ONLINE VOTING SYSTEM

Software Project Lab-2

Supervised by

NADIA NAHAR

LECTURE

IIT, University of Dhaka

Md Alamgir Kabir

BSSE0907

Md Samsarabbi

BSSE0916

ACKNOWLEDGEMENT

We are highly indebted for getting such a wonderful opportunity to prepare the Software Requirements Specifications report of the project ‘ONLINE VOTING SYSTEM. We would like to thank whole-heartedly our supervisor, Nadia Nahar, Lecture, Institute of Information Technology, University of Dhaka.

ABSTRACT

The online voting system will be a website which can do the all the tasks related to a fruitful election. This web application contains the services for both of the voters and candidates. It also helps to determine the preference of the candidates by providing all the informations on a particular election.

CHAPTER 1

INTRODUCTION OF ONLINE VOTING SYSTEM

This chapter is a part of our software requirement specification for the project “Online Voting System”. In this chapter we will focus on the intended audience for this project.

1.1 PURPOSE

This document briefly describes the Software Requirement Analysis of Online Voting System. It contains the functional, non-functional and the supporting requirements and establishes a requirements baseline for the development of the system. The requirements contained in the SRS are independent, uniquely numbered and organized by topics. The SRS serves as an official means of communicating user requirements to the developer and provides a common reference point for both the developer team and the stakeholder community. The SRS will evolve over time as users and developers work together to validate, clarify and expand its contents.

1.2 INTENDED AUDIENCE

This SRS report is intended for several audiences including the users as well as the project managers, designers, developers, and testers.

1. The user will use this SRS to verify that the developer team has created a website that is acceptable to the user.

1. The project managers of the developer team will use this SRS to plan milestones and a delivery date and ensure that the developing team is on track during development of the system.

1. The designers will use this SRS as a basis for creating the system's design. The designers will continually refer back to this SRS to ensure that the system they are designing will fulfill the user's needs.

1. The developers will use this SRS as a basis for developing the system's functionality. The developers will link the requirements defined in this SRS to the website. They ensure that they have developed a website that will fulfill all of the user's documented requirements.
2. The testers will use this SRS to derive test plans and test cases for each documented requirement. When portions of the website are complete, the testers will run their tests on that part to ensure that the website fulfills the requirements documented in this SRS. The testers will again run their tests on the entire system when it is complete and ensure that all requirements documented in this SRS have been fulfilled.

1.3 CONCLUSION

This analysis of the audience helped us to focus on the users who will be using our analysis. This overall document will help each and every person related to this project to make the project effective and efficient.

CHAPTER 2

INCEPTION OF ONLINE VOTING SYSTEM

2.1 INTRODUCTION

The online voting system will be a website which can do the all the tasks related to a fruitful election. This web application contains the services for both of the voters and candidates. It also helps to determine the preference of the candidates by providing all the informations on a particular election.

2.2 INCEPTION OF ONLINE VOTING SYSTEM

At the beginning of our project, we entered the inception stage. This stage includes, how the project will be started and their scope and limitation. The main goal of this phase is to identify the requirements, demand and establish some sort of mutual understanding between the developers and the stakeholders. In order to make this phase effective we took the following steps:

1. Identifying the client of our project
2. Icebreaking
3. Identifying the stakeholders
4. Identifying the multiple viewpoints of stakeholder

2.3 IDENTIFY THE CLIENT OF OUR PROJECT

We all know that election is a common process to select the right person to a particular responsibility. That's why it's easy to understand that almost all the organizations and different societies will need an election automation system that will help them to perform the election smartly. We will try to make our website according to their demand. So, now our clients are different organizations and clubs.

2.4 ICEBREAKING

Icebreaking refers to the fact that to diminish the communication barrier between you and the other person. It is a crucial part since it denotes the acceptation of our proposal. We started this face by talking with them with context free languages. Their behavior, respond to our question or willing to take a change in their organization solely depends on this phase.

2.5 IDENTIFYING THE STAKEHOLDERS OF ONLINE VOTING SYSTEM

Stakeholder refers to any person or group who will be affected directly or indirectly by the system. Stakeholders include end-users who interact with the system and everyone else in an organization who may be affected by its use. The organizations we visited have limited number of stakeholders. Identification of the stakeholders were done from the information provided by the members. Their names are given below:

Voter
Candidate
Election Commissioner
Club
Organization

2.6 IDENTIFYING MULTIPLE VIEWPOINTS OF THE STAKEHOLDER

Different stakeholders expect different benefits from the system as every person has his own point of view. So, we have to recognize the requirements from multiple viewpoints. Different viewpoints of the stakeholders about the expected website are given below:

VOTER'S VIEWPOINT:

1. Easy use of online voting system
2. It will have an attractive user interface
3. See details on different Candidates
4. None can see one's vote
5. No location dependency
6. Analyzing preferences of candidates
7. Store the information of an election
8. Result will be authentic
9. Vote will be safe

CANDIDATE'S VIEWPOINT:

1. Publish their election propaganda
2. Store the information of an election
3. They can upload files and campaign on election
4. Result will be authentic
5. They can send notification to voters
6. Store details of the election

ELECTION COMMISSIONER'S VIEWPOINT:

1. Selecting candidates of the election
2. Publish notification to all the voters
3. Provide nomination among the selected candidates
4. They can control everything of an election

5. Casting and counting vote honestly
6. Publish result numerically and graphically
7. Ease of maintenance

CLUB AND ORGANIZATION'S VIEWPOINT:

1. Automation of election
2. High security and managing capability
3. It will able to serve all the purpose of election
4. Honest and reliable election

2.7 CONCLUSION

Online Voting System is a website providing services for a smart election. By the help of this website, people can vote, see the result, get the notification, determine candidate's preferences. We hope that we can provide all the benefits and fulfill the requirements given by the stakeholders.

CHAPTER 3

ELICITATION OF ONLINE VOTING SYSTEM

After discussing on the Inception phase, we need to focus on the Elicitation phase. So this chapter specifies the Elicitation phase. We have collected all informations on the election process from the senior vice president of DUITS (Dhaka University IT Society). They hold their election only for the President and General Secretary. All other posts are given by the election commissioners who are generally the previous committee members. We have also collected information from the members of the organization.

3.1 INTRODUCTION

Requirements Elicitation is a part of requirements engineering that is the practice of gathering requirements from the voters, candidates and other stakeholders. We have not faced so many difficulties, like understanding the problems. But making questions for the stakeholders, limited communication with the stakeholders due to a short amount of time and volatility make some problems. Though it is not easy to gather requirements within a very short time, we have surpassed these problems in an organized and systematic manner.

3.2 ELICITING REQUIREMENTS

We have followed Question and Answer (Q&A) approach at the time of collecting information. The main task of this phase is to combine the elements of problem solving, elaboration, negotiation and specification. The collaborative working approach of the stakeholders is required to elicit the requirements. We have finished the following tasks for eliciting requirements-

Collaborative Requirements Gathering
Quality Function Deployment
Usage Scenarios

3.3 COLLABORATIVE REQUIREMENTS GATHERING

We have met with many stakeholders in the Inception phase such as the voter, candidates and other users. These meetings created an indecisive state for us to elicit the requirements. To solve this problem, we have met with the stakeholders (who are acting a vital role in the whole process) few times to elicit the requirements.

3.4 QUALITY FUNCTION DEPLOYMENT

Quality Function Deployment (QFD) is a technique that translates the needs of the customer into technical requirements for any application. Ultimately the goal of QFD is to translate subjective quality criteria into objective ones that can be quantified and measured and which can then be used to design and manufacture the product. It is a methodology that concentrates on maximizing customer satisfaction from the software engineering process. So, we have followed this methodology to identify the requirements for the project. The requirements, which are given below, are identified successfully by the QFD.

3.5 NORMAL REQUIREMENTS

Normal requirements are generally the objectives and goals that are stated for a system during meetings with the customer. The presence of these requirements fulfills customer's satisfaction. The following are the normal requirements for our project.

A user friendly interface
Keep information about an election
Store information of voters and candidates
Minimum effort to use the website
Authentic result publishing

3.6 EXPECTED REQUIREMENTS

These requirements are intrinsic to the system and may be so elementary that the customer does not explicitly state them. Their absence will be a cause for significant dissatisfaction. Below the expected requirements for our project are briefly described.

Election automation
Authentic voting system
Interactive and attractive user interface
Storing all information of an election
Time-saving
No location dependency
Securities

3.7 CONFLICTING REQUIREMENTS

Conflicting requirements differ among stakeholders. But we found that there are not many conflicting requirements. Almost all the stakeholders wanted the same thing and provide similar type of information. But some voters told not to limit nomination of specific number of candidates.

3.8 EXCITING REQUIREMENTS

These requirements are for features that go beyond the customer's expectations and prove to be very satisfying when present. Following are some exciting requirements of our project.

Election campaign
Sending Notification
Graphical representation

CHAPTER 4

SCENARIO BASED DIAGRAM

4.1 USE CASE SCENARIO

Online Voting System is a website providing services for a smart election. It is an automated system for the following purposes:

- Authentication
- Profile Maintenance
- Notification
- Publishing Result

4.2 AUTHENTICATION

We are concerned about handling the EC, candidate and the voter basically. Here, Admin set the election commissioner as admin to a particular Election. Election is identified by election id. The users of the system can be categorized into three main categories:

1. Election Commissioners
2. Candidate
3. Voter

4.3 REGISTRATION/SIGN-UP: DATA ENTRY

The first person to be registered into the system is the election commissioners. The information he/she has to provide at the time of registration includes:

- First name
- Last name
- Gender
- Birthdate
- Permanent Address
- NID
- Contact Number
- Current address
- Email
- Password
- Image and Symbol for candidates only
- One backup question and it's corresponding answer

4.4 **REGISTRATION/SIGN UP: VALIDITY CHECKING AND STORING INFORMATION**

At the time of data entry there would be a validity check. The password must contain minimum 8 characters and maximum 20 characters including at least one digit. The format of email, password is verified at the time of input. Correct entries result in account creation. The registration information of the individual is stored in Database.

System collect the information from the users at the time of registration and save them in the database. System may have several databases. Website use these databases for the following purposes:
Creating profile,
Logging in,
Validate vote,
Casting vote,
Counting vote,
Publishing result,

Finding preferences of vote on different factors,
and other tasks as well.

