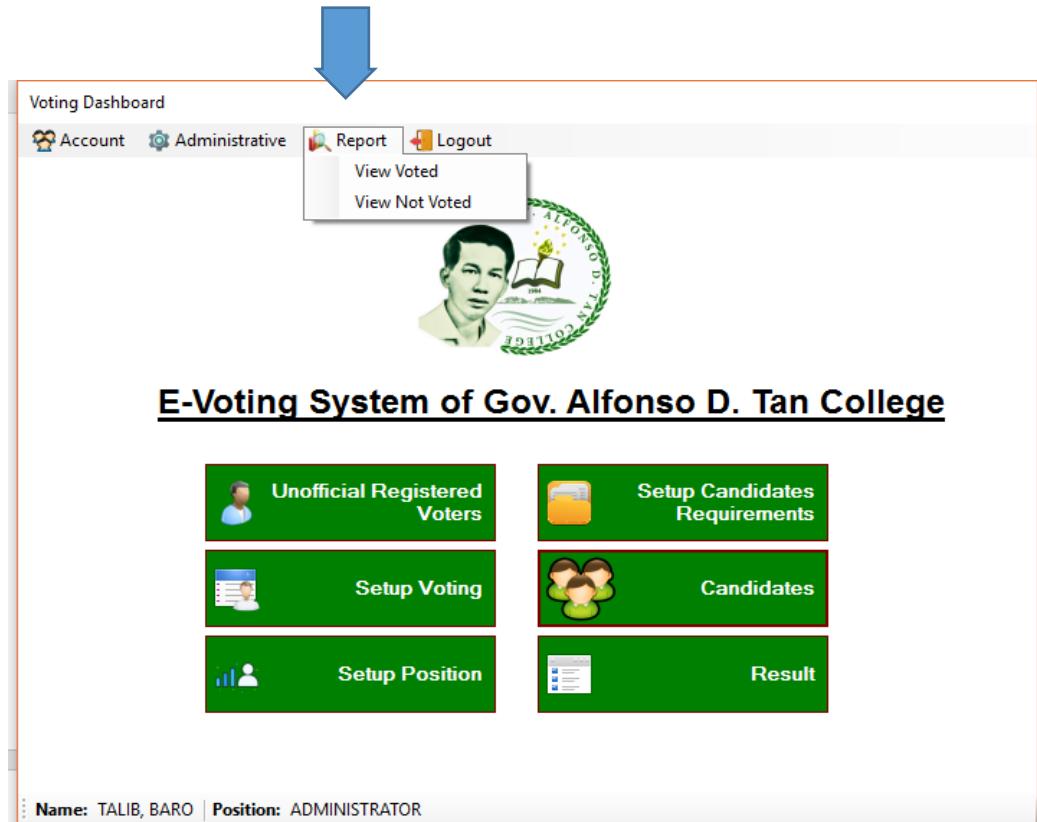
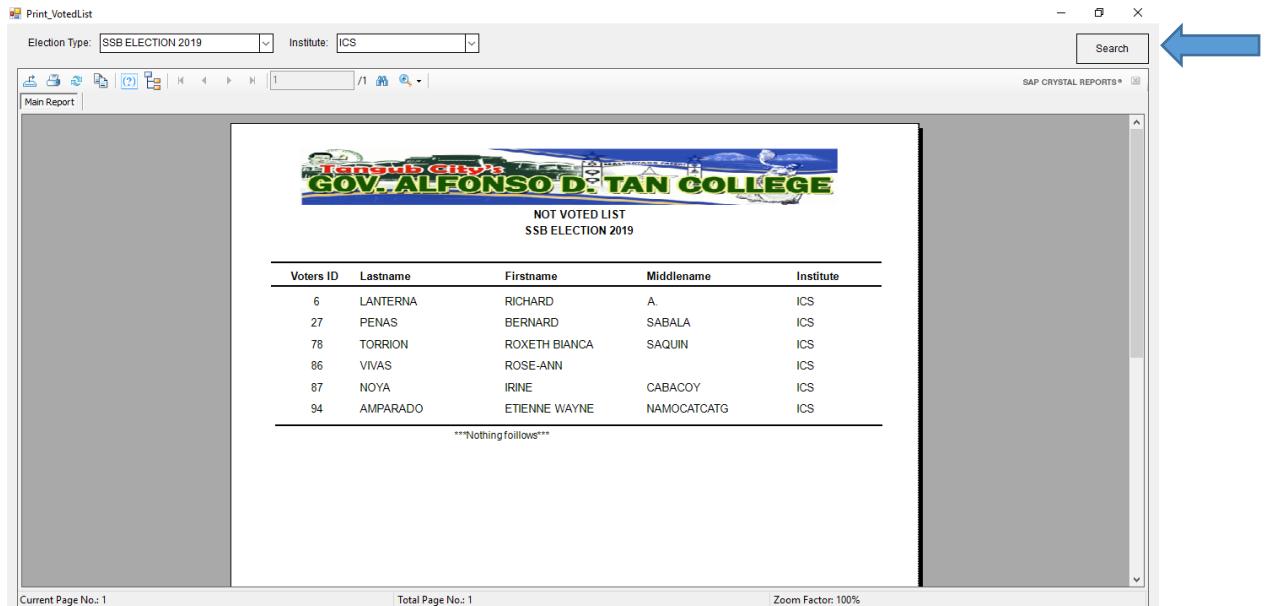


Figure 105: Report Not Voted List

- Next step is you will browse and select an election type and institute then click “Search” button to view the Not Voted list and then click the print icon to print.



The screenshot shows a SAP Crystal Reports interface titled "Print_VotedList". At the top, there are dropdown menus for "Election Type" (set to "SSB ELECTION 2019") and "Institute" (set to "ICS"). To the right of these is a "Search" button with a magnifying glass icon, which has a blue arrow pointing towards it from the right side of the image. Below the search bar is a toolbar with various icons. The main report area displays a title banner for "Tangub City's GOV. ALFONSO D. TAN COLLEGE" followed by "NOT VOTED LIST SSB ELECTION 2019". Below the banner is a table with the following data:

Voters ID	Lastname	Firstname	Middlename	Institute
6	LANTERNA	RICHARD	A.	ICS
27	PENAS	BERNARD	SABALA	ICS
78	TORRION	ROXETH	BIANCA	ICS
86	VIVAS	ROSE-ANN		ICS
87	NOYA	IRINE	CABACOY	ICS
94	AMPARADO	ETIENNE	WAYNE	NAMOCATCATG

At the bottom of the report, a note states "***Nothing follows***". The footer of the report includes "Current Page No.: 1", "Total Page No.: 1", and "Zoom Factor: 100%".

Figure 106: Not Voted list

E-Voting System of Gov. Alfonso D. Tan College User's Manual

Voters Side

1. Getting Started

- The Voters need to use any kind of android phone or a personal computer in order to vote for a particular candidate.
- The voters need to connect to a default network use in voting process.
- Next step is you must visit the default IP Address. After completing this steps, the registered voters will login and input username and password as shown below:

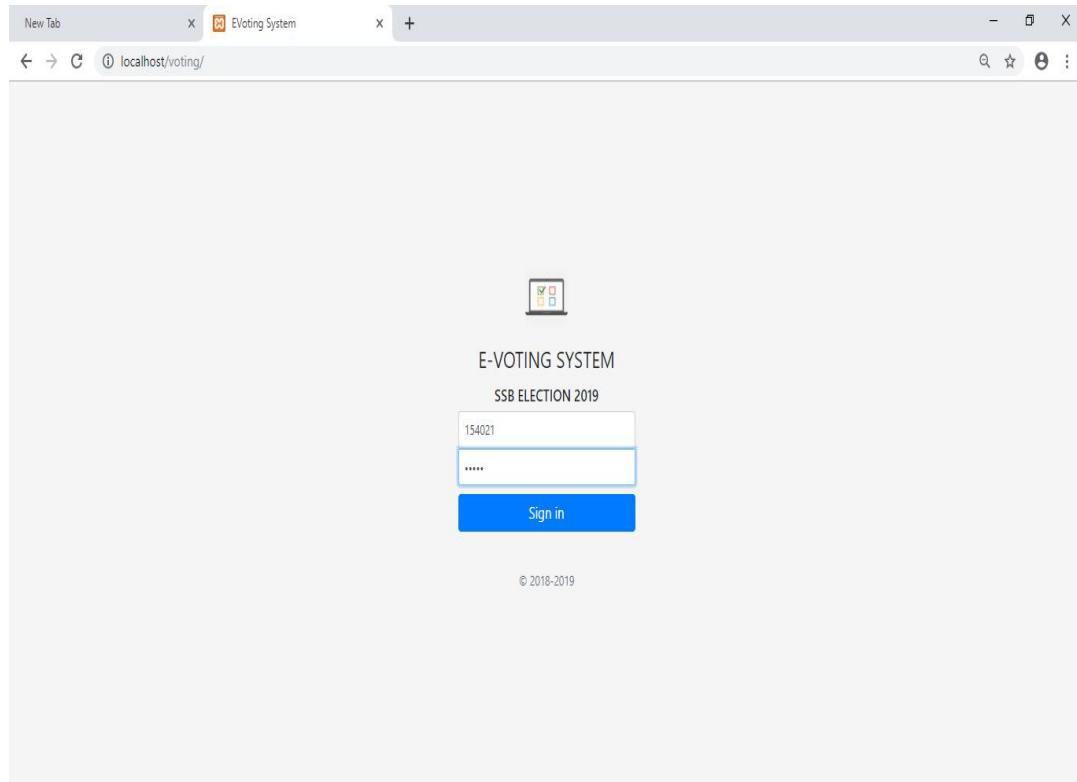


Figure 107: Login (Voter's Side)

2. Homepage

Once you've Login this will be shown:

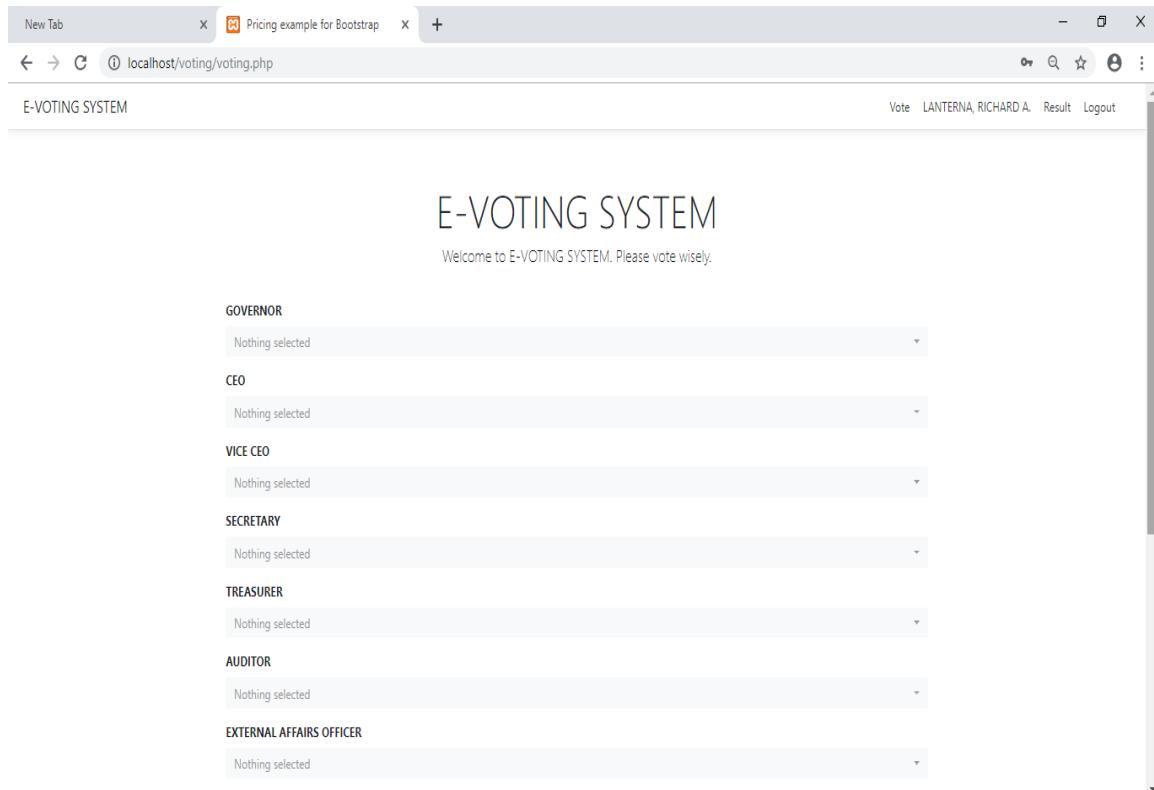


Figure 108: Homepage (Voter's Side)

3. Voting

To vote for a candidate, you must tap the name of the candidates you want to vote and then click “Submit” button as shown below:

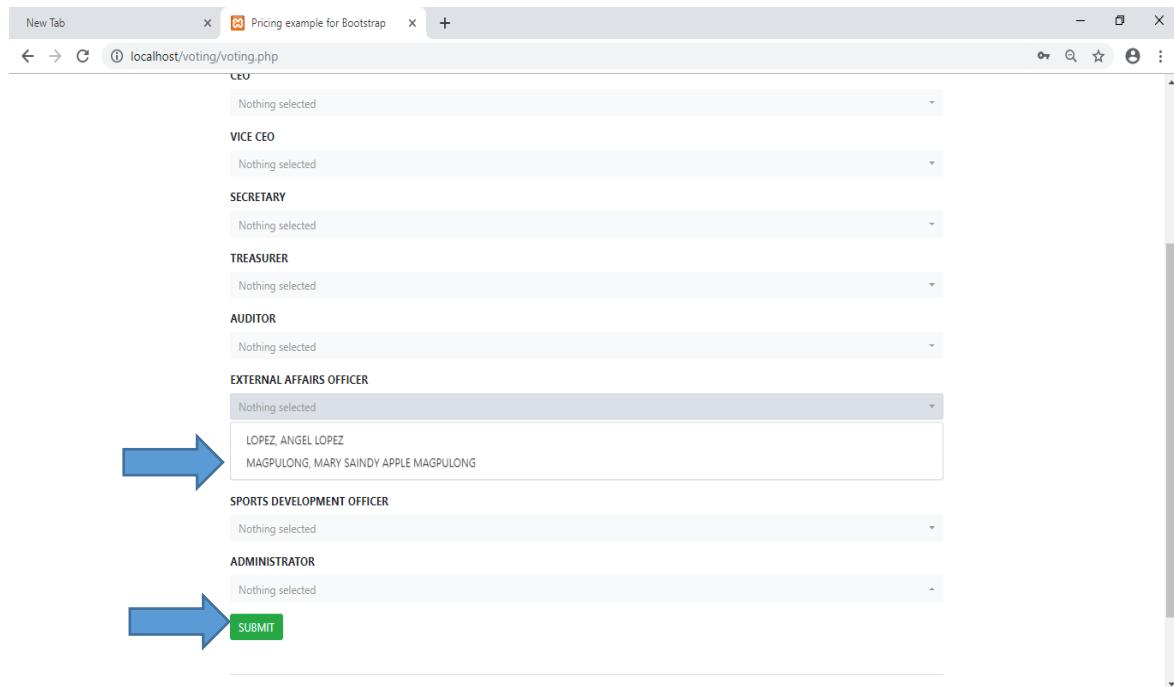


Figure 109: Voting Options

NOTE: Think before you click the “Submit” button. After clicking “Submit” button, your vote will be saved and you cannot go back to voting option anymore, in other words you cannot vote again. So vote wisely.

- After clicking the “Submit” button, a message will appear as notifying you that you’ve voted for a candidate.

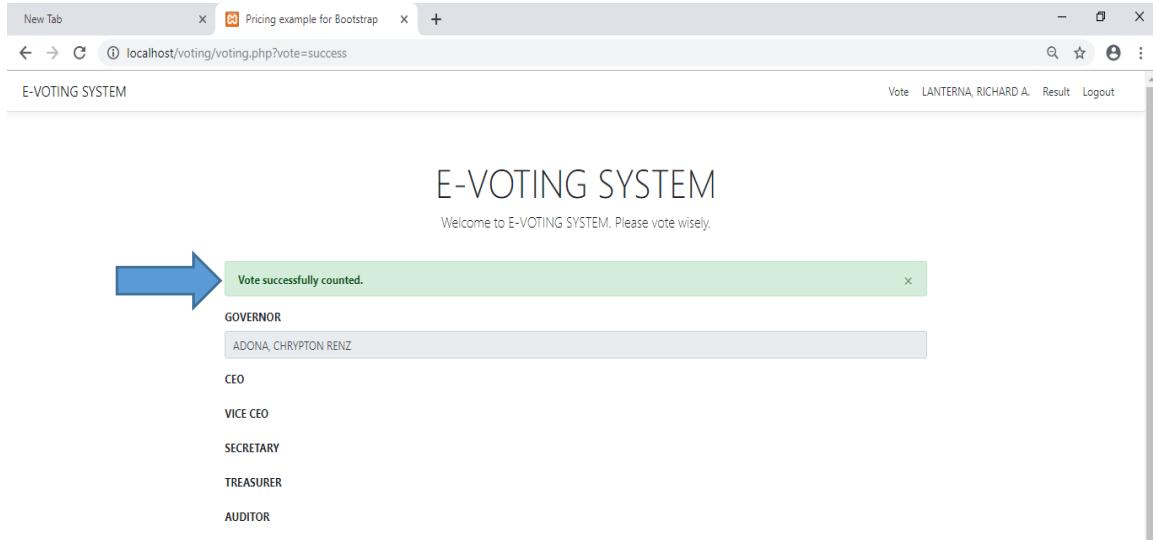


Figure 110: Successfully Voted

NOTE: Your Vote is now counted.

4. Voter's Profile

You can view your profile by clicking “Your Name” at the upper right corner as shown below:

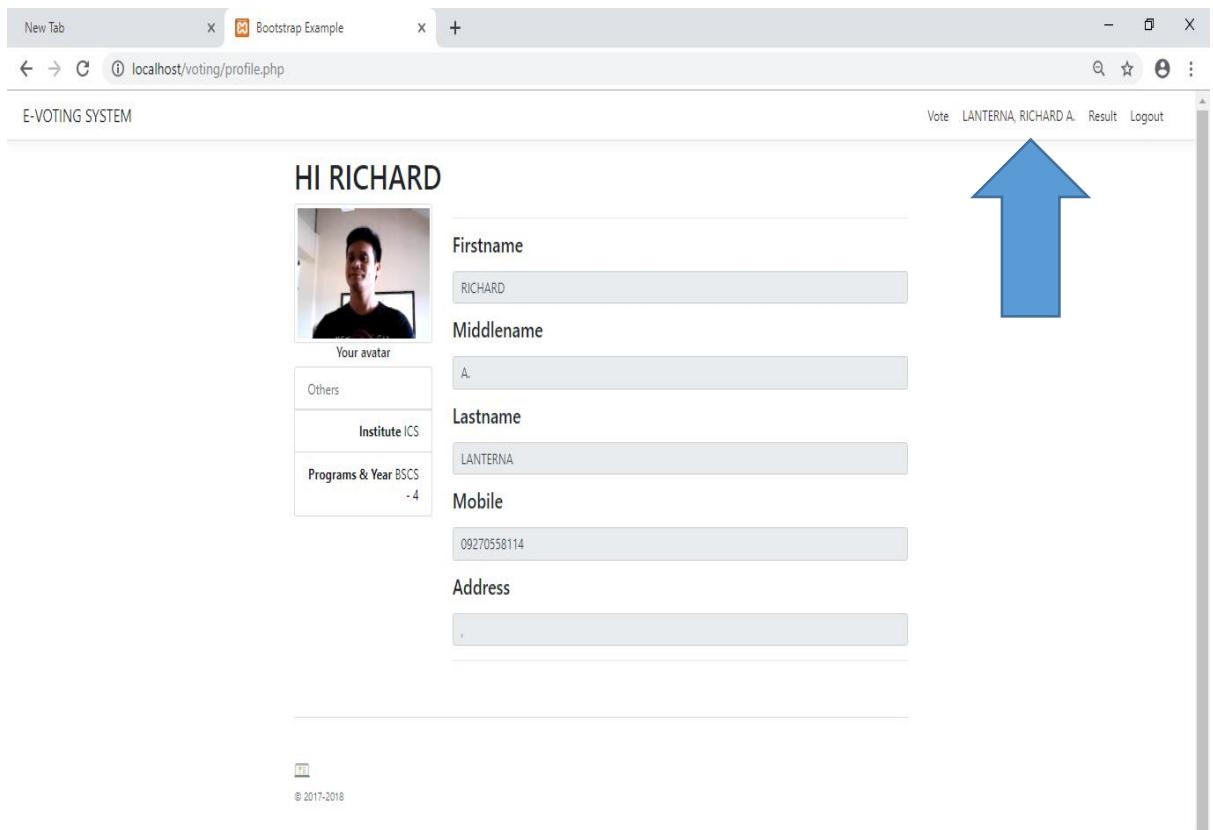
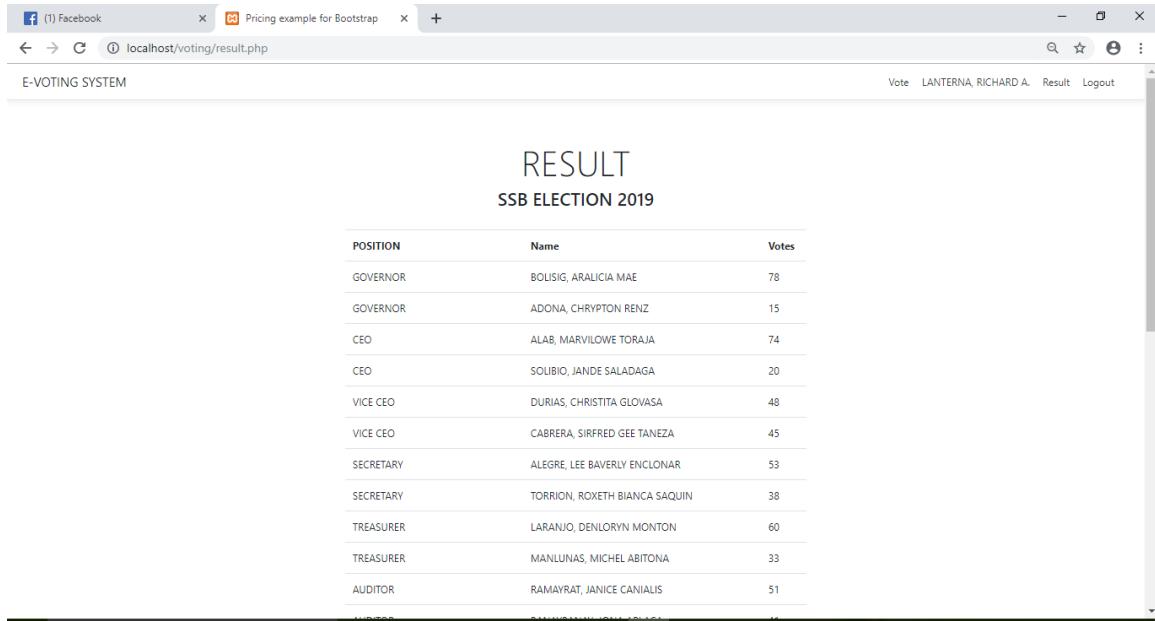


