# Chapter 1

# Introduction

## 1.1 Overview

Voting is one of the fundamental duties of the community . Election allows the people to choose their representatives and express their favorites for how they will be governed. Naturally, the integrity of the election process is fundamental to the integrity of democracy itself.

Election in the history of wollo University has been held for many times. But, still students are complaining on the announcement of the final results; because of the thought they have on fraudulent activities. The election system must be sufficiently robust to resist a variety of fraudulent behaviors and must be sufficiently clear and understandable that voters and candidates can accept the results of an election. To achieve all this, ICT technology is required.

Nowadays, the application of software engineering is introduced at several domains of fields. Its’ multidimensional benefits is becoming more visible from time to times. The economical benefit gained from the technology is the most significant one. Furthermore, it helps to increase the qualities of the work, reduces the complexities of tasks, keeps the security of data in most favorable condition, and makes data transfer more easily.

We are interested to design a web based application which enable students to elect their representatives in the more reliable way than manual voting system.

## Background Information

Wollo University is one of the public higher educational institutions established at dessie in 1997. It is located 400 km West of Addis Ababa at the outskirt of dessie town on the 150 hectares of

Student union of wollo University established at the same time when the University is established. The need to have a president, vice president, general secretary, leaders of the colleges and others members lead the need to conduct election. Wollo university election board which is the main area of our study is the one who is responsible to conduct this election in every two years. When it is established, it had no any well organized management and no computerized voting system. It has been held yet manually and facing many problems which will be stated on the statements of problem.

This union has three participants; election committee (election officer) is the one who takes care of the overall commitments of the voting process, voters are students who participates in the election by casting his/her vote and candidates are students who participates in election by contesting in it.

## Strength and Weakness of the current system

1.1.1 Strength

Although the system is manual and paper based it has some strengths as per the time of implementation. The strengths are:

* The voters can interact with candidates physically
* Election officers can directly contact all the candidates and voters during the registration and voting process.
* Nobody out of the institution can cast vote, as this is achieved by the help of students’ identification card.
* After the voter gave vote, the election officers attached a sign (using the parker) on his/her thumb; this is especially useful for securing the process.
* Having the mother document on which the candidates and voters are registered with their information.

1.1.2 Weakness

The weakness of the current system includes:

* Many of the votes are not counted due to slight mistake by the voter, e.g. Voter left Thumb impression in between two check boxes etc.
* Presiding Officers count the ballot papers, but human counting is usually sluggish and has chances of error as well; which means an ultimate disaster ahead in the process of Democracy.
* Time is very important factor for measuring the efficiency of the system. The current system requires a lot of time for data process circulation and calculation.
* Problem for election board to manage the records.
* Problem for duty staff to search the record of a particular voter.
* Problem for duty staff to calculate the vote’s results
* Problems in filling the forms.
* Unauthorized vote casting.
* Because the current system cannot enhance the data validation in a meaningful manner, voters often the incomplete and incorrect data.
* A small mistake can cause of rejection of vote.

## 1.1.Statement of the problem

Manual voting has caused some difficulties for voting process and also it has some disadvantages for the students.

The main problems of the current system are:

* Delay announcement of the final result: The time for declaring the final result has also be beyond the schedules because there is a delay on counting activity as it is performed manually.
* Wastage of vote: There are losing of votes due to the fraudulent activities; such as, buying and stolen of vote’s (security threats).
* Some voters doesn’t fulfilled their voting duty: Sometimes students may not have a time to register and vote when there are a lot of students queued and waiting for vote or register. Because of that reason, they do not fulfill their voting duty.
* Wastage of man power and lots of paper: As the volume of the work is massive and wide, it needs a lot of people to participate in the election process and several numbers of papers will be wasted.
* Lack of full information about candidates:voter usually doesn’t know too much detail about the candidates in the time of election, so that they have faced problems of identifying the candidate who is qualified to be.
* It is complex and error prone task:vote counting processes and generates more accurate results within a short time is very complex. And, there might be ticks that are void which leads to errors.
* High Cost:The material cost required for managing the whole voting process is also high.
* Appointment of polling stations is problematic. Polling stations can help in corruption during the polling, so many of the candidates try to appoint their own staff at their polling stations and thus the authorities are approached by such candidates. Increasing number of polling stations raises the number of staff required. Polling station is appointed away from the area where they belong to, thus many of the people could not cast their vote.
* A huge number of ballot papers are printed and distributed on polling stations and this cost billions; not acceptable for a developing countries like Ethiopia.
* There is no check and balance on the timings of casting votes as some influential Candidates let their own people cast vote even after the time is over; as it is normally From 8 AM till 5 PM. So this means people go on voting even after 5 PM.
* Many of the votes are not counted due to slight mistake by the voter, e.g. Voter left Thumb impression in between two check boxes etc.
* Presiding Officers count the ballot papers, but human counting is usually sluggish and has chances of error as well; which means an ultimate disaster ahead in the process of Democracy.
* Time is very important factor for measuring the efficiency of the system. The current system requires a lot of time for data process circulation and calculation.
* Problem for election board to manage the records.
* Problem for duty staff to search the record of a particular voter.
* Problems in filling the forms.
* Unauthorized vote casting.
* Because the current system cannot enhance the data validation in a meaning full manner, voters often the incomplete and incorrect data.
* A small mistake can cause of rejection of vote.