Election commissioners will handle everything as per the election criteria.

4.5 SIGN IN

When the user wants to sing-in, he or she must undergo authentication. He/she enters his/her respective email and password. The entered data is matched with the corresponding data stored in Database. If entered data matches the stored data, the user gains access to the system. The login time of a user is recorded on a daily basis.

4.6 ACCOUNT RECOVERY

If the user fails to recall his/her password, he/she directly choose the account recovery option. Otherwise he/she can try ten times. If the entered password is incorrect on the 10th attempt, the user will be directed to the account recovery option. In case of updating password, confirmation code will be sent to the users email. According to that code he/she can update his/her password and get access to the system.

4.7 SIGN OUT

When a user attempts to sign out, the system checks for unsaved data(active process) whether there remains any active or unsave data. All of their update activities will be saved before logout. System will contain the last logout time.

4.8 PROFILE MAINTENANCE

Users who register in a particular election also can create a account by providing the information. That is, every users has an account according to the information they provided at the time registration. There are three user accounts in the system. The users can see the update of the election process by logging in the system. They can update or change their respective account. But all modification will be safe only if the election commissioners approve it. Candidates are also a voter. For making a candidate profile, they may have to provide some additional informations. Specially, they must have to upload images of their respective pictures for identification and their election symbols.

4.9 PUBLISHING RESULT

Publishing result fulfills the criteria of a particular election. After casting the votes, system will count the votes of the respective candidates. It publishes the result numerically and graphically. System also provides the result according to the attributes. That is, suppose a voter may want to know how many female voter vote a specific candidate. Result shows the details about the vote difference among the candidates, how many people from which ages or from which areas.

Only the voters can vote and no duplicate vote can be occurred. By ensuring this our website will have only valid votes so that result can be authentic. For publishing the result of an election we will follow the steps below:

4.10 CASTING VOTE

When a voter wants to vote, first we check whether the voter is in voter lists. After that we check whether the voter casts his vote. If both condition will true then the voter can see the candidate lists. After that he/she can select his favourite candidates. System will say whether the selected candidates is confirmed. If the voter agrees, then his vote is casted.

4.11 COUNTING VOTE

System will count the votes only after casting a vote. It will increase the total count as well as corresponding candidate's count when a voter casts his vote. Time of vote and all other necessary information will be saved in the database. But which voter vote which candidate is recorded through hash value and in encrypted mode so that none can see.

4.12 GRAPHICAL REPRESENTATION

After having finished the election time result will be shown according to the approval of election commissioners. Users can see the result both numerically and graphically which may feel good to understand the result.

4.13 NOTIFICATION

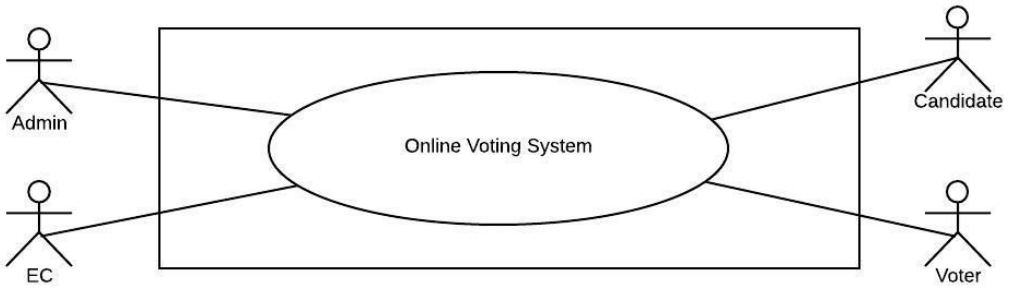
Election Commissioner can send file to inform the rules and regulations about the elections and all the candidates and voters will get the notification of it. Candidate can campaign about election, publish their election propaganda by uploading file in the website. The notifications sent from the candidates will be approved by the election commissioners. All the voters get notification of the file which may help them to find the right candidates. Voters can only see the notification, they can not send notification.

None can see that which voter vote which candidate. Everyone will have the proof of the election result. No duplications of vote can occur. Voter need to be careful at the time of voting because vote confirmation will be held only once.

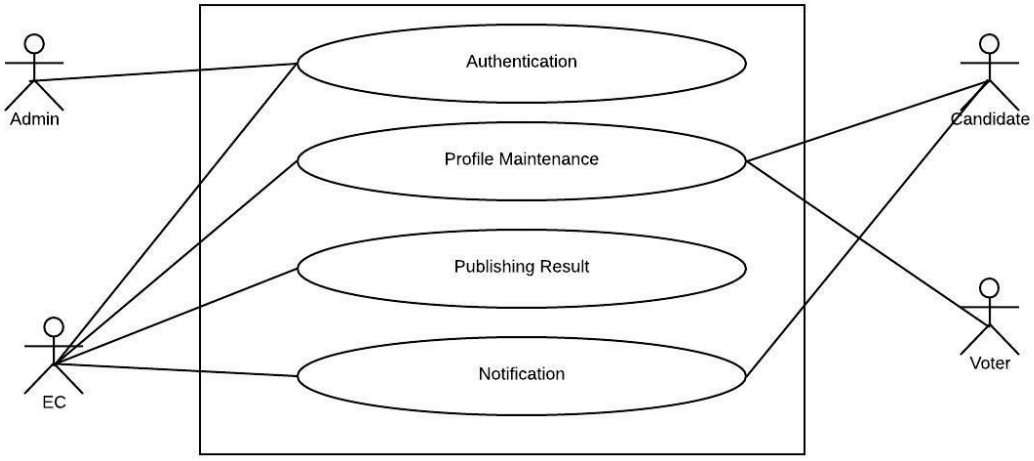
4.14 ELECTION CAMPAIGN

Candidates can declare their election propaganda and try to convince people to vote themselves. That is, they can send file for publicity and all the voter will get the notification of it so that people can judge the candidates. It will help the voters to choose the right candidates they actually wanted. All information about the election will be recorded according to an election id.