Figure 111: Voter's Profile

NOTE: Only the Admin and Commissioner can perform any changes on you profile information. If you wish any changes, please visit and see the admin and commissioner.

5. View Result

You can also see the result of the election by clicking “Result” button as shown below:



The screenshot shows a web browser window with the title "Pricing example for Bootstrap". The address bar indicates the page is "localhost/voting/result.php". The main content area displays the results of the "SSB ELECTION 2019" under the heading "RESULT". The results are presented in a table with three columns: "POSITION", "Name", and "Votes". The data in the table is as follows:

POSITION	Name	Votes
GOVERNOR	BOLISIG, ARALICIA MAE	78
GOVERNOR	ADONA, CHRYPTON RENZ	15
CEO	ALAB, MARVILOWE TORAJA	74
CEO	SOUIBIO, JANDE SALADAGA	20
VICE CEO	DURIAS, CHRISTITA GLOVASA	48
VICE CEO	CABRERA, SIRFRED GEE TANEZA	45
SECRETARY	ALEGRE, LEE BAVERLY ENCLONAR	53
SECRETARY	TORRION, ROXETH BIANCA SAQUIN	38
TREASURER	LARANJO, DENIORYN MONTON	60
TREASURER	MANLUNAS, MICHEL ABITONA	33
AUDITOR	RAMAYRAT, JANICE CANIALIS	51

Figure 112: View Result (Voter's Side)

Appendix B

GUI (Graphical User Interface)

Administrator and Commissioner side

Homepage

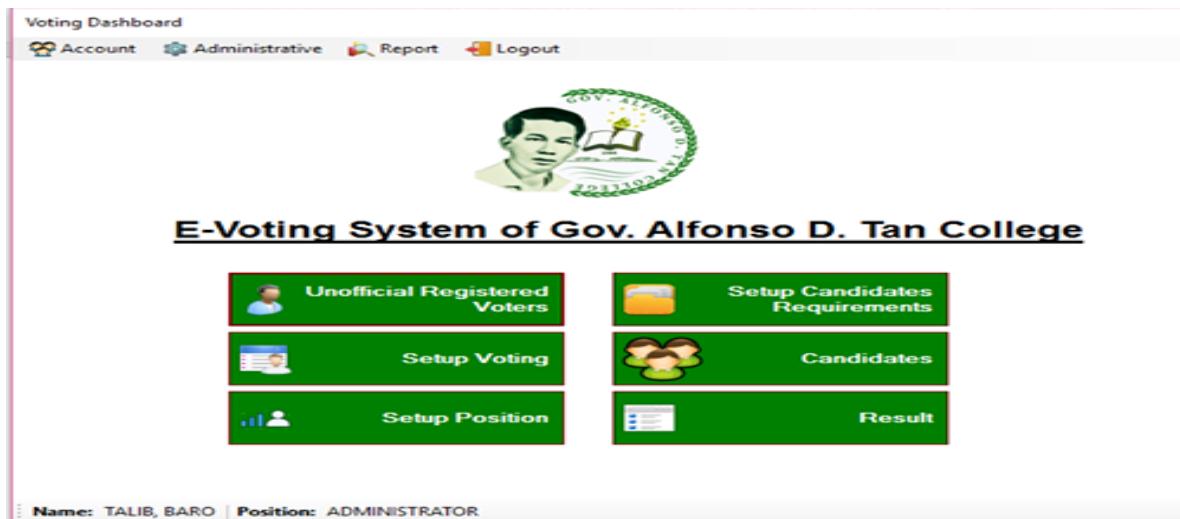


Figure 113: Homepage

Account

Account						
Voters ID:		Lastname:	Firstname:	Middlename:	Position	
User ID	Username	Lastname	Firstname	Middlename		
6	admin	TALIB	BARO		ADMINISTRATOR	
8	anchie	JAMISOLA	ANNCHIE	CEMAFRANCA	COMMISSIONER	
9	joan	DOCY	JOAN	LACIA	COMMISSIONER	
10	godfrey	YAMA	GODFREY	MARLO	COMMISSIONER	

At the bottom are buttons for 'Add', 'Modify', 'Refresh', 'Delete', and 'Reset Password'.