Generally, manual based voting system has bad management and security threats. And, these entire problems can be solved using computerized voting system.

## Objectives of the project

### **1.1.1.General Objective**

The main objective of the project is to automate students’ representative voting system of Wollo University by replacing the current manual system.

### **Specific Objective**

The specific objectives of the system are:

Study the existing system

Develop a new automated system

* **Make operation easy:** The system should be easy to operate and should be such that it can be developed within a short period of time and fit in the limited budget of the user.
* **Increase integrity:** The necessary mechanism should be employed in order to guarantee that no one can duplicate his or someone else's vote (Not duplicability) and no one can change someone else's vote (Not changeability). Protect the fraudulent activities on the vote due to corruption.
* **Increase Accuracy:** The system functionality should ensure that no one can falsify or modify the result of the voting by eliminating a valid vote or counting an invalid vote in the final count.
* **Make convenience:** The system should allow and assist voters to cast their votes quickly.
* **Improve efficiency:** The election can be held in a timely manner (i.e. all computations during the election are done in a reasonable amount of time and voters are not required to wait on other voters to complete the process).
* **Scalable:** The size of the election should not drastically affect performance.
* Testing the developed system.

## Scope and Limitation of the project

The limitations of the system are: it does not concern any other kind of election and does not include mobile based election.

The scope of the study is only for Wollo University students’ representative election. Our study contains the following major activities:

1. Pre\_Voting

* Register candidates based on qualification and provide complete profiles of them.
* Setting starting and ending date for both registration and voting.
* Generating secrete keys and distribute to voters and candidates.

1. Voting(voting process itself)

* Create account.
* Secure login procedure.
* Registrations of voters.
* Review complete candidates profiles.
* Give votes to the alleged candidates.

1. Post –Voting

* Counting votes.
* Generating the final result.
* Reset the election. Allow to conduct election again.

## Feasibility study

A significant transformation in election system has occurred in the world. The online voting system and improvements has been made through recognizing difficulties encountered by the voters and the board. Voters and board members training sessions have influenced the online voting process. As a result this had produced an efficient and user friendly system, that relies on an effective online voting system, but on the coordination between election board members and voters. A comprehensive feasibility study of social, economical and technical aspects has also been made and implemented as below:-

* + 1. Operational feasibility

The new system is operationally feasible because:-

* The new system fits with the existing system.
* Satisfy the user needs or requirements
* Provides the end users and managers with timely, accurate, reliable, flexible and usefully formatted information.
* Provides adequate through put and response time.
* The system offers adequate control to protect against fraud and embezzlement (misuse) to guarantee the accuracy and security of the data and information.

### **Social Feasibility**

It has simplified the election procedure. Voters and election board had a huge acceptance to the idea. It had a good social impact and no problems regarding the project are found

### **Economic Feasibility**

The project is economically feasible since we are getting sufficient free software required for the project from Internet and others materials are covered by the group members.

### **Technical Feasibility**

Minimum requirement for execution of the project is vista or window 8, window 10 and minimum of 4 GB of RAM, 500 GB of Hard Disk, a server, a client, network and a web browser.

## Significance of the system

Significances of the system are:

* Reducing the chances of error occurrence, while counting votes.
* Minimizing time being consumed.
* Increase the security of votes.
* Eliminating the chance of vote’s rejection.
* Easy management of records.
* Make searching of desire candidate record easy.
* The polling is online by being wherever.
* To know easily whether the voter is registered or not.
* High and reliable security can be achieved.
* Incomplete and incorrect data about voters or candidates cannot be registered.
* It supports data validation
* Reduces cost by minimizing the number of peoples participated in the election process.
* It contributes some to growth of democracy in Ethiopia too much.
* Voters can learn details about the candidates and they will be interacting with each other before the Election Day.
* Small response time and many services
* Easy and fast voting process management.
* It increases the counting speed.
* It saves a lot of time and money wasted by traditional paper ballots.
* It facilitates the vote counting processes and generates more accurate results within a short time; thanks to the computer technology.
* It speedup the registration process because more than one voter can register at one time.

## Beneficiaries of the system

Up on the successful completion of the project, the beneficiaries of the project are voters, candidates, election officers, members of the project team and Wollo University.

* The project team got benefit out of the project by getting software development experience.
* Voters can learn complete profile about the candidates.
* The new system makes the system users benefited by providing user friendly interface, decreasing their work load.
* Voters and candidates can easily ensure whether they are registered or not
* Voter and candidates are beneficial by gaining efficient and effective service out of the new system.
* Finally the Wollo University Student Union would get value from the new system by minimizing cost via minimizing the number of peoples participated in the election process, by rendering services which fulfill user requirements

## Methodology of the project

### **System development methodology**

Waterfall model

The waterfall model is a software development model in which a system’s development is viewed as flowing downwards through the phases of the system development process**.**

**Advantages**

* Good for large projects
* Waterfall suits a principled approach to design
* Waterfall divides the project into manageable areas

### **Data collection methodology**

* + - 1. Data Source

To learn the details of the system, we used both primary and secondary data sources including working place and documents.