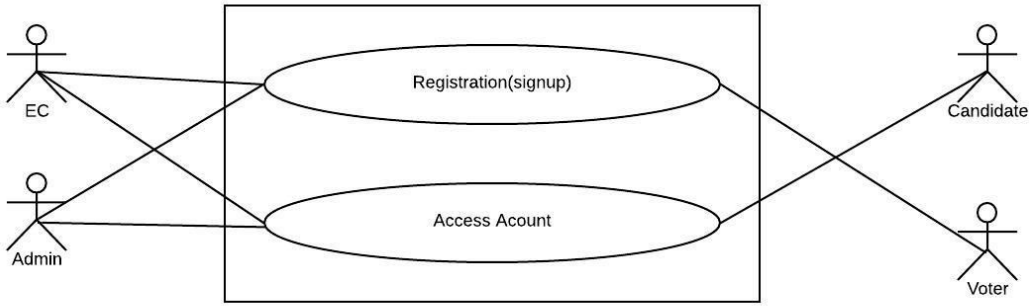
4.15 USE CASE DIAGRAM



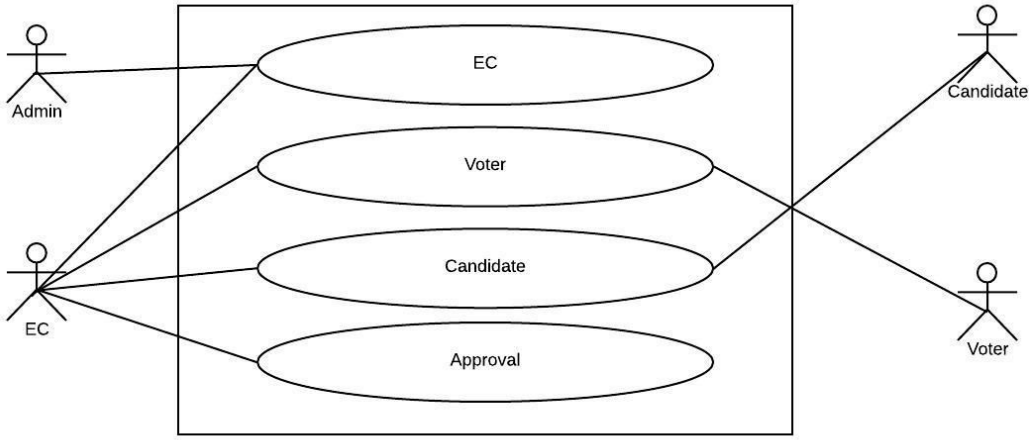
Use Case:Level-0



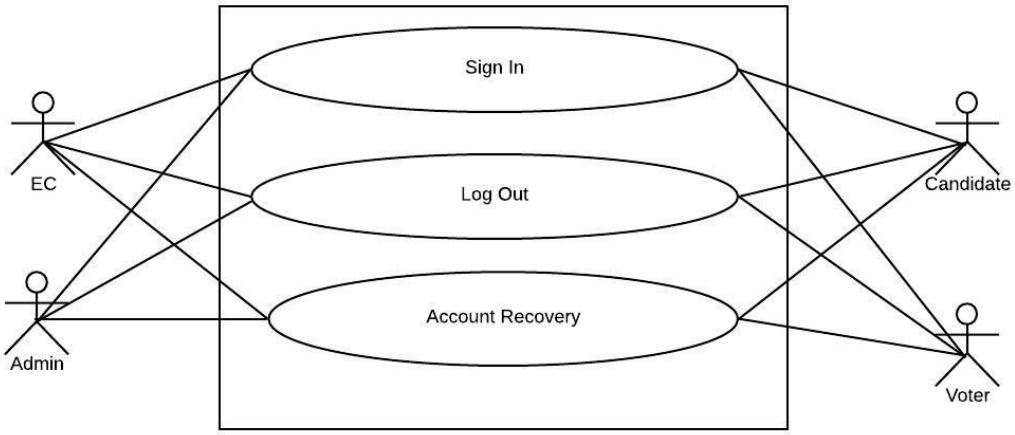
Use Case :Level-1



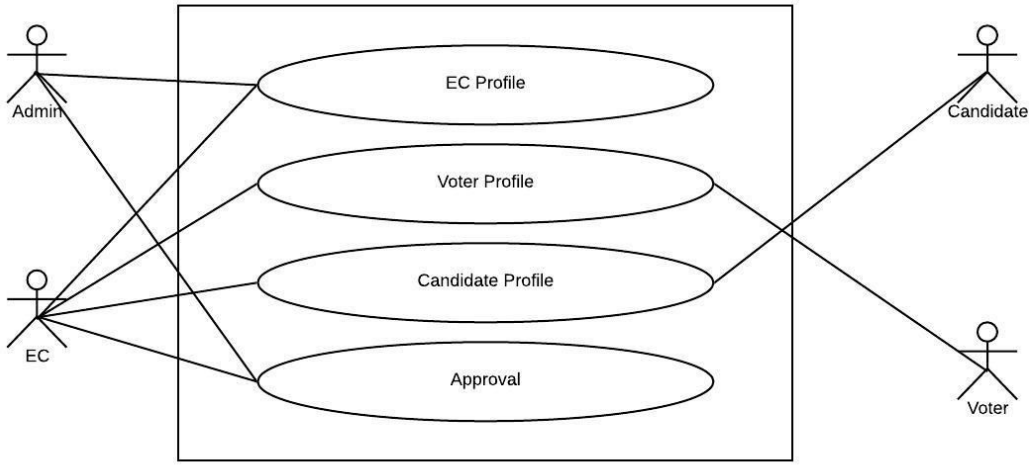
Use Case:Authentication-1.1



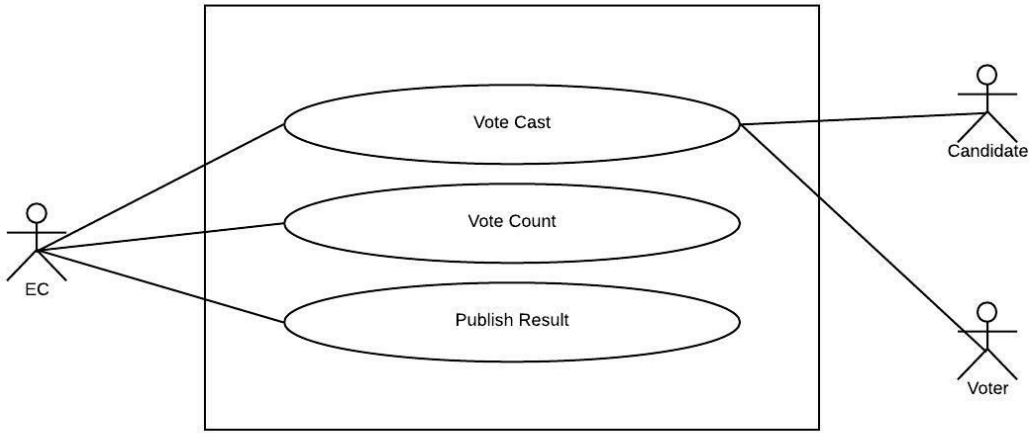
Use Case:Registration(Sign up)-1.1.1



Use Case:Access Account-1.1.2

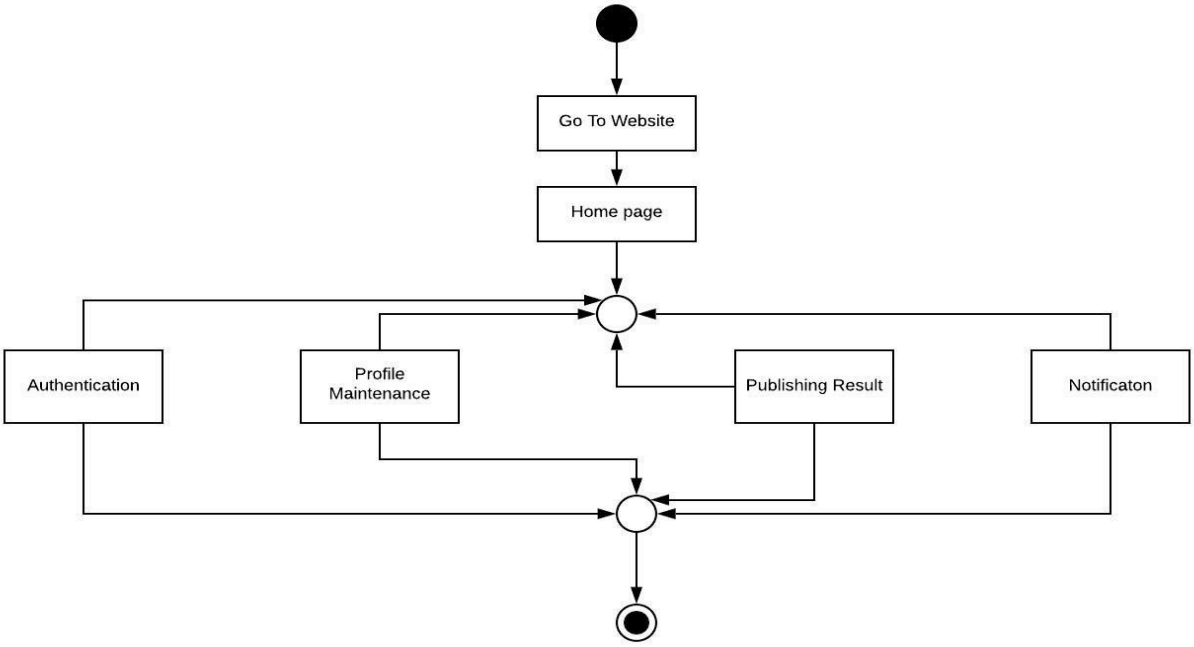