Figure 114: Account

Add/Modify Account

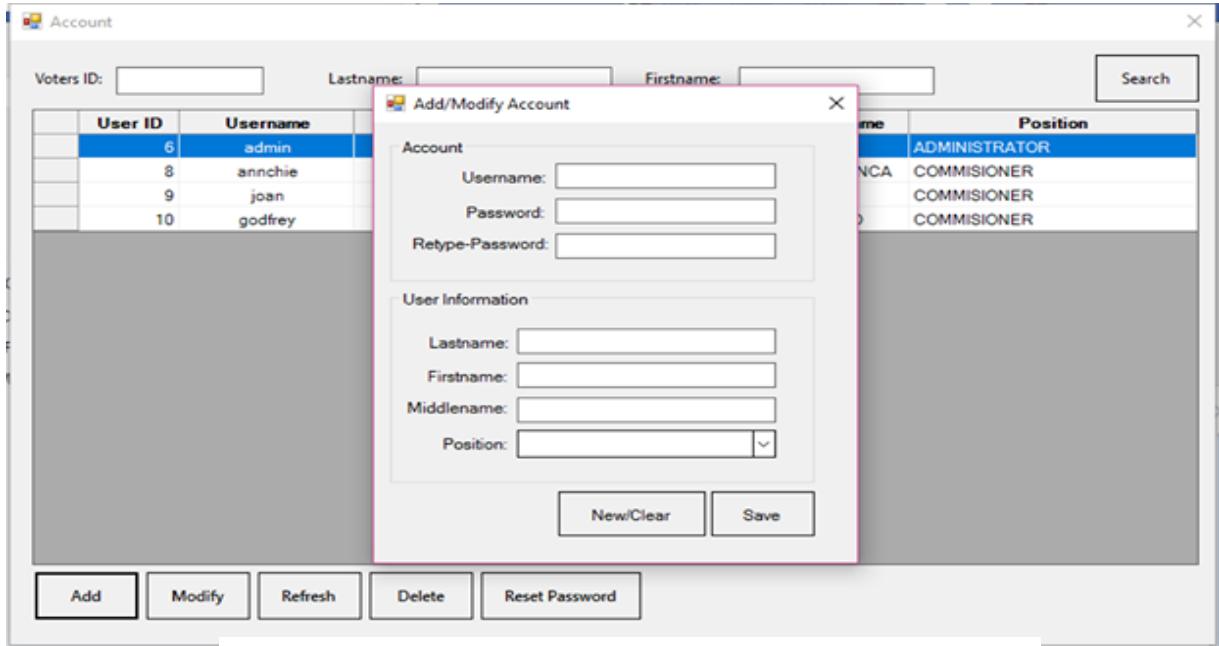


Figure 115: Add/Modify Account

Reset Password

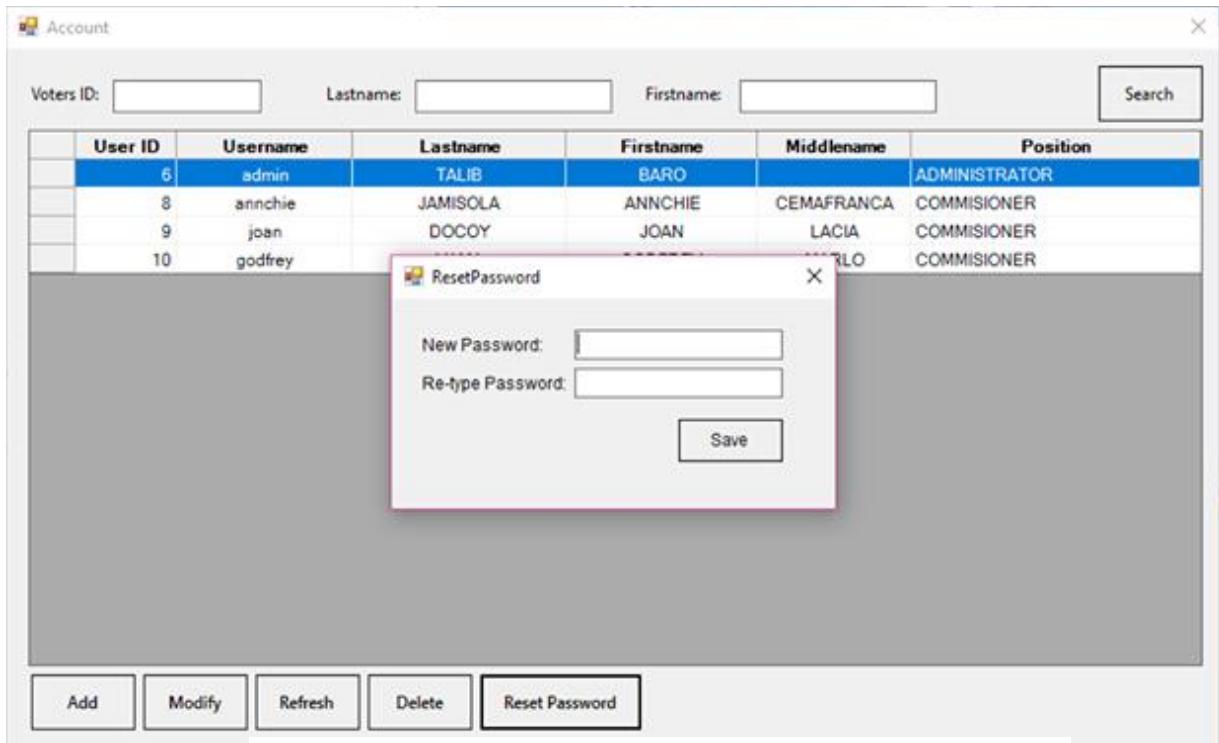


Figure 116: Reset Password

Province Main Form

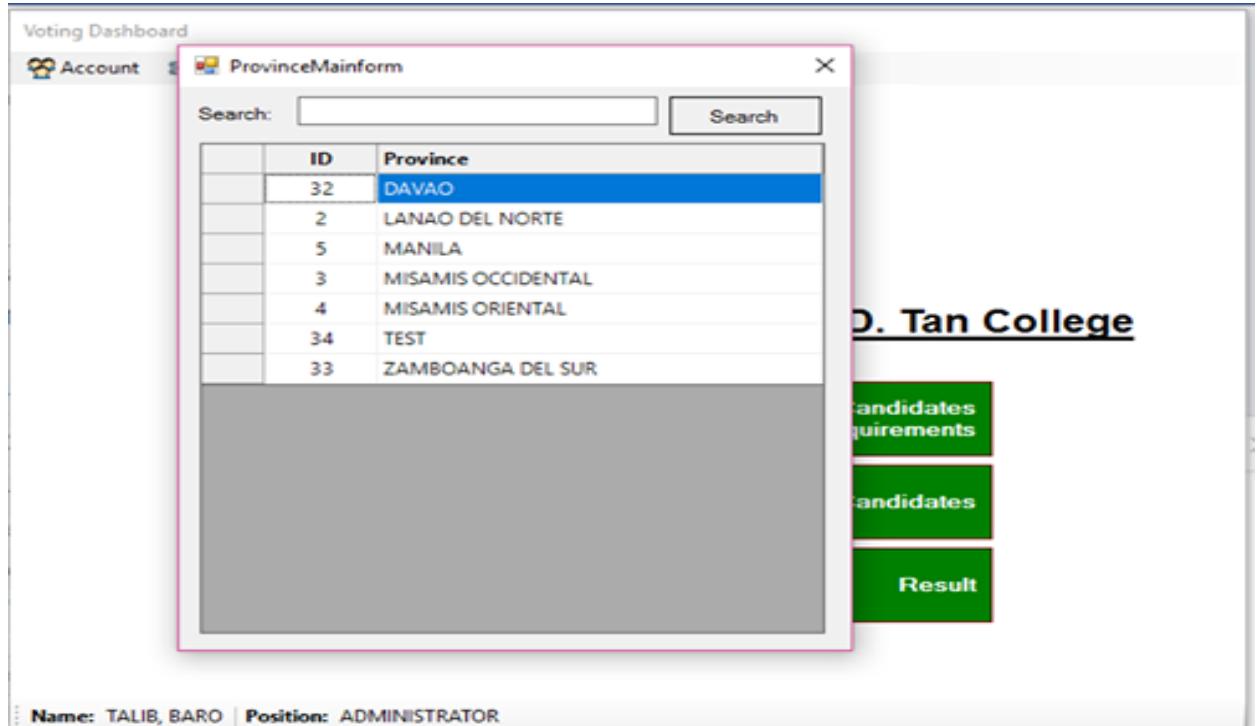


Figure 117: Province Main Form

Add/Modify Province

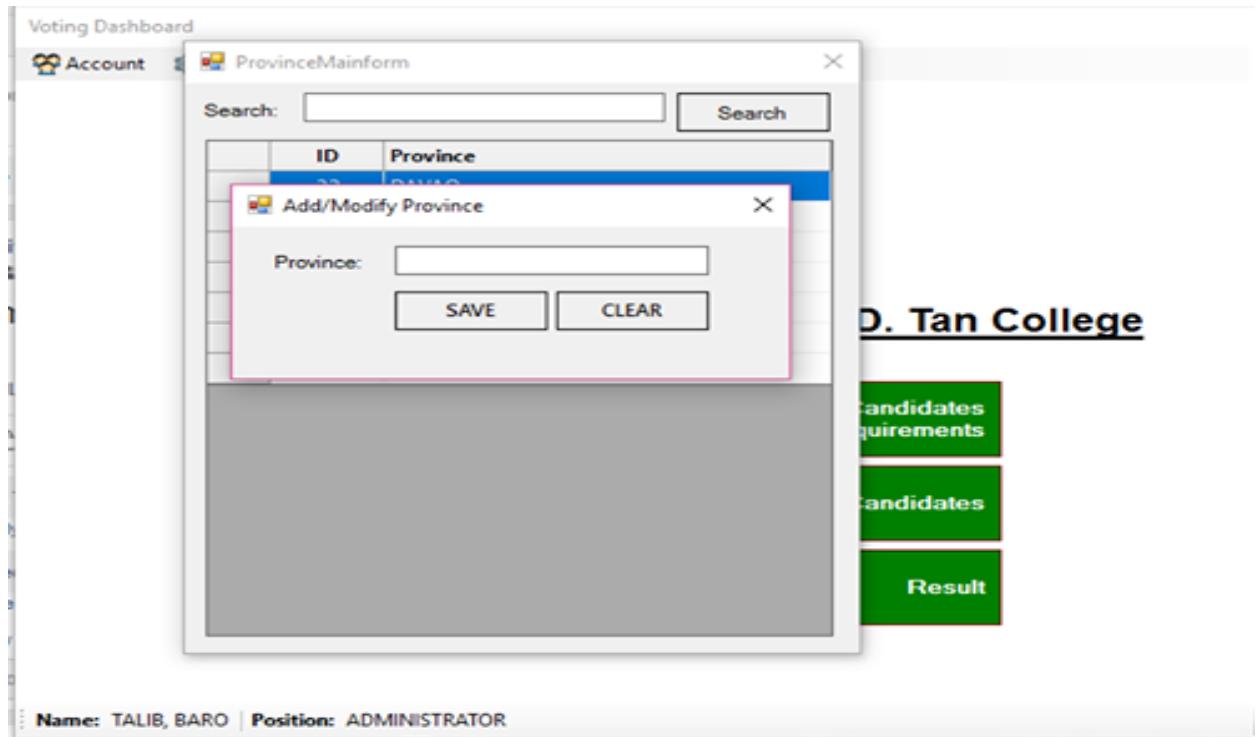


Figure 118: Add/Modify Province

City/Town Main Form

The screenshot shows a Windows application window titled "CityTownMainform". At the top, there are two text input fields: "Province" and "Zip:", followed by a "Search" button. Below these is a grid table with four columns: "ID", "Province", "City/Town", and "Zip Code". The data in the table is as follows:

ID	Province	City/Town	Zip Code
30	MISAMIS ORIENTAL	ALUBUID	9018
15	MISAMIS OCCIDENTAL	BALIANGAO	7211
52	LANAO DEL NORTE	BAROY	9210
67	ZAMBOANGA DEL SUR	BAYOG	7011
17	MISAMIS OCCIDENTAL	CALAMBA	7210
18	MISAMIS OCCIDENTAL	CLARIN	7201
19	MISAMIS OCCIDENTAL	CONCEPCION	7213
68	ZAMBOANGA DEL SUR	DIMATALING	7032
37	MISAMIS ORIENTAL	GINGOOG	9014
73	ZAMBOANGA DEL SUR	GUIPOS	7042
53	LANAO DEL NORTE	ILIGAN	9200
39	MISAMIS ORIENTAL	INITAO	9022
74	ZAMBOANGA DEL SUR	JOSEFINA	7027
41	MISAMIS ORIENTAL	KINOGUITAN	9010
76	ZAMBOANGA DEL SUR	LABANGAN	7017

Figure 119: City/Town Main Form

Add City/Town

The screenshot shows the same "CityTownMainform" window. A modal dialog box titled "CityTownAdd" is open over the main grid. The dialog contains three input fields: "Province", "City/Town", and "Zip:". Below these fields are two buttons: "Save" and "Clear". The main grid table remains the same as in Figure 119.

Figure 120: Add City/Town

Barangay Main Form

Barangay:				Search
	ID	City/Town	Zip Code	Barangay
	1	BAROY	9210	POBLACION
	2	BAROY	9210	BAROY DACU
	3	TANGUB CITY	7214	MALORO
	6	TANGUB CITY	7214	MIGCANAWAY
	7	TANGUB CITY	7214	BARANGAY 1
	8	LALA	9211	MARANDING
	9	KOLAMBUGAN	9207	RIVERSIDE

Figure 121: Barangay Main Form

Add Barangay

Barangay:				Search
	ID	City/Town	Zip Code	Barangay
	1	BAROY	9210	POBLACION
	2	BAROY	9210	BAROY DACU
	3	TANGUB CITY	7214	MALORO
	6	TANGUB CITY	7214	MIGCANAWAY
	7	TANGUB CITY	7214	BARANGAY 1
	8	LALA	9211	MARANDING
	9	KOLAMBUGAN	9207	RIVERSIDE