* + - 1. Data Collection

We used both primary and secondary data collection techniques. The primary data collection techniques are: interview, observation and questionnaire; and secondary data collection technique is document analysis. We used the following data collection methodologies throughout the project:

* Document analysis
* It has been used to get business objectives
* Understanding the business rule
* Business form-used for many functions example, candidate and voter registration, vote, and report.
* Study the procedures manual and written documents on Wollo University Students’ Union Election Board, to obtain information on how the existing system works.

* Interview
* To find where problems are in the system
* What data they used to their work
* Important questions were raised to understand the system.
* Observation
* The team has used observation to get information that cannot be gained using the above methodologies (interview and document analysis) such as how the voters, candidates and election committees act in work area, because the team is the part of this society.
* Questionnaire
* This methodology has been used to collect the users’ ideas through the form of questionnaire.

You can refer the questionnaire attached in the appendix.

### **Development Environment/Programming Tools and Other Tools**

Hardware requirement

For the successful run of the proposed system, the personal computer should have the following minimum hardware requirement:

* A processor of Pentium , hard Disk 500GB and 4 GB of RAM.
* Flash 16 GB
* Disctop

Software requirement

The minimum software requirements are

* java
* Gitub
* Enterprise Architecture .
* Microsoft word 20013, and PDF Writer: are used for reporting and preparing the documentation.
* Snipping Tool
* Microsoft power point 20013 modeling tool: is used for drawing all the UML diagrams.

## Organization of the Document

This document is divided in to three chapters:

Chapter one is about introduction of the study. This chapter deals with background information of the current system including its strengths and weaknesses. The other components of the chapter are statement of the problem, objectives (general and specific), scope and limitation of the project, feasibility analysis (operational, technical and economic feasibilities) and significance as well as methodologies used in data collection season.

Chapter two is about requirements and system modeling. It deals with what kind of functionality did the users expect from the system and what quality requirement the system should achieved. It contains some behavioral and structural UML diagram type. From behavioral diagram, it possesses use case and sequential diagram. And from structural diagram, it contain conceptual class diagram. It also specifies the business rule of the new system.

1.14 Work breakdown

The table below represents the main activities of the project together with their respective start and end date

Table 1 for time schedule of the project

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Tasks to be done | March | April | May | June | Date |
| 1 | Proposal |  |  |  |  | 10-29/07/11 |
| 2 | Functional Requirement |  |  |  |  | 01-30/08/11 |
| 3 | Implementation and Testing |  |  |  |  | 1-9/09/11 |
| 4 | Conclusion |  |  |  |  | 10-21/09/11 |

**Chapter Two**

# Requirements and System Modeling

## 2.1 Functional Requirement

The functional requirements are functions or features that the system must include to satisfy the user need and to be acceptable by the user . The functional requirements of the new system include:

* Datastorage, count and retrieval: All the voters and candidates’ information should be kept properly in well organized database; so that counting vote and retrieving records from the database will be easy and faster.
* The system should register voters and candidates.
* The system should provide cast vote facility.
* The system should be able to generate report including candidates’ and voters’ information, progress and final result.
* The system should provide Searching, and deleting records mechanism (this is useful whenever the election needs to be held again).
* The system should be flexible to get data of voters and candidates whenever required.
* System do online vote polling.
* Result compilation i.e. correct result will be generated.

## 2.2 Non-Functional Requirement

The non-functional requirement describes constrains for implementing the project . Some of them are; the central server has to be provided at secured area, the system must be maintainable and expandable, the network infrastructure has to be private network, client machines anywhere inside the campus could access the system. The voter and candidates should have also basic computer skills. The input value which is used to generate the encryption key must be provided from the election officials, and needs to be kept securely.

Non-Functional requirements of the system described as follow:

* The software will be user friendly.
* It will have the standard graphical user interface.
* Security of data.
* The system should provide user authentication mechanism that it provides an access only to an authorized user through account.
* The system restricted the entire voters from voting more than the specified in business rule identification section.
* Establish communication to the database must be done by username and password.
* Only the system should count the final result and report it.
* Ensure system development and coding is well documented for future use.

### **2.2.1 User Interface**

The users want the interface to be graphical interface and to have drop down menus. In addition the members require the interface to be attractive and user-friendly.

### **2.2.2 Security Issue**

A top priority requirement for the proposed system is to have highly efficient and secure features, to safe guard the integrity of the voting system. With the system recording confidential information, users (voters and candidates) other than Election officers can only access part of the system that is relevant to them. In order to make the system secure, two forms of security measures where engineered into the system to safeguard the data flow within the system and the information being stored in the database.

### **2.2.3 Error handling issue**

This system handles error done by the user giving error message when the user enters wrong inputs. In addition, the system provides error handling mechanism on the ways system interacts with database and input accept or fields. To reduce input fault, the system enable the user to confirm that details are correct before creation, process or modification occurs. Respond to error inputs by asking the user to reenter data in the correct format.