Use Case:Profile Maintenance -1.2

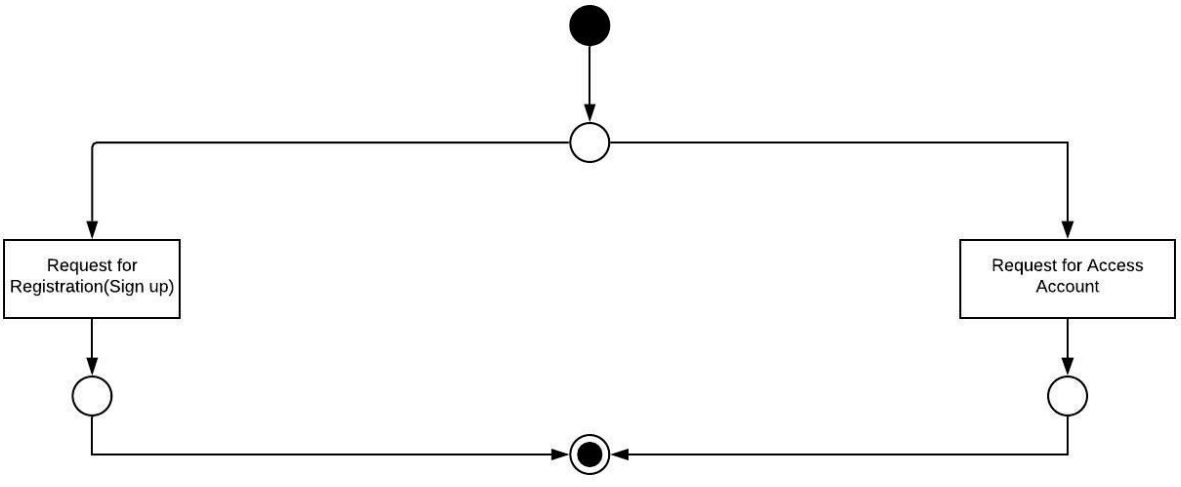


Use Case:Result Publishing-1.3

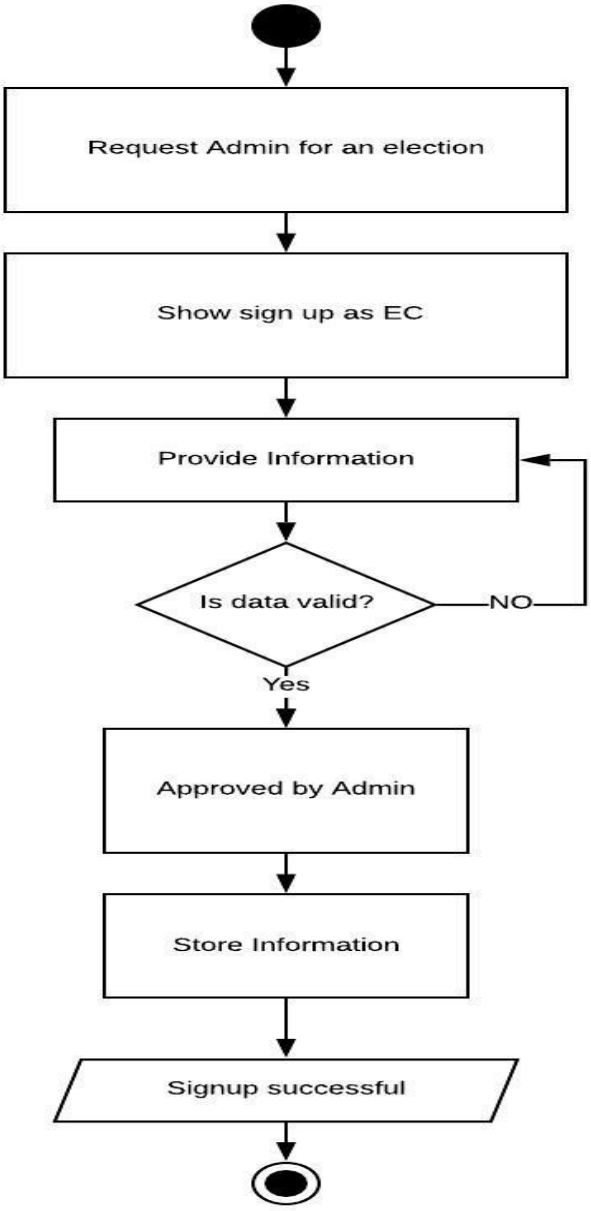
4.16 ACTIVITY DIAGRAM



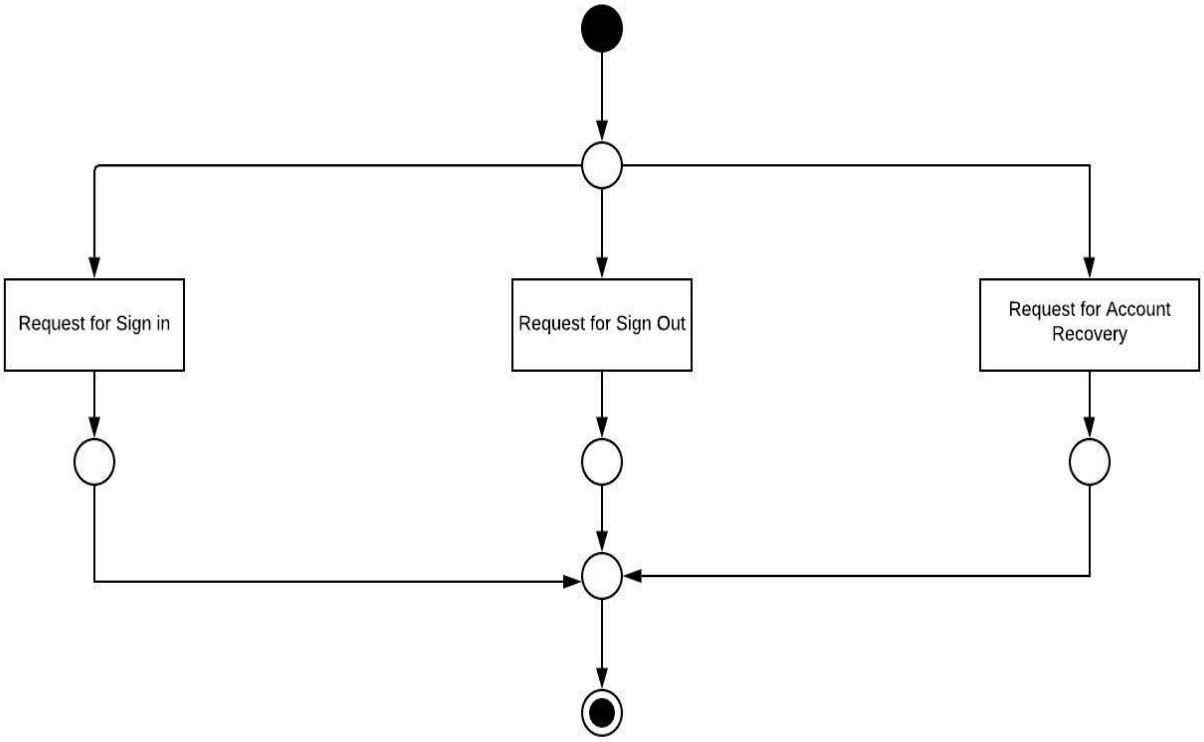
Activity Diagram:Level-1



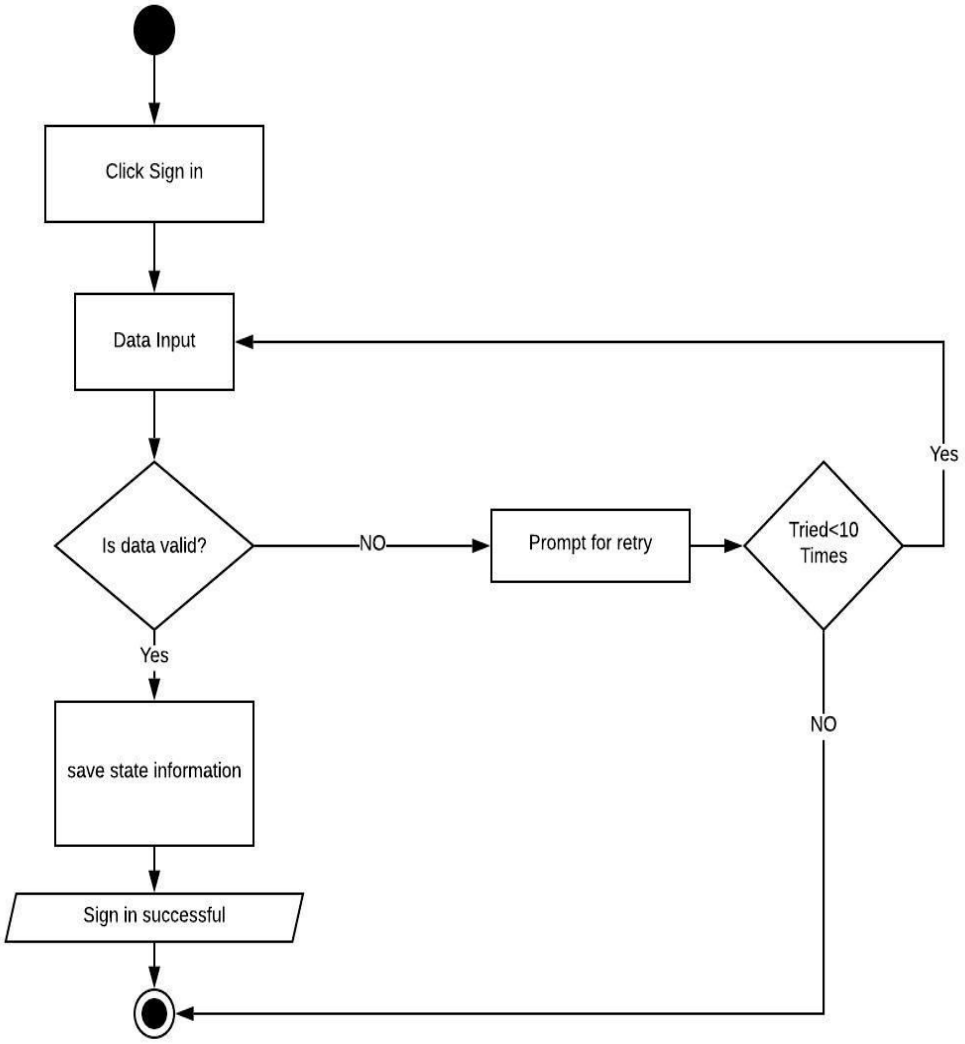
1.1 Activity diagram-Authentication



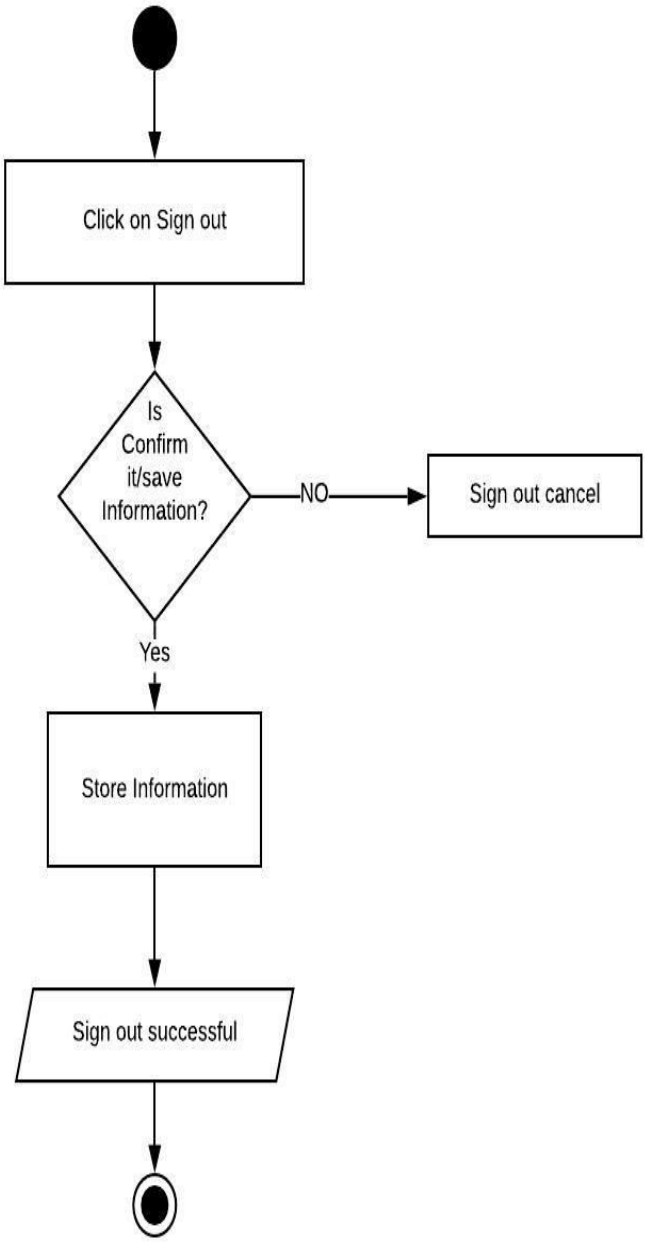
1.1.1 Activity diagram-Registration(Sign up as EC)