Figure 122: Add Barangay

Modify Barangay

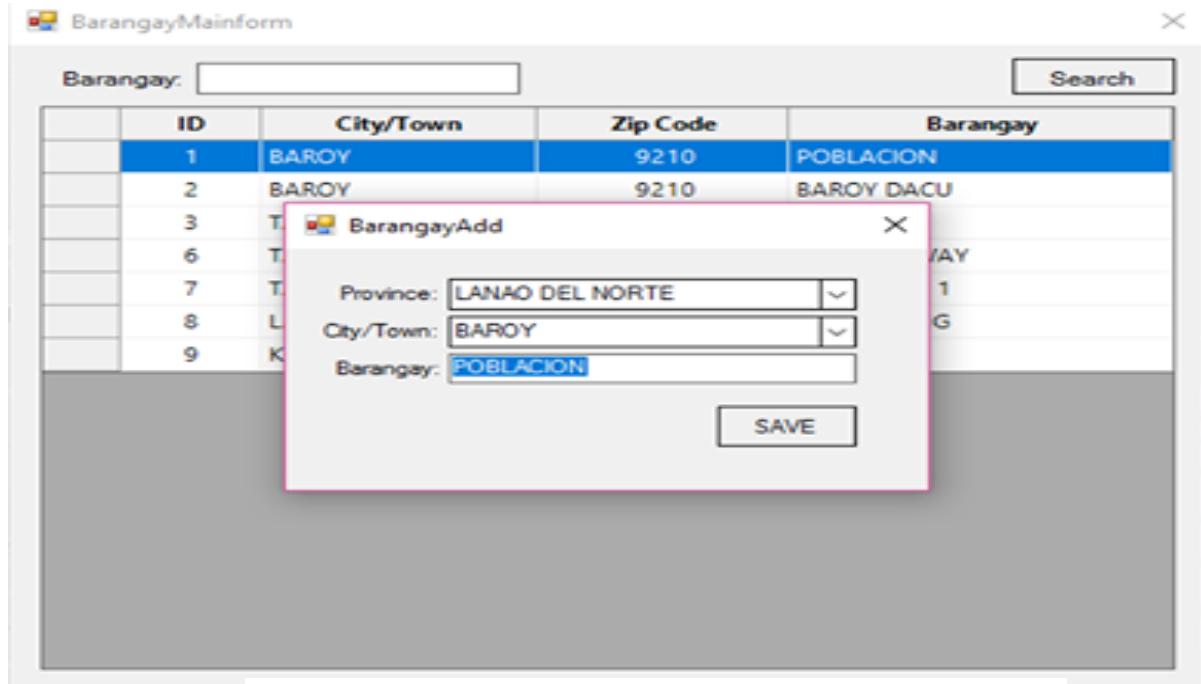


Figure 123: Modify Barangay

Institute Main Form

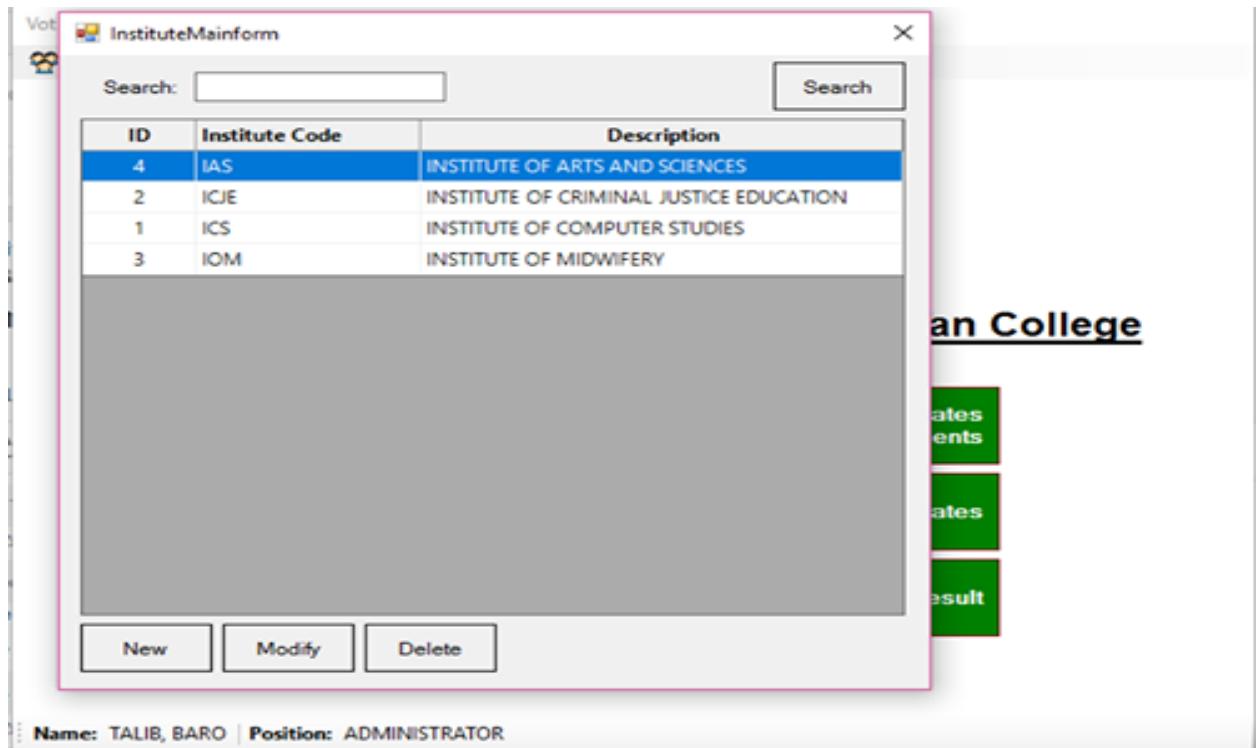


Figure 124: Institute Main Form

Add/Modify Institute

The screenshot shows a Windows application window titled "InstituteMainform". At the top left is a search bar labeled "Search:" with a "Search" button to its right. Below the search bar is a table with columns "ID", "Institute Code", and "Description". The table contains four rows with data: ID 4 (IAS), ID 2 (ICJE), ID 1 (ICS), and ID 3 (IOM). The row for IAS is highlighted with a blue background. A modal dialog box titled "Add/Modify Institute" is overlaid on the main form. It contains fields for "Institute Code" and "Description", both of which are currently empty. A "Save" button is located at the bottom right of the modal. At the bottom of the main form are three buttons: "New", "Modify", and "Delete".

Figure 125: Add/Modify Institute

Program Main Form

The screenshot shows a Windows application window titled "Programs". At the top left is a search bar with a "Search" button to its right. Below the search bar is a table with columns "ID", "Institute", "Program Code", and "Program Title". The table contains eight rows of data. The first row, with ID 13 (IAS) and Program Code AB-COM, is highlighted with a blue background. The other rows show various program codes like AB-ENGLISH, AB-POLSCIE, ACT, BSCRIM, BSCS, and GEN MID, along with their respective titles.

ID	Institute	Program Code	Program Title
13	IAS	AB-COM	AB-COM
14	IAS	AB-ENGLISH	AB-ENGLISH
12	IAS	AB-POLSCIE	AB-POLSCIE
2	ICS	ACT	ASSOCIATE IN COMPUTER TECHNOLOGY
3	ICJE	BSCRIM	BACHELOR OF SCIENCE IN CRIMINOLOGY
1	ICS	BSCS	BACHELOR OF SCIENCE IN COMPUTER SCIENCE
11	IOM	GEN MID	GENERAL MIDWIFERY