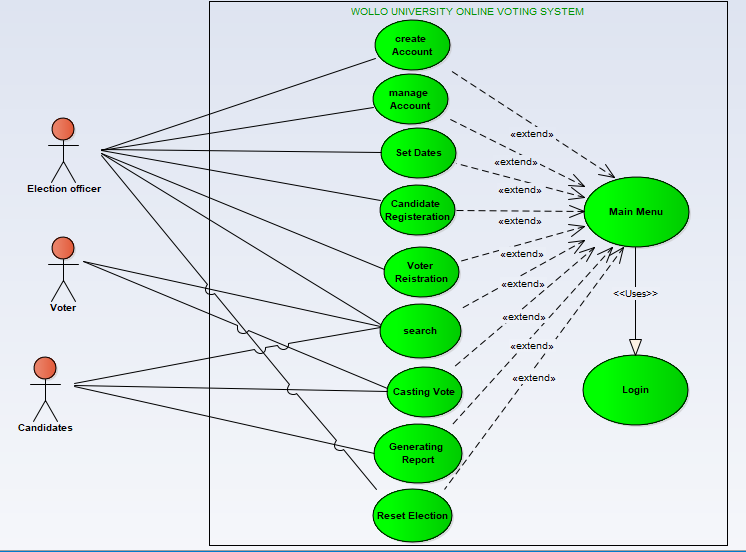
### **2.2.4 Availability**

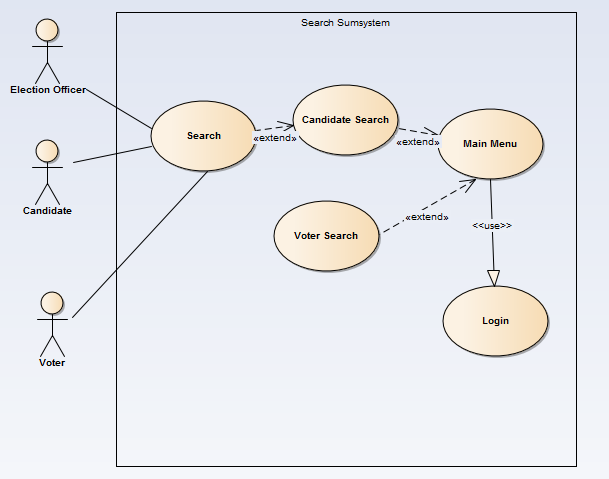
The system should easily be available at any desired time. If by any chance the system fails, customers will get difficulties on getting information.

## 2.3 System Modeling

### **2.3.1 Use Case Diagram**

A use case is an interaction between users and a system. It captures the goal of the users and the responsibility of the system to its users .





#### **areb3**

#### **2.3.1.1 Actors**

Actors represent system users. An actor is someone or something that participates in the election process .

The system has three players: Election Committees (Election Officers), Voter, Candidate

1. Election Officer is the one who takes care of the overall commitments of the voting process.
2. Voter is a student who participates in the election by casting his/her vote.
3. Candidate is a student who participates in election by contesting in it.

Table 2.1 General Usecase **Description**

|  |  |  |
| --- | --- | --- |
| **Name** | **Identifier** | **Description** |
| Set Date | UC1 | Used to set the starting and ending date of both for registration and voting. |
| Manage Account  1. Create Account  2. Delete Account  3. Update Account | UC2 | Contain three sub use cases |
| UC2.1 | Enable users to create a user account |
| UC2.2 | Enables election officer to delete an account |
| UC2.3 | Enables users to update an account |
| Login | UC3 | Enable all user to login into the system |
| Candidate Registration | UC4 | Enable election officer to register election Candidates |
| Voter Registration | UC5 | Enable voters to register themselves |
| Search  1. Voter Search  2. Candidate Search | UC5 | Contain two sub use cases |
| UC6.1 | Enables all users to search a particular voter |
| UC6.2 | Enables all users to search a particular Candidates |
| Report | UC7 | Enables users to generate the aggregate voter report, aggregate candidate report, to generate counted result of the candidates and candidates’ progress. |
| Casting Vote | UC8 | Enable voters and candidates to give votes to alleged candidates |

### **2.3.2 Essential use case Documentation**

The process of a business is defined as a number of different business use cases, each of which represents a specific work flow and has its own target in the business [2]. A business use case defines what should happen in the business, what conditions must first fulfill, what will be the outcome/result and others when it is performed.

From Table 2.1to Table 2.11 show the use case description of the system.