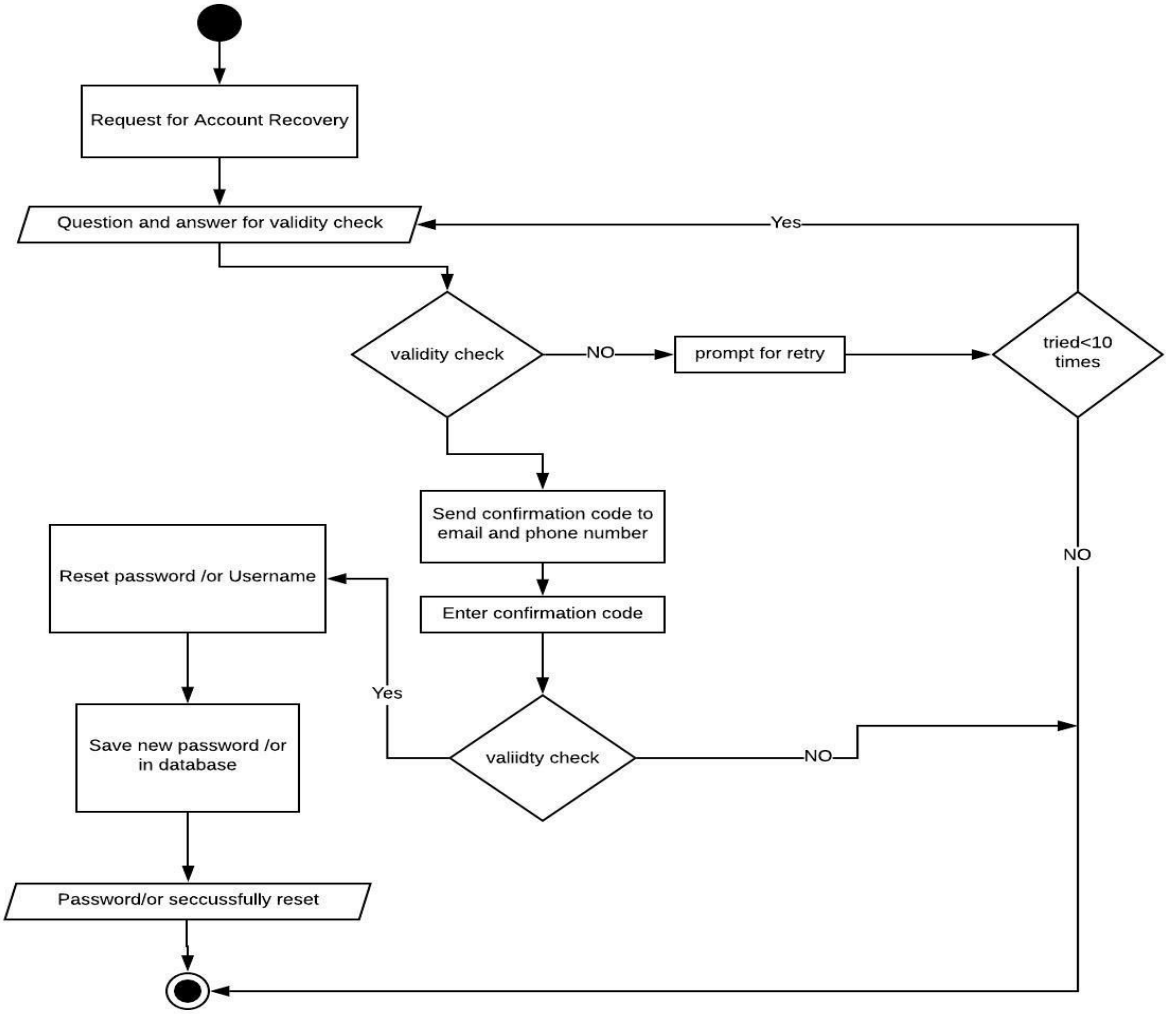
1.1.2 Activity diagram-Access Account



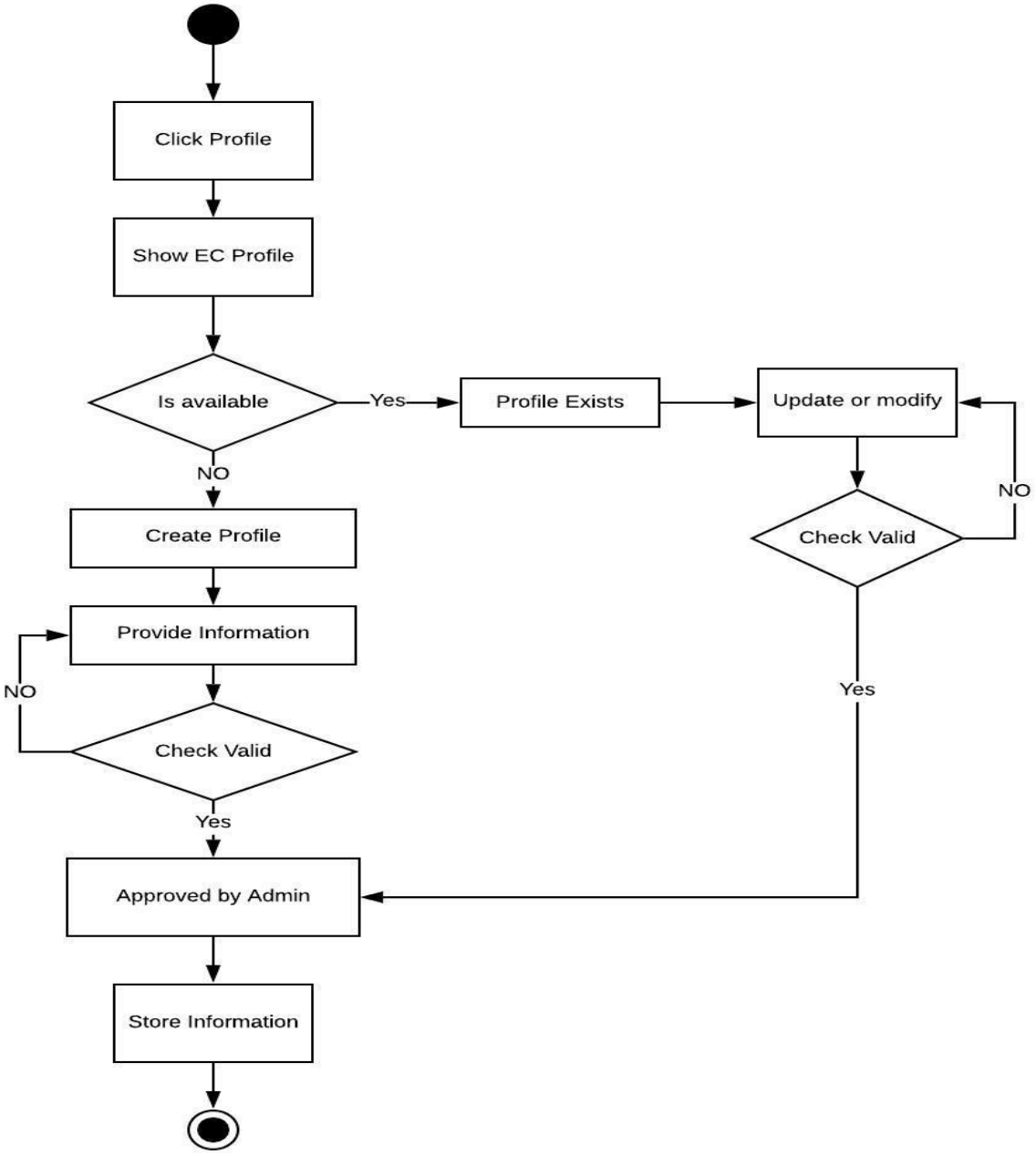
1.1.2.1 Activity diagram-Sign in



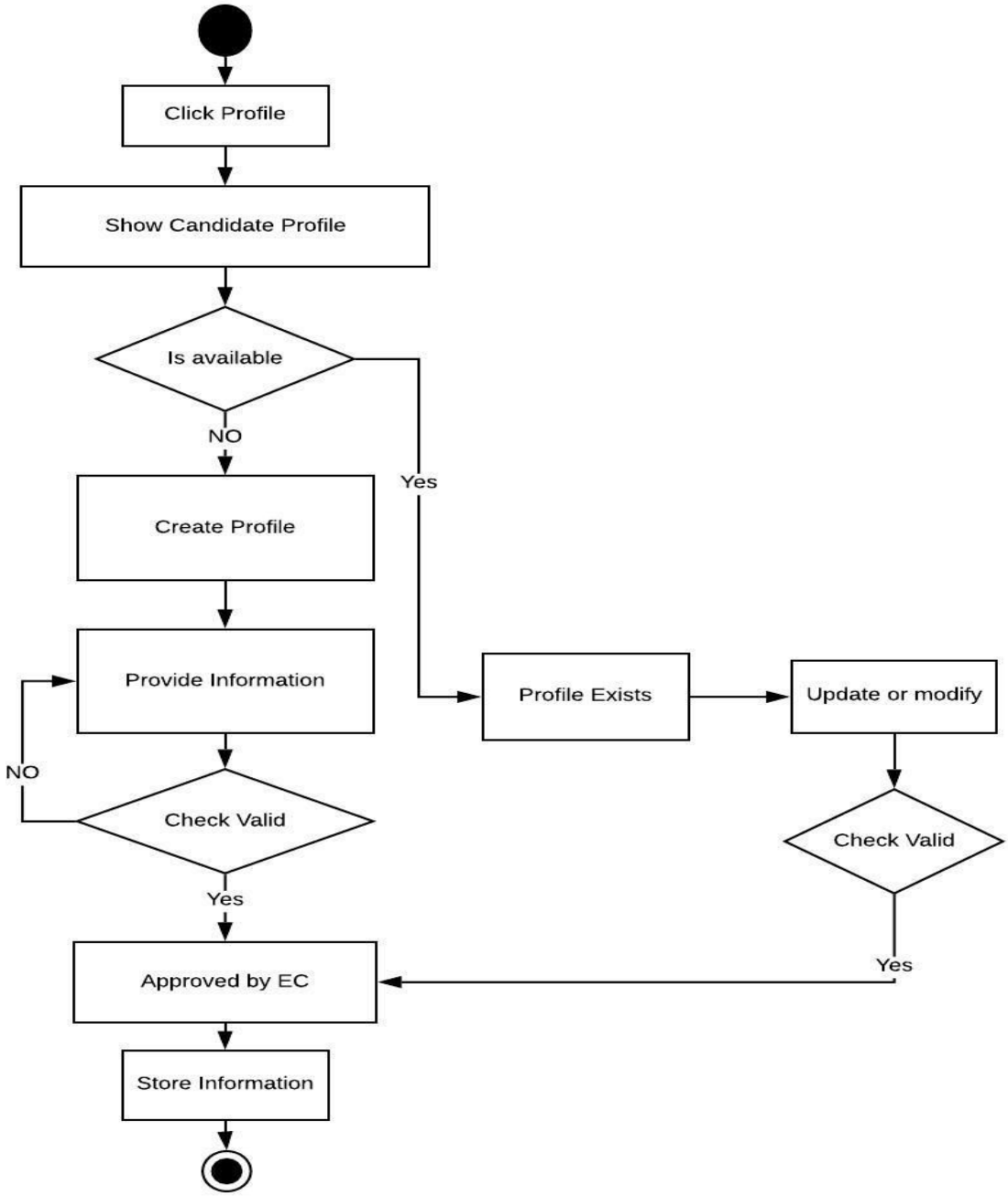
1.1.2.2 Activity diagram-Sign out