Figure 126: Program Main Form

Add/Edit Programs

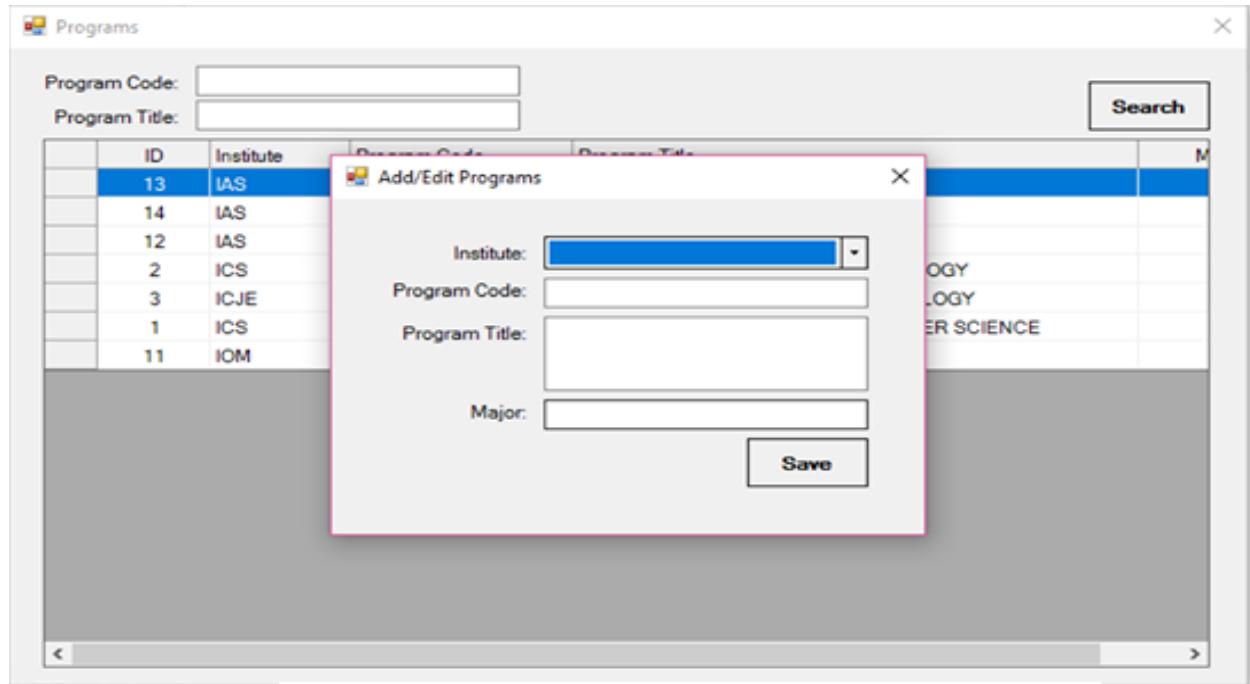


Figure 127: Add/Edit Programs

Position List

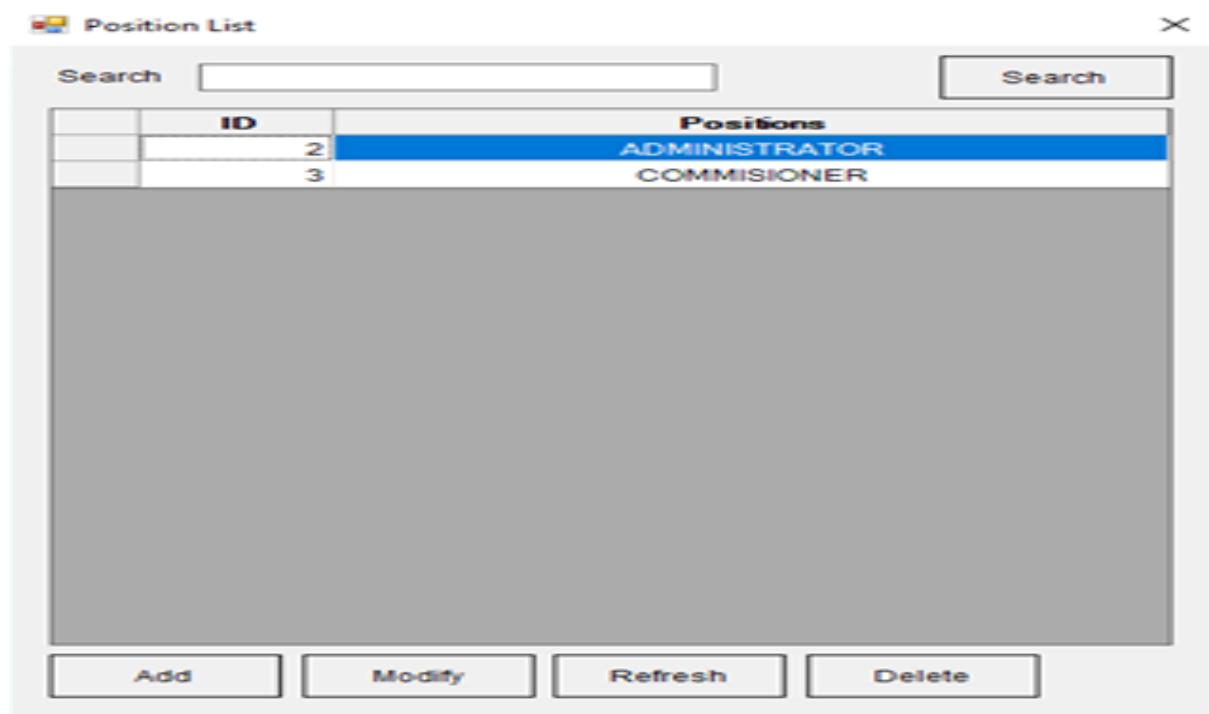


Figure 128: Position List

Add/Modify Position

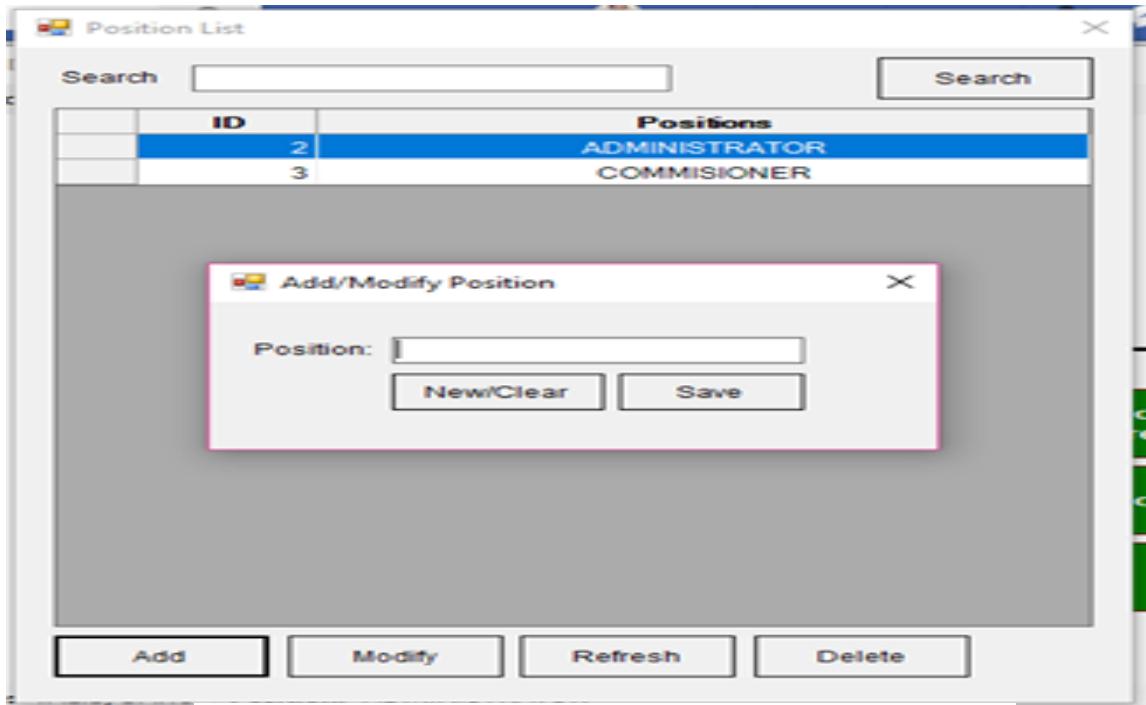


Figure 129: Add/Modify Position

Election Setup

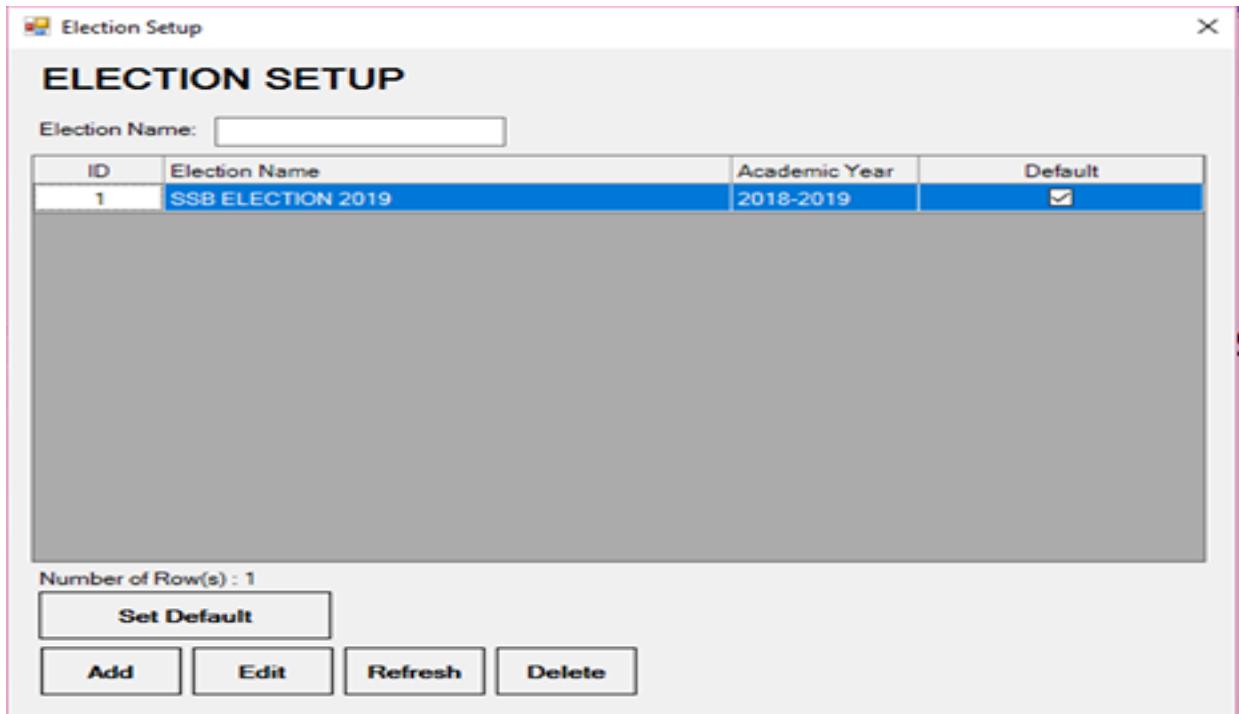


Figure 130: Election Setup

Add/Edit Election

The screenshot shows the 'ELECTION SETUP' window. At the top, there is a table with columns: ID, Election Name, Academic Year, and Default. A row is selected with ID 1, Election Name 'SSB ELECTION 2019', Academic Year '2018-2019', and Default checked. Below the table is a modal dialog box titled 'Add/Edit Election'. It contains three input fields: 'Academic Year' (dropdown), 'Election Name' (text input), and 'Election Date' (date input set to '2019-03-24'). A 'Save' button is at the bottom right of the dialog. At the bottom of the main window, there is a button labeled 'Set Default'.

Figure 131: Add/Edit Election

Voters List

The screenshot shows the 'Voters List' window. At the top, there is a search bar with fields for 'ID No.', 'Lastname', 'Firstname', 'Program', and a 'Search' button. Below the search bar is a table containing 101 rows of voter data. The columns are: Agency No., Lastname, Firstname, Middlename, Program, Year Level, and Birthdate. The last row of the table is highlighted with a blue background. At the bottom of the table are three buttons: 'Add' (green plus icon), 'Edit' (pencil icon), and 'Delete' (red X icon). To the right of the table, it says 'No of Rows: 101'.

Figure 132: Voters List

Position List

Position List

Position	Description	Election Name	Academic Year	Order No
GOVERNOR	GOVERNOR	SSB ELECTION 2019	2018-2019	1
CEO	CEO	SSB ELECTION 2019	2018-2019	2
VICE CEO	VICE CEO	SSB ELECTION 2019	2018-2019	3
SECRETARY	SECRETARY	SSB ELECTION 2019	2018-2019	4
TREASURER	TREASURER	SSB ELECTION 2019	2018-2019	5
AUDITOR		SSB ELECTION 2019	2018-2019	6
EXTERNAL AFFAI	EXTERNAL AFFAIRS OFFICER	SSB ELECTION 2019	2018-2019	7
SOCIO CULTURAL	SOCIO CULTURAL OFFICER	SSB ELECTION 2019	2018-2019	8
SPORTS DEVELO	SPORTS DEVELOPMENT OFFICI	SSB ELECTION 2019	2018-2019	9
ADMINISTRATOR	ADMINISTRATOR	SSB ELECTION 2019	2018-2019	10

New **Update** **Delete** **Search**

Figure 133: Position List

Add/Edit Position

Position List

Position	Description	Election	Academic Year	Order No
GOVERNOR	GOVERNOR	SSB ELECTION 2019	2018-2019	1
CEO	CEO	SSB ELECTION 2019	2018-2019	2
VICE CEO	VICE CEO	SSB ELECTION 2019	2018-2019	3
SECRETARY	SECRETARY	SSB ELECTION 2019	2018-2019	4
TREASURER	TREASURER	SSB ELECTION 2019	2018-2019	5
AUDITOR		SSB ELECTION 2019	2018-2019	6
EXTERNAL AFFAI	EXTERNAL AFFAIRS OFFICER	SSB ELECTION 2019	2018-2019	7
SOCIO CULTURAL	SOCIO CULTURAL OFFICER	SSB ELECTION 2019	2018-2019	8
SPORTS DEVELO	SPORTS DEVELOPMENT OFFICI	SSB ELECTION 2019	2018-2019	9
ADMINISTRATOR	ADMINISTRATOR	SSB ELECTION 2019	2018-2019	10

New **Update** **Delete** **Search**

Add/Edit Position

Election:	<input type="text"/>
Position:	<input type="text"/>
Description:	<input type="text"/>
Order:	<input type="text"/>
# of Winner(s):	<input type="text"/>
Can Vote:	<input type="text"/>
Save	

Figure 134: Add/Edit Position

Candidate's List

The screenshot shows a Windows application window titled "Candidate List". At the top, there are search fields for "Lastname:" and "Firstname:" and a dropdown menu for "Positions:". Below these is a table with columns: ID, Lastname, Firstname, Middlename, Position, and Election Name. The table contains 20 rows of data. Row 20, which corresponds to the highlighted row, has the following values: ID=20, Lastname=GUMOLON, Firstname=RODRIGO JR., Middlename=BALITON, Position=SPORTS DEVELOPMENT OFF, and Election Name=SSB ELECTION 2019. At the bottom of the table are four buttons: "New Candidate", "Modify Candidate", "Refresh", and "Delete Candidate".