Table 2.2: Set Date use case description

|  |  |
| --- | --- |
| Name | Set Date |
| Identifier | UC1 |
| Description | Enable Election officer to set the starting and ending dates of both registration and voting. |
| Actor | Election officer |
| Precondition | Election officer should have to come with the dates which are approved by election committee. |
| Post condition | Dates for election have set. |
| * Alternate actions(A): | A1. If the entered dates are the past or fields are not correctly filled then the system request the applicant to enter again.  A2. The use case resumes at step four of flow of events. |

Table 2.3: Create Account sub use case description

|  |  |
| --- | --- |
| Name | Create Account |
| Identifier | UC2.1 |
| Description | Enable users to create an account |
| Actor | Election officer, voter and candidate |
| Per condition | Actors should have to come with a privilege key. |
| Post condition | Account will be created and given to the actor who made these actions. |
| * Alternate actions(A): | A1. If the entered security key does not exist in the database or other fields are not correctly filled then the system request the applicant to enter again.  A2. The use case resumes at step four of flow of events. |

Table 2.4: Update Account sub use case description

|  |  |
| --- | --- |
| Name | Update Account |
| Identifier | UC2.2 |
| Description | Enable users to update an account |
| Actor | Any of the users |
| Per condition | The account to be updated should be known and existed in the database. |
| Post condition | An account is updated |
| Alternate actions(A): | A1. The system searches for the invalidity, let the user know, and request to enter again.A2.The use case resumes at step four of flow of events. |

Table 2.5: Delete Account sub use case description

|  |  |
| --- | --- |
| Name | Delete Account |
| Identifier | UC2.3 |
| Description | Enable election officer to delete an account |
| Actor | Election officer |
| Per condition | The account to be deleted should be known and existed in the database. |
| Post condition | An account is deleted |
| Alternate actions(A): | A1. The system searches for the invalidity and let the election officer know and request to enter again. |

Table 2.6: Login use case description

|  |  |
| --- | --- |
| Name | Login |
| Identifier | UC3 |
| Description | Validates the user to enter to the system. |
| Actor | Election officer, voter and candidate |
| Per condition | Actors should have to come with username and password. |
| Post condition | The users logs into the system |
| Alternate actions(A): | A2. The system info user to reenter username and/or password. |

Table 2.7: Candidate Registration use case description

|  |  |
| --- | --- |
| Name | Candidate Registration |
| Identifier | UC4 |
| Description | Enable election officer to register election Candidates |
| Actor | Election Officer |
| Per condition | The list of candidates which are ready to register by the election officer should pass the qualification. |
| Post condition | The candidates will be registered and the vote\_Id which enables her/him to vote will be given to the registered candidates. |
| Alternate actions(A): | A1. If the entered information are incorrect, then the system request the election officer to enter again.  A2. The use case resumes at step four of flow of events. |

Table 2.8: Voter Registration use case description

|  |  |
| --- | --- |
| Name | Voter Registration |
| Identifier | UC5 |
| Description | Enable voters to register themselves. |
| Actor | Voter |
| Per condition | The voters which are ready to register should have an account (username and password) and login to the system and make sure that they are not registered yet. |
| Post condition | The voter will be registered and the vote\_Id which enables her/him to vote will be given to the registered voter. |
| Alternate actions(A): | A1. If the entered information is incorrect, then the system requests the voter to enter again.  A2. The use case resumes to step four of flow of events. |

### Table 2.9: Voter Search sub use case description

|  |  |
| --- | --- |
| **Name** | Voter Search |
| Identifier | UC6.1 |
| Description | Enables all users to search a particular voter. |
| Actor | Election Officer, Candidate and Voter |
| Per condition | The user should have an account and logged into the system through it. |
| Post condition | The user possibly searched a wanted voter. |
| Alternate actions(A): | A1. If the entered Voter ID is incorrect, then the system requests the user to enter again. A2. The use case resumes at step four of flow of events. |

### Table 2.10: Candidate Search sub use case description

|  |  |
| --- | --- |
| **Name** | Candidate Search |
| Identifier | UC6.2 |
| Description | Enables all users to search a particular candidate. |
| Actor | Election Officer, Candidate and Voter |
| Per condition | The user should have an account and logged into the system through it. |
| Post condition | The user possibly searched a wanted candidate. |
| Alternate actions(A): | A1. If the entered Candidate ID is incorrect, then the system requests the user to enter again.  A2. The use case resumes at step four of flow of events. A3. Use case ends. |

### Table 2.11: Report use case description

|  |  |
| --- | --- |
| Name | Report |
| Identifier | UC7 |
| Identifier | Enables all users to generate the aggregate candidate’s final result. |
| Actor | Election Officer, Candidate and Voter |
| Per condition | The users should have to login to the system through a legal account. |
| Post condition | The user can possibly review the counted candidates progress or final result report |
| Alternate actions(A): | A1. The system will display a message for the corresponding error and requests the user to enter again. A2. The use case resumes to step four of the flow of events. |

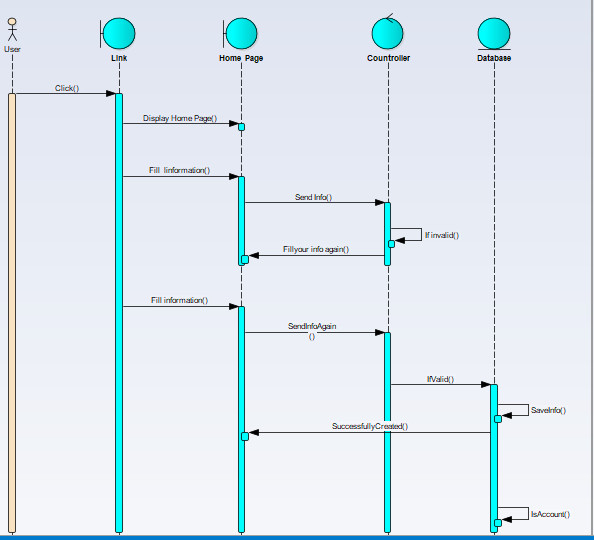
### :