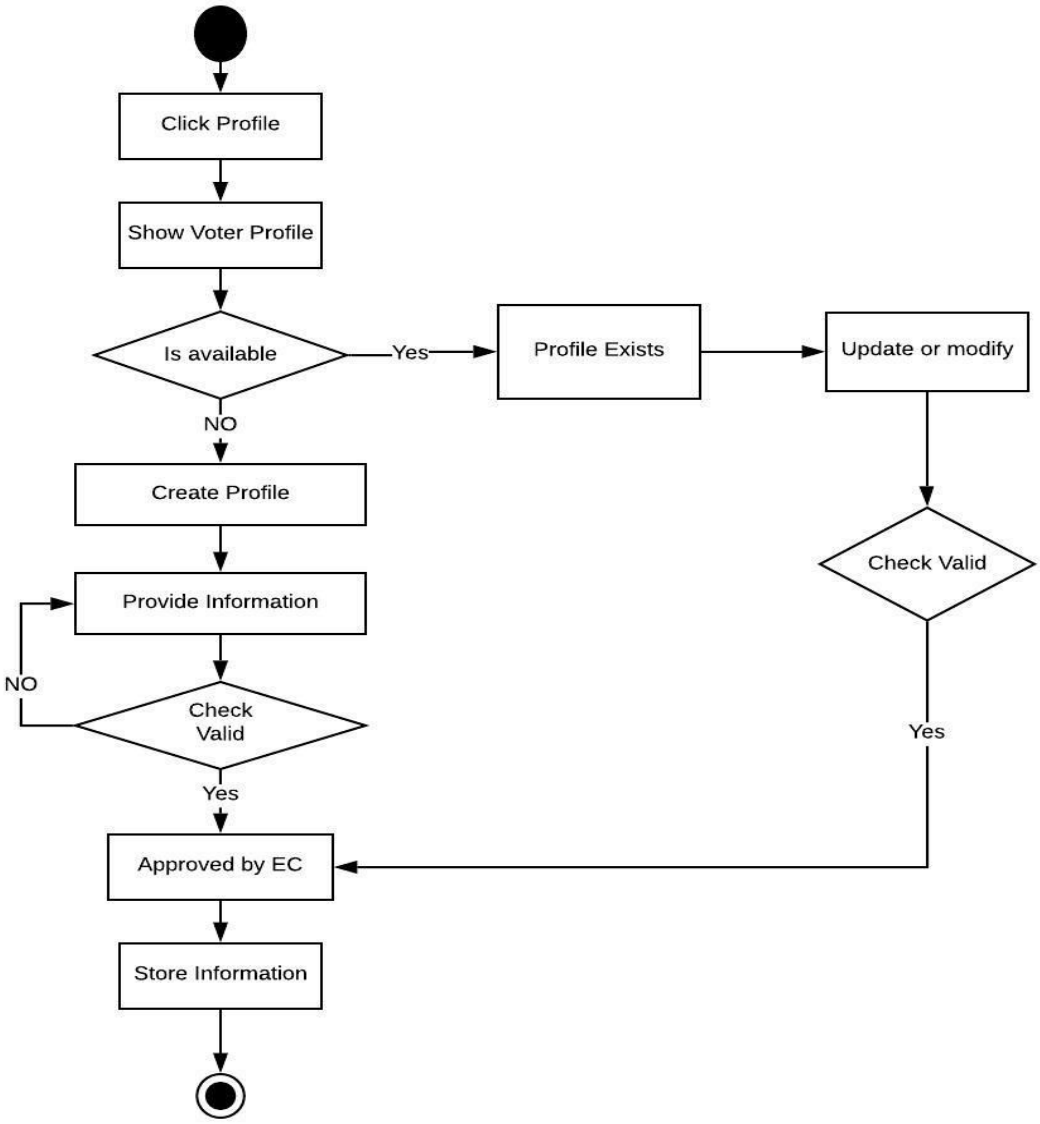
1.1.2.3 Activity diagram-Account Recovery



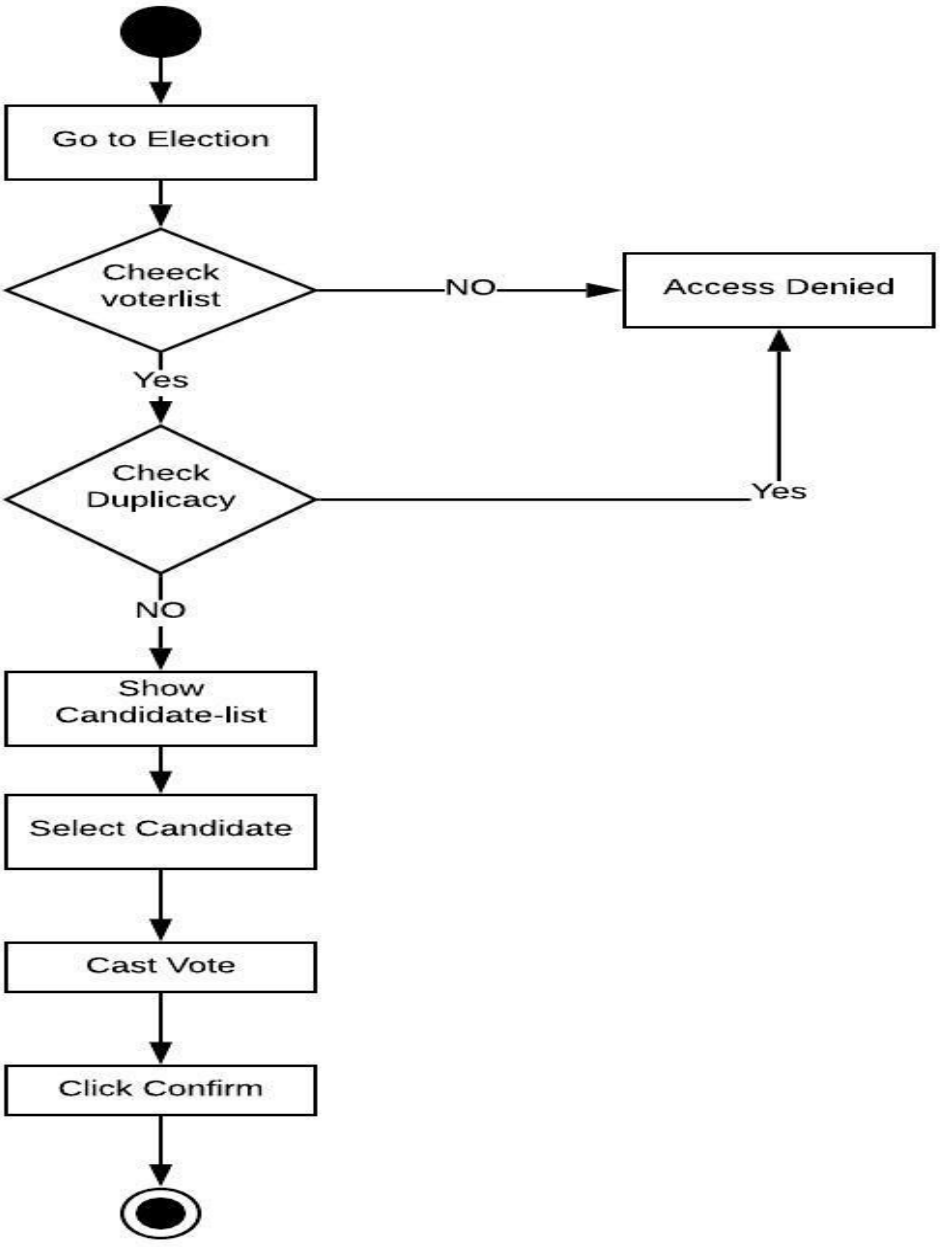
1.2.1 Activity diagram-EC Profile



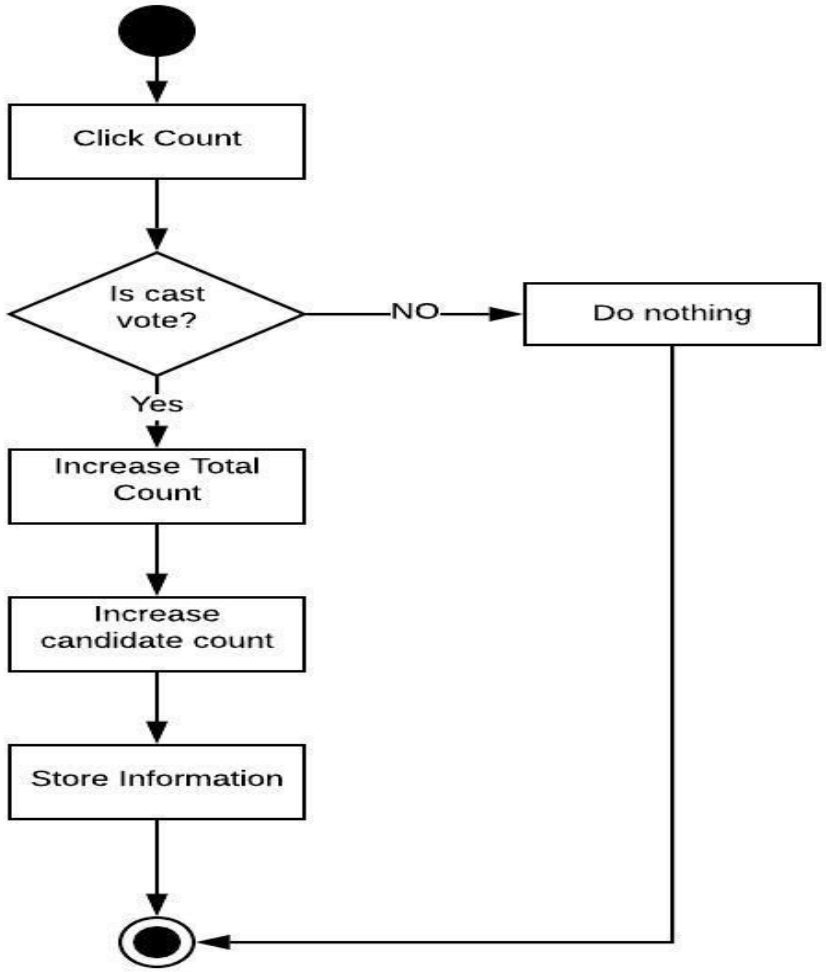
1.2.2 Activity diagram-Candidate Profile