ID	Lastname	Firstname	Middlename	Position	Election Name
1	ADONA	CHRYPTON	RENZ	GOVERNOR	SSB ELECTION 2019
2	ALAB	MARVILOWE	TORAJA	CEO	SSB ELECTION 2019
12	ALCALA	JACQUELINE	TULO	ADMINISTRATOR	SSB ELECTION 2019
7	ALEGRE	LEE BAVERLY	ENCLONAR	SECRETARY	SSB ELECTION 2019
13	AUXTERO	ZYRA JEAN	BONGCALES	ADMINISTRATOR	SSB ELECTION 2019
16	BANAYBANAY	JONA	APLACA	AUDITOR	SSB ELECTION 2019
23	BOLISIG	ARALICIA MAE		GOVERNOR	SSB ELECTION 2019
19	CABASAG	MARITCHO	ANTIPUESTO	SPORTS DEVELOPMENT OFF	SSB ELECTION 2019
5	CABRERA	SIRFRED GEE	TANEZA	VICE CEO	SSB ELECTION 2019
10	DUERME	CHELLA MAE	RUSIANA	SOCIO CULTURAL OFFICER	SSB ELECTION 2019
6	DURIAS	CHRISTITA	GLOVASA	VICE CEO	SSB ELECTION 2019
11	ENERO	LYKA	PALACIO	ADMINISTRATOR	SSB ELECTION 2019
20	GUMOLON	RODRIGO JR.	BALITON	SPORTS DEVELOPMENT OFF	SSB ELECTION 2019
25	HAMIS	JOSEPH RIEL	ECHAVIA	ADMINISTRATOR	SSB ELECTION 2019
14	HIBAYA	NIMCHIE	PALMA	ADMINISTRATOR	SSB ELECTION 2019
21	LARANJO	DENLORYN	MONTON	TREASURER	SSB ELECTION 2019
9	LOPEZ	ANGEL	PUEBLAS	EXTERNAL AFFAIRS OFFICER	SSB ELECTION 2019
17	MAGUILONG	MARY CAIMDV ADDIE	RASTOR	EXTERNAL AFFAIRS OFFICER	SSB ELECTION 2019

Figure 135: Candidates List

Add/Modify Candidates

The screenshot shows a modal dialog box titled "Add/Modify Candidates" overlaid on the "Candidate List" window. The dialog box contains fields for "Position" (dropdown), "Description" (text area), "Election Type" (dropdown), "Voters ID" (text area with a browse button), "Lastname" (text area), "Firstname" (text area), "Middlename" (text area), and a "Save" button. The background "Candidate List" window shows a grid of candidate data, with row 20 highlighted. The "Election Name" column for row 20 shows "MENT OFF".

Figure 136: Add/Modify candidates

Setup Candidates Requirements

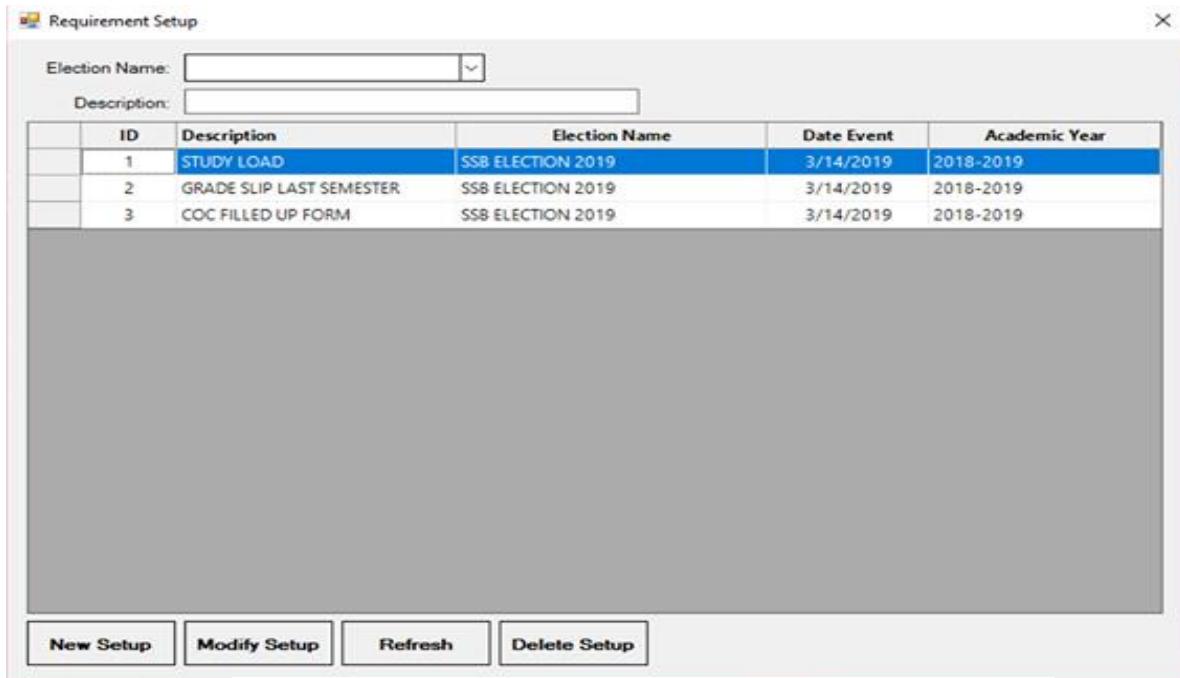


Figure 137: Setup Candidates requirements

Add/Modify Requirements

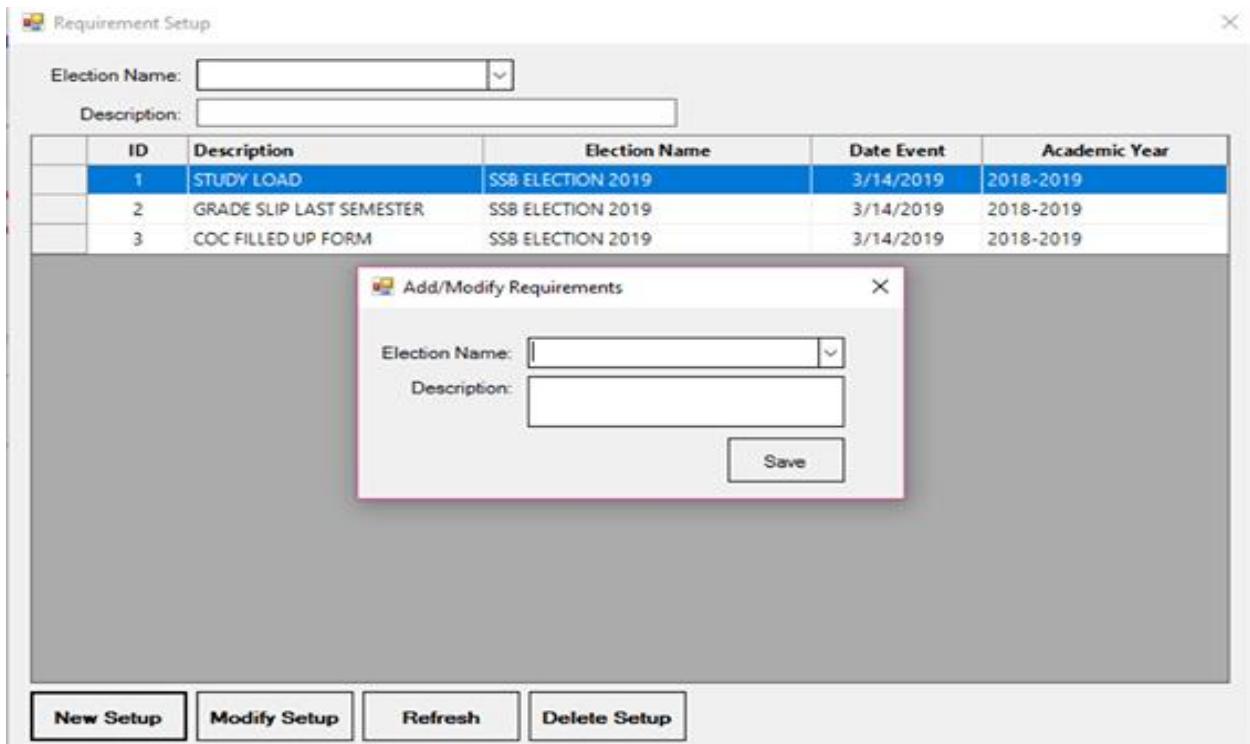


Figure 138: Add/Modify Requirements

Printed Voted List

Voters ID	Lastname	Firstname	Middlename	Institute
26	ALAB	MARVILOWE	TORAJA	ICS
75	ALCALA	JACQUELINE	TULO	ICS
38	ALDUHESA	FEEL MARY	CORDOVA	ICS
19	ALEGRE	LEE BAVERLY	ENCLONAR	ICS
61	ALGADIPE	HEZZLE	BARINAN	ICS
10	ALIA	ALBERT	BORJA	ICS
93	ALIAS	ROSELYN	MACASARTE	ICS
44	ASON	GENELIZA	CATULAG	ICS
74	AUXTERO	ZYRA JEAN	BONGCALES	ICS
89	BALAYONG	JUSTINE JED	ASARI	ICS
77	BANAYBANAY	JONA	APLACA	ICS
11	BANO	ANNIE JANE	MAGLANGIT	ICS
51	BARHIYO	JOHNA	ANSAO	ICS

Figure 139: Print Voted List

Printed Not Voted List

Voters ID	Lastname	Firstname	Middlename	Institute
6	LANTERNA	RICHARD	A.	ICS
27	PENAS	BERNARD	SABALA	ICS
78	TORRION	ROXETH BIANCA	SAQUIN	ICS
86	VIVAS	ROSE-ANN		ICS
87	NOYA	IRINE	CABACOY	ICS
94	AMPARADO	ETIENNE WAYNE	NAMOCATCATG	ICS

Nothing follows

Figure 140: Print Not Voted List

Print Result

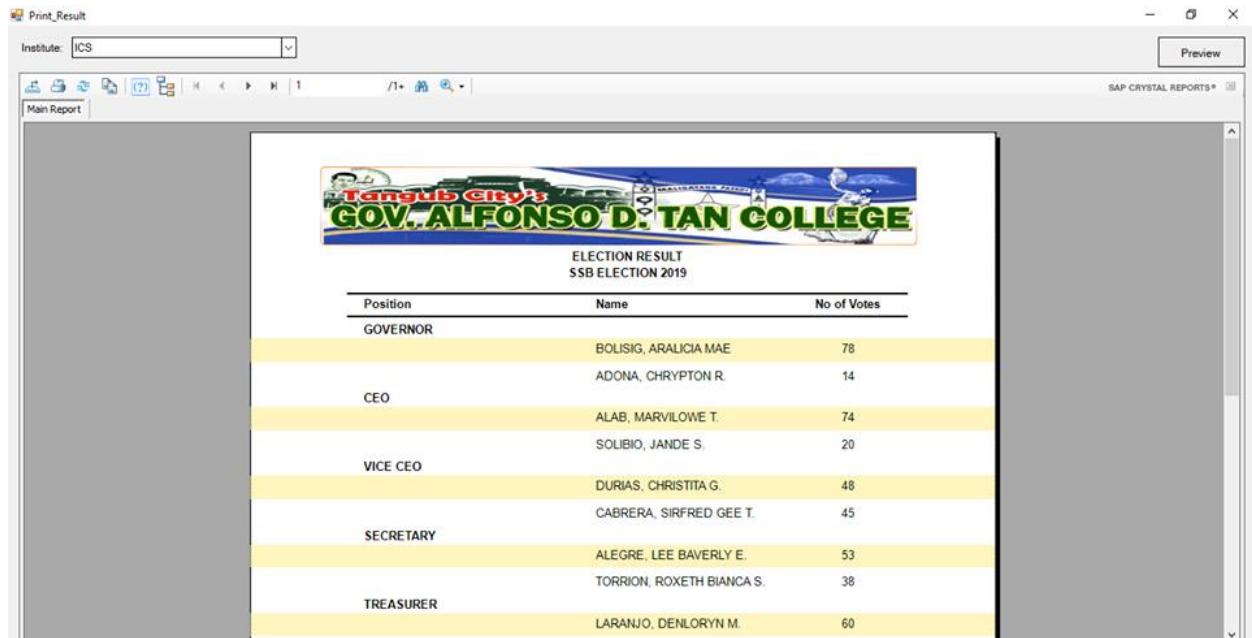


Figure 141: Print Result

GUI (Graphical User Interface)

Voters Side

Login

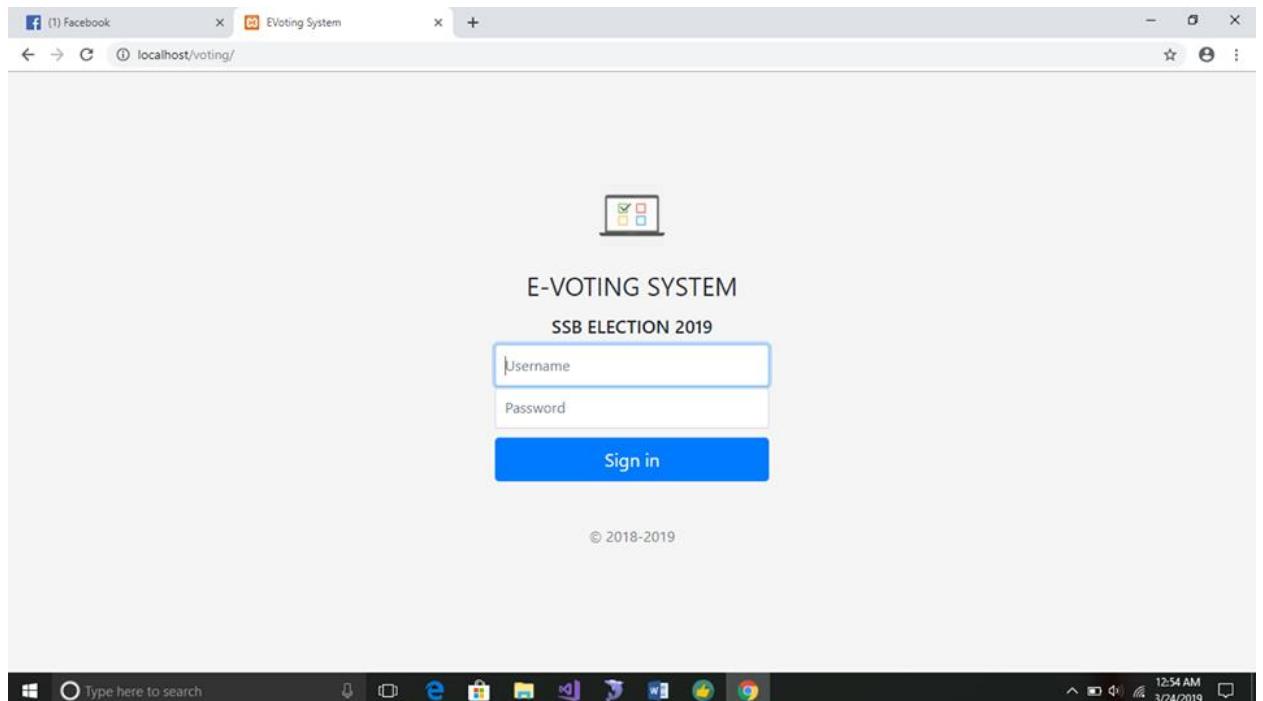


Figure 142: Login (Voter Side)

Homepage

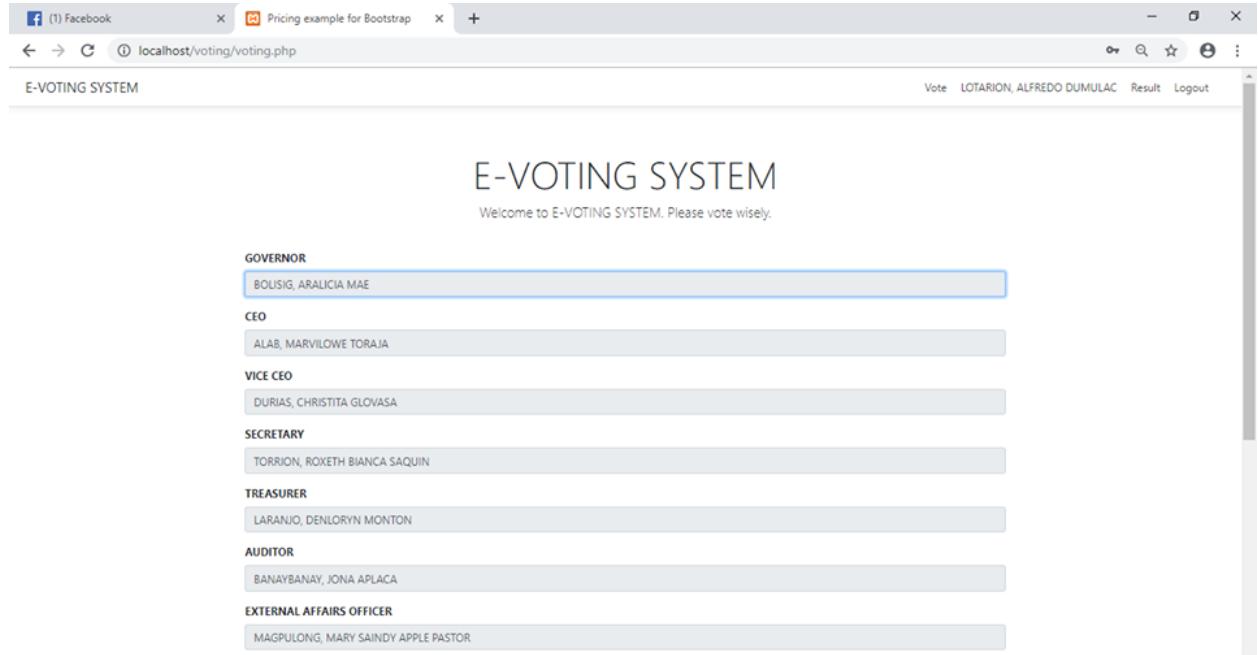


Figure 143: Homepage (Voter Side)

Vote for Candidate

The screenshot shows a web browser window with the URL `localhost/voting/voting.php`. The page title is "E-VOTING SYSTEM". The main content is a form titled "Vote for Candidate" with dropdown menus for selecting candidates:

- SECRETARY**: Nothing selected
- TREASURER**: Nothing selected
- AUDITOR**: Nothing selected
- EXTERNAL AFFAIRS OFFICER**: Nothing selected
- SOCIO CULTURAL OFFICER**: Nothing selected
- SPORTS DEVELOPMENT OFFICER**: Nothing selected
- ADMINISTRATOR**: Nothing selected

At the bottom of the form is a green "SUBMIT" button.

Figure 144: Vote for a Candidates (Voter Side)

Voter's Profile

The screenshot shows a web browser window with the URL `localhost/voting/profile.php`. The page title is "E-VOTING SYSTEM". The main content area displays a profile for a voter named "RICHARD". The profile includes a placeholder image labeled "Your avatar", dropdown menus for "Others", "Institute ICS", and "Programs & Year BSCS - 4", and a "Mobile" number field containing "09270558114". The voter's details are listed as follows:

Firstname	RICHARD
Middlename	A.
Lastname	LANTERNA
Mobile	09270558114
Address	[empty]

At the bottom left, there is a small logo and the text "© 2017-2018". At the top right, there are links for "Vote", "LANTERNA, RICHARD A.", "Result", and "Logout".

Figure 145: Voter's Profile (Voter Side)

View Result

The screenshot shows a web browser window with the URL `localhost/voting/result.php`. The page title is "E-VOTING SYSTEM". The main content area displays the results of the "SSB ELECTION 2019" with the heading "RESULT". The results are presented in a table:

POSITION	Name	Votes
GOVERNOR	BOLISIG, ARALICIA MAE	78
GOVERNOR	ADONA, CHRYPTON RENZ	15
CEO	ALAB, MARVILOWE TORAJA	74
CEO	SOLUBIO, JANDE SALADAGA	20
VICE CEO	DURIAS, CHRISTITA GLOVASA	48
VICE CEO	CABRERA, SIRFRED GEE TANEZA	45
SECRETARY	ALEGRE, LEE BAVERLY ENCLONAR	53
SECRETARY	TORRION, ROXETH BIANCA SAQUIN	38
TREASURER	LARANJO, DENLORYN MONTON	60
TREASURER	MANLUNAS, MICHEL ABITONA	33
AUDITOR	RAMAYRAT, JANICE CANIALIS	51

Figure 146: View Results (Voter Side)

REFERENCES

Oyite David Robinson(2011), *Mobile Voting Systems*. Retrieved from
<http://mobilevotingsystems.blogspot.com>

Abdooz (2013), *Mobile Voting System*. Retrieved from
<http://www.studymode.com/essays/Mobile-Voting-System-1563752.html>

MarkAngel(2011) *PSU Bayambang Campus Voting System*. Retrieved from
<http://www.studymode.com/essays/Psu-Bayambang-Campus-Voting-System-708937.html>

Divya (2014), *advanced online voting system*. Retrieve from
<https://docplayer.net/44361849-Advanced-online-voting-system.html>
Robinson (2011) Mobile Voting System. Retrieve from
<http://www.academia.edu/32592256/thesis.docx>

MohammedSidqi (2014), *Voting System Using SMS*. Retrieve from
https://www.researchgate.net/publication/269207469_A_Novel_Voting_System_Using_SMS