### **2.3.3 Sequence diagram**

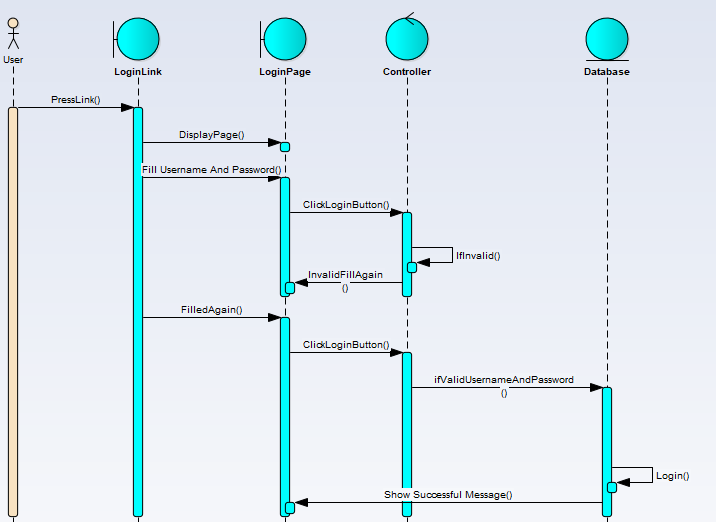
Sequence diagrams are used to model the logic of usage scenario . Usage scenario is exactly what its name indicates-the description of potential ways our system is used. Sequence diagrams are a great way to validate and flesh out the logic of use case scenarios and to document the design of the system . The project team has modeled sequential diagram for the major use cases identified in use case modeling part.

From Figure 2.1 to Figure 2.3 shows sequential diagram the system.

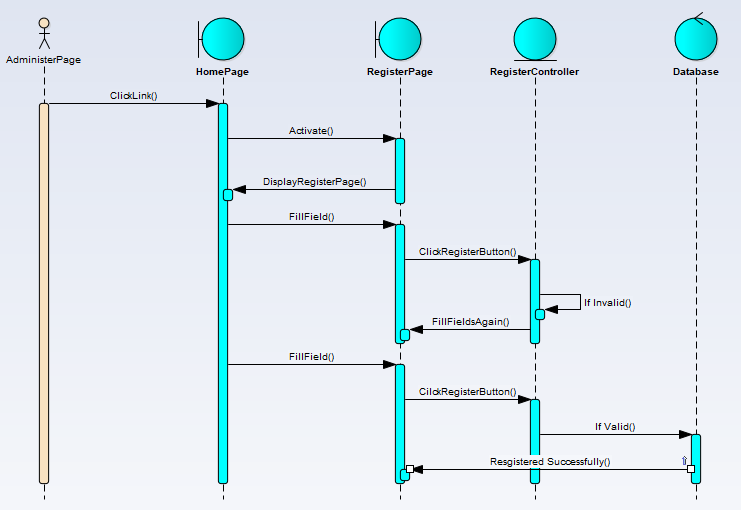
Figer 2.1 Sequence diagram for Create Account



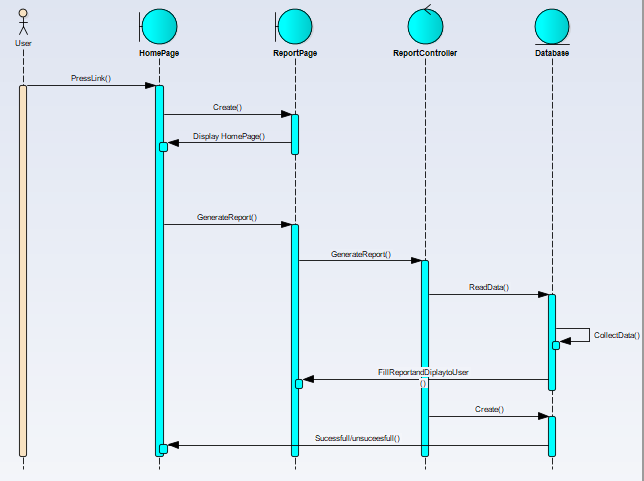
Figer 2.2 Sequence Diagram For Login System