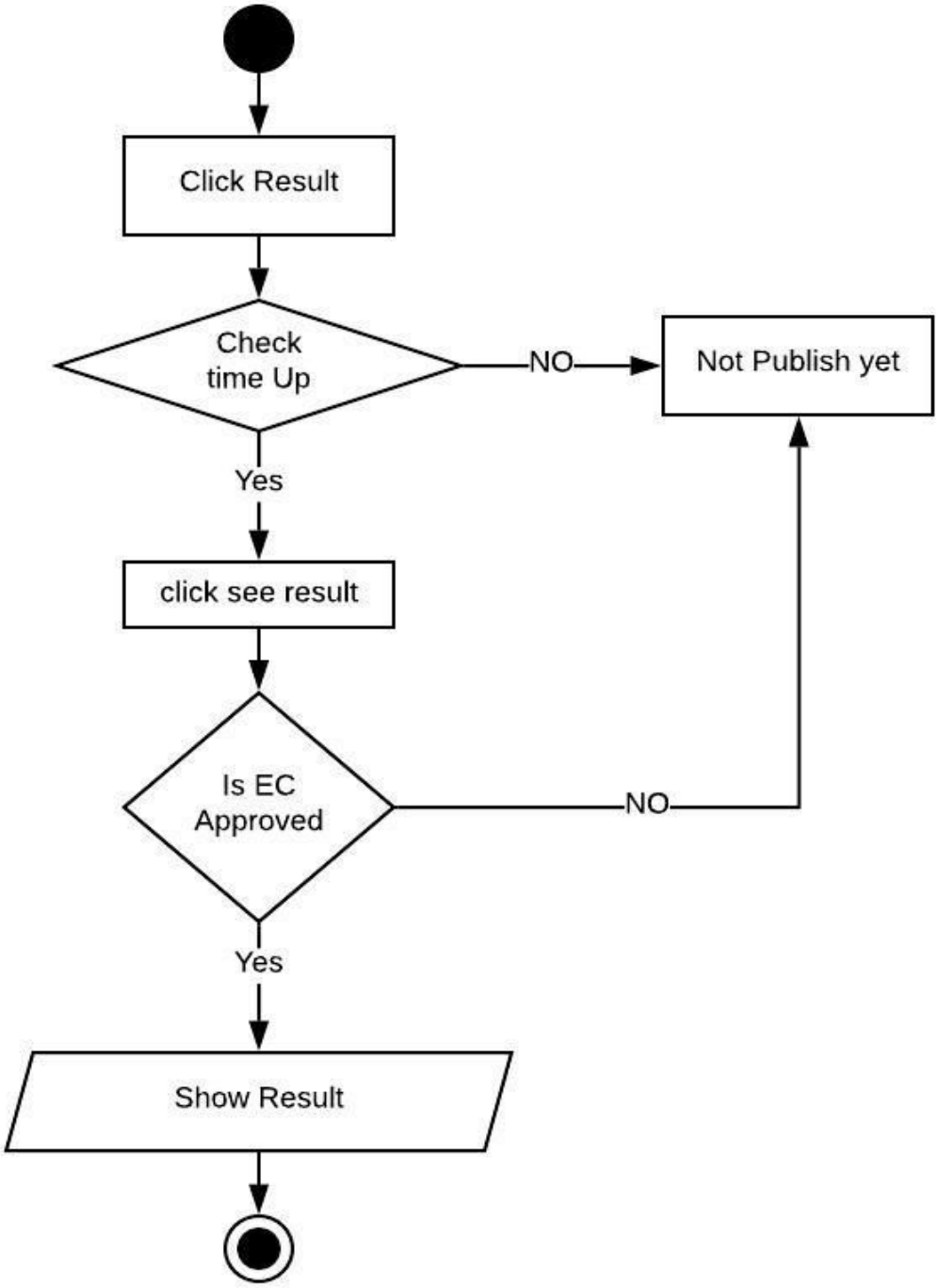
1.2.1 Activity diagram-Voter Profile



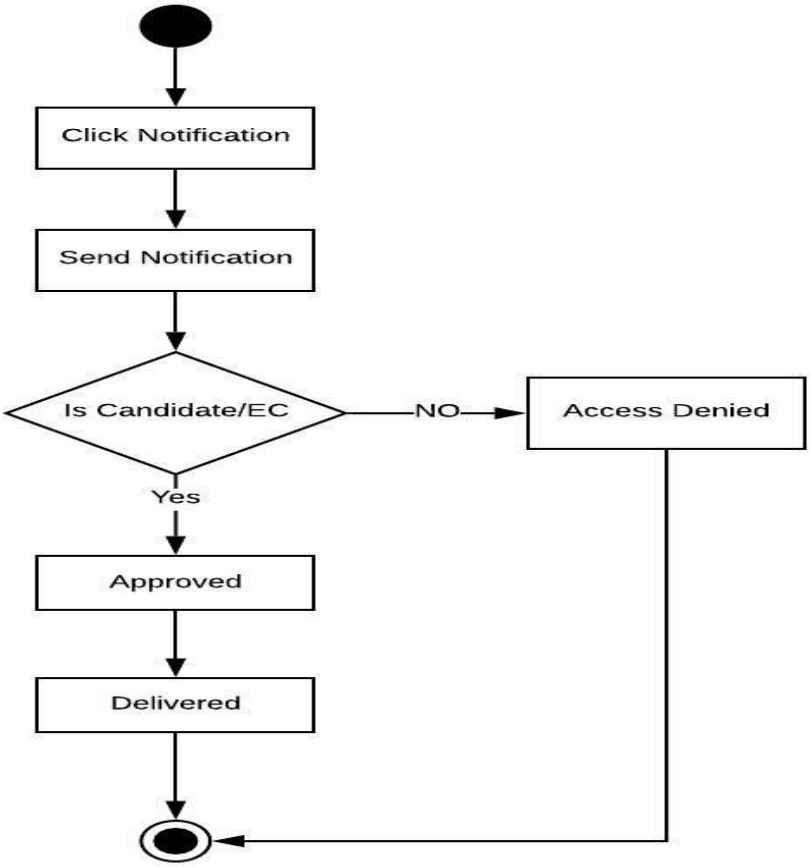
1.3.1 Activity diagram-Vote Casting



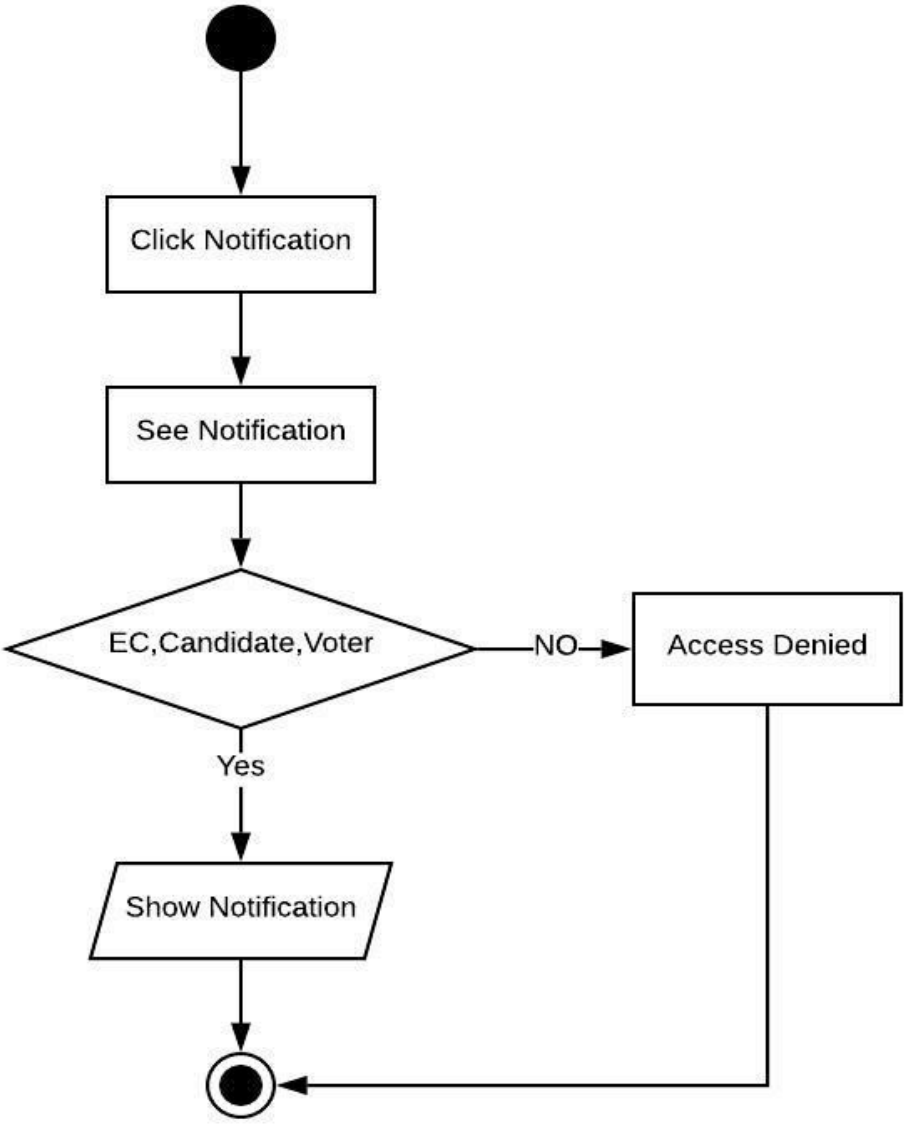
1.3.2 Activity diagram-Vote Count



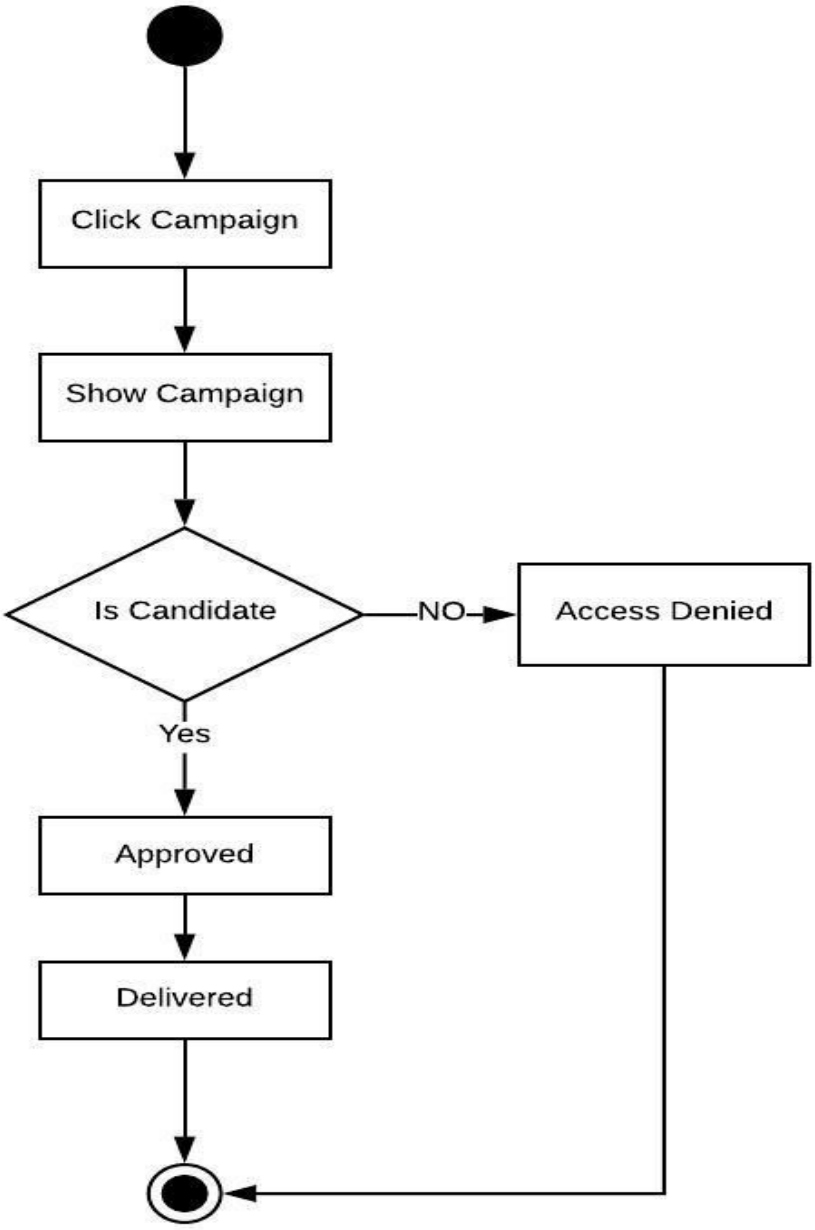
1.3.3 Activity diagram-Publish Result



1.4.1 Activity diagram-Send notification

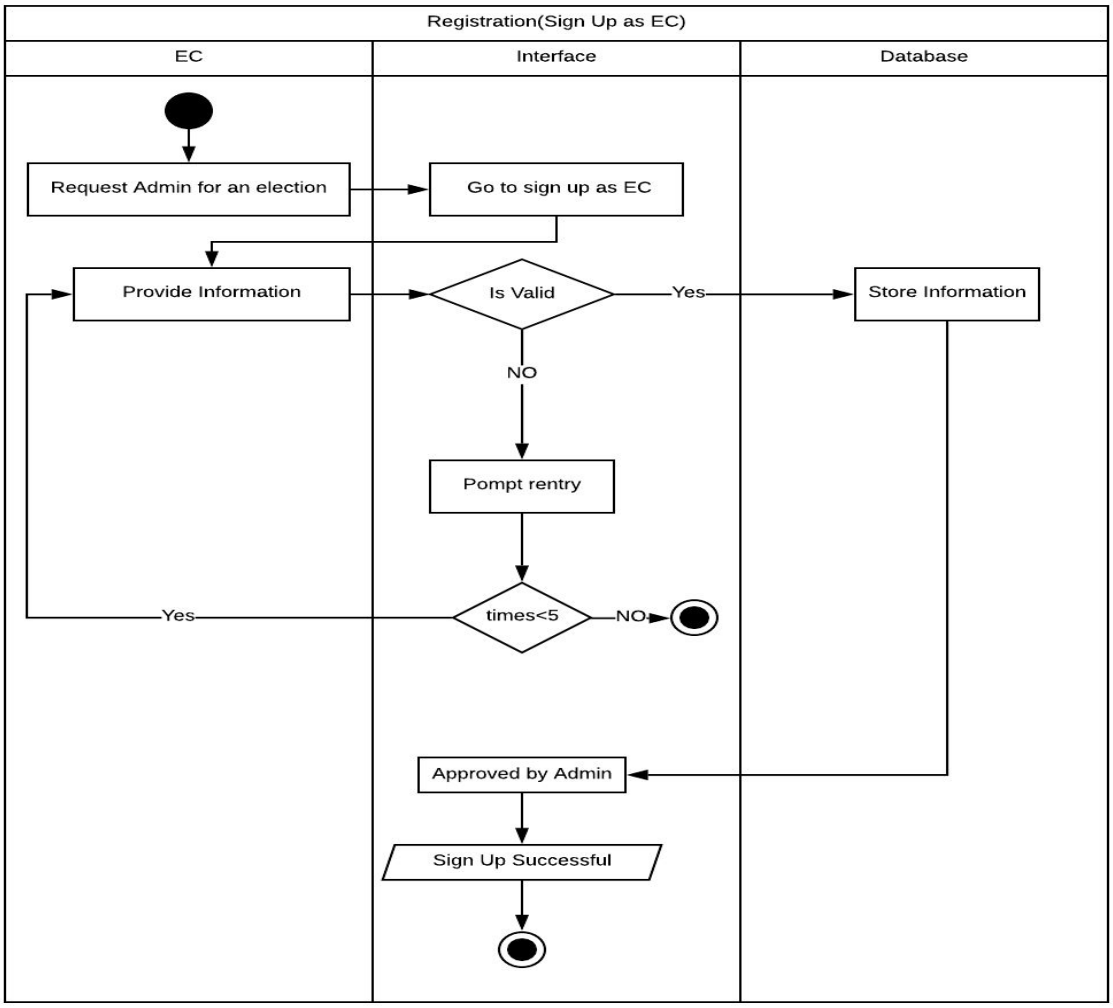


1.4.2 Activity diagram-See notification

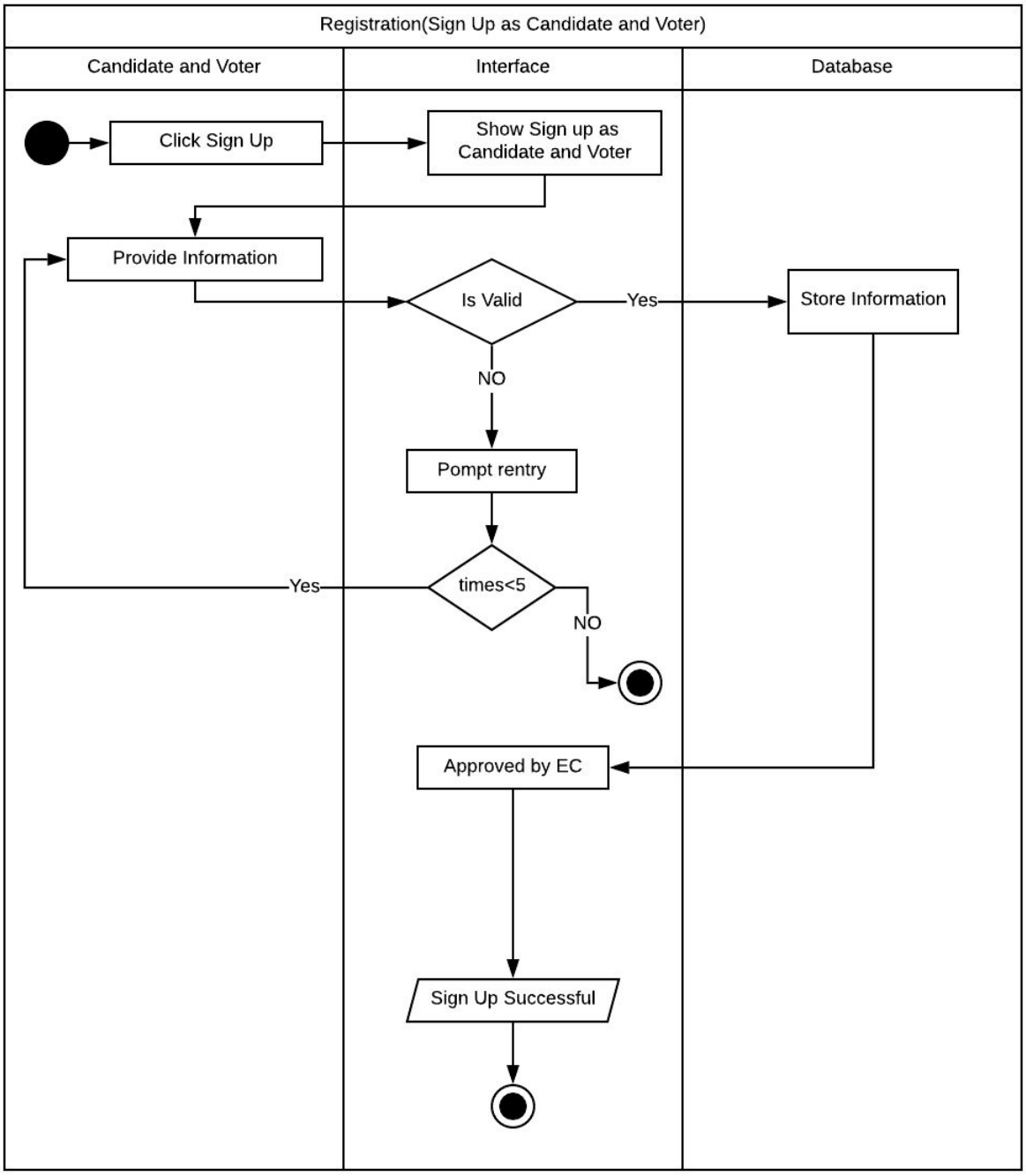


1.4.1 Activity diagram-Campaign