Figer 2.3 Sequence Diagram For Registration

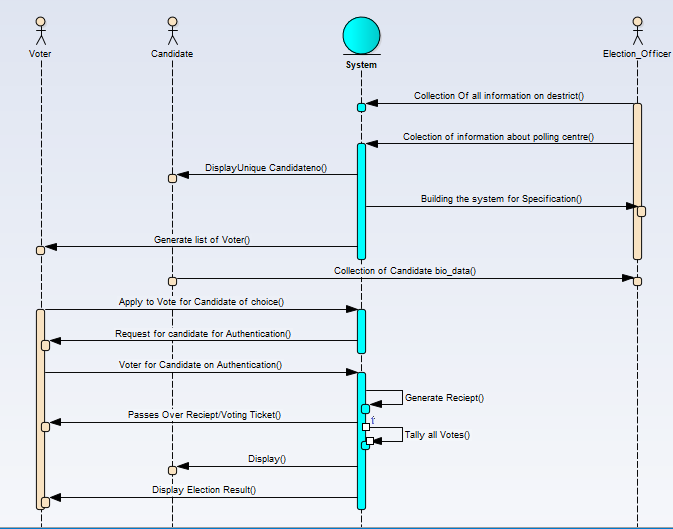


Figer 2.3 Sequence Diagram for Generate Report



### 

Figer 2.4 Sequence Diagram for Casting Vote

****

**Activity Diagram**

Activigty diagram – is basically a flowchart to represent the flow from one activity to an other activity.The activity can be described as operation of the system .

**Activity diagram for casting vote**

****

**Activity diagram for login to Account**



### **2.3.4 Analysis level class diagram (Conceptual Modeling)**

Another technique that is applied to support the conceptual analysis effort is the use of Class diagrams. The UML Class Diagram is used to support both the analysis and the design phase. Classes are an important aspect of object-oriented software . When we recognize a pattern in which there a common group of things that have similar attributes and behaviors, we typically represent that as a class.

We are developing class diagrams by start at the “conceptual” level. The team start by laying out the fundamental elements we think will be needed to support the objectives as identified in the Use Case diagrams.

Implementation issues are dealt with during the design phase as we refine our class diagrams during our development process. Developing at the conceptual level means to keep a perspective that is language and platform neutral, and focus on the essential pieces needed to make our software system work.

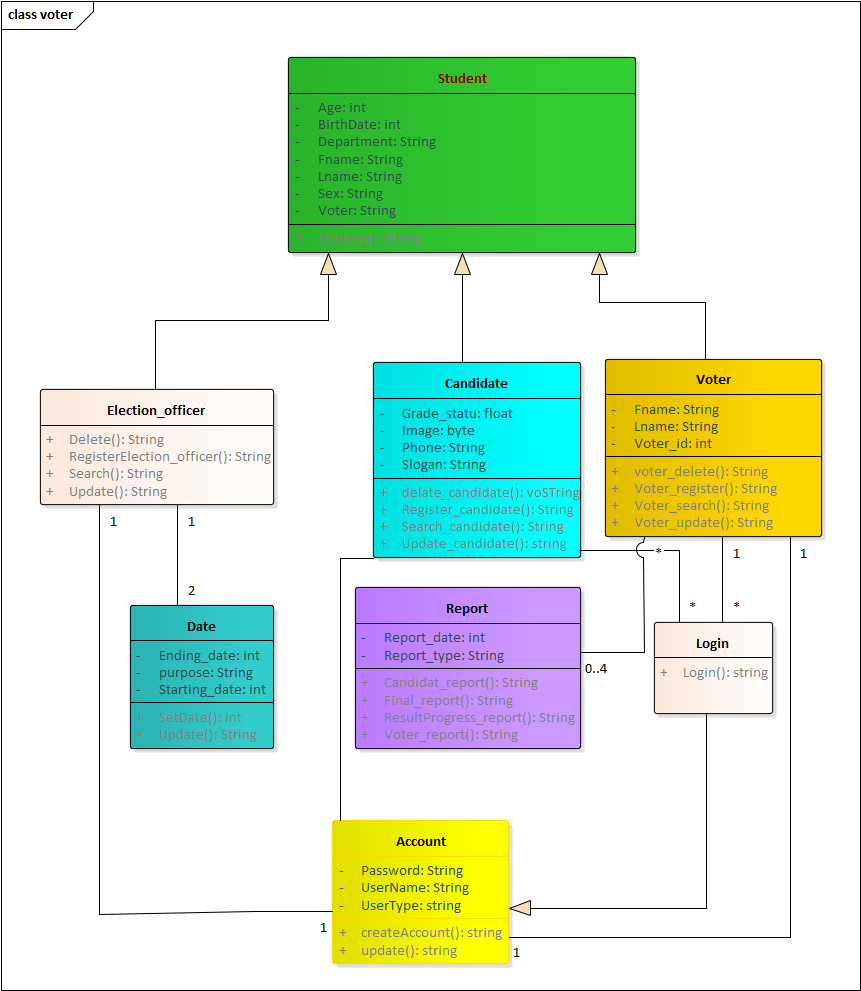


Figure 2.9. Conceptual Level Classes diagram

Since this is the conceptual level, the classes are kept abstract. The goal is to identify the types of objects we expect for voting system. For some of these classes, we have also identified a few attributes we anticipate, and the potential operations for voting Processing.

Description of the class shown in figure 2.15

Account: - helps for creating a user account

Date: - helps during setting registration and voting time interval

Candidate: - helps during registering candidates to be elected

Voter: - helps during registering voters

Report: - helps during generating report of candidates, voters and results

**Chapter three**

**3.1 Implementation of the system**

In this phase the overall procedures, activities and methods of execution during the implementation phase of the project are included. This step takes much time when compared with other steps of the project competence. The source code or script of the project is included in the next part of the project.

**package Package1;**

**/\*\***

**\* @author HP\_pc**

**\* @version 1.0**

**\* @created 28-May-2019 6:13:25 AM**

**\*/**

**public class Students {**

**public int bach;**

**private String Department;**

**public String Fname;**

**public String Lname;**

**public String Sex;**

**private String voter;**

**public void finalize() throws Throwable {**

**}public String Students(){**

**return "";}**

**}//end Students**

**package Package1;**

**/\*\***

**\* @author HP\_pc**

**\* @version 1.0**

**\* @created 28-May-2019 6:13:26 AM**

**\*/**

**public class Election\_officerss extends Students {**

**public Election\_officerss(){}**

**public void finalize() throws Throwable {**

**super.finalize();}**

**public String delete(){**

**return "";}**

**public String RegisterElection\_officer(){**

**return "";**

**}**

**public String Search(){**

**return "";}**

**public String student(){**

**return "";}**

**public String Update(){**

**return "";**

**}**

**}//end Election\_officerss**

**package Package1;**

**/\*\***

**\* @author HP\_pc**

**\* @version 1.0**

**\* @created 28-May-2019 6:13:26 AM**

**\*/**

**public class candidate extends Students {**

**private String Grade\_statu;**

**private String image;**

**private String Phone;**

**private String Slogan;**

**public Account m\_Account;**

**public candidate(){}**

**public void finalize() throws Throwable {**

**super.finalize();**

**}**

**public String delete\_candidate(){**

**return "";}**

**public String regster\_candidate(){**

**return "";}**

**public String search\_candidate(){**

**return "";}**

**public String student(){**

**return "";}**

**public String update\_candidate(){**

**return "";}}//end candidate**

**package Package1;/\*\***

**\* @author HP\_pc**

**\* @version 1.0**

**\* @created 28-May-2019 6:13:26 AM**

**\*/**

**public class voter extends Students {**

**private String Fname;**

**private String Lname;**

**private int voter;**

**public Account m\_Account;**

**public voter(){}**

**public void finalize() throws Throwable {**

**super.finalize();**

**}**

**public String student(){**

**return "";}**

**public String voter\_delete(){**

**return "";}**

**public String voter\_regster(){**

**return "";}**

**public string voter\_search(){**

**return "";}**

**public String voter\_update(){**

**return "";}**

**}//end voter**

**package Package1;/\*\***

**\* @author HP\_pc**

**\* @version 1.0**

**\* @created 28-May-2019 6:13:26 AM**

**\*/**

**public class Account {**

**private int Password;**

**private String userName;**

**private String Usertype;**

**public Account(){**

**}**

**public void finalize() throws Throwable {}**

**public String creatAccount(){**

**return "";}**

**public String update(){**

**return "";}**

**}//end Account**

**package Package1;**

**/\*\***

**\* @author HP\_pc**

**\* @version 1.0**

**\* @created 28-May-2019 6:13:26 AM**

**\*/**

**public class Date {**

**private int ending\_date;**

**private String Purpose;**

**private int Starting\_date;**

**public Election\_officer m\_Election\_officer;**

**public Date(){}**

**public void finalize() throws Throwable {}**

**public String SetDate(){**

**return "";}**

**public String update(){**

**return "";}**

**}//end Date**

**package Package1;/\*\***

**\* @author HP\_pc**

**\* @version 1.0**

**\* @created 28-May-2019 6:13:26 AM \*/**

**public class login extends Account {**

**public voter m\_voter;**

**public candidate m\_candidate;**

**public login(){}**

**public void finalize() throws Throwable {**

**super.finalize();**

**}**

**public String Login(){**

**return "";}}//end login**

**package Package1;/\*\***

**\* @author HP\_pc**

**\* @version 1.0**

**\* @created 28-May-2019 6:13:26 AM**

**\*/public class Report {**

**private String Report\_date;**

**private int Report\_type;**

**public voter m\_voter;**

**public Report(){**

**}public void finalize() throws Throwable {}**

**public String Candidate\_report(){**

**return "";}**

**public String final\_report(){**

**return "";**

**}public String Resultprogress\_report(){return "";}**

**public String Voter\_report(){**

**return "";}}//end Report**

**3.2 Testing**

Testing the final case of implementation. Testing is a proccess to show the correctness of the program. It is checking of the work ability in attempt to descover error and avoiding of such from the system.

**Some example of testing is: -**

**Unit** **Testing**: each model is tested individuallyto discover any errors in its code.

**System** **Testing**: \_ this is the next level in the testing and tests the system as the whole.

**Chapter 4**

**4.1 conclusion**

This online voting system will manage the voters information by which voters can login and his voting rights. The system will incorporate almost all feature of voting system. It provides the tools for maintaining voters vote to every party and it count total number of votes of every party. This system is a web based application to serve the voters as well as the working group of the system in different direction. It decreases the cost and time of voting process. It is very easy to use and it is varying less time consuming. It is very easy to debug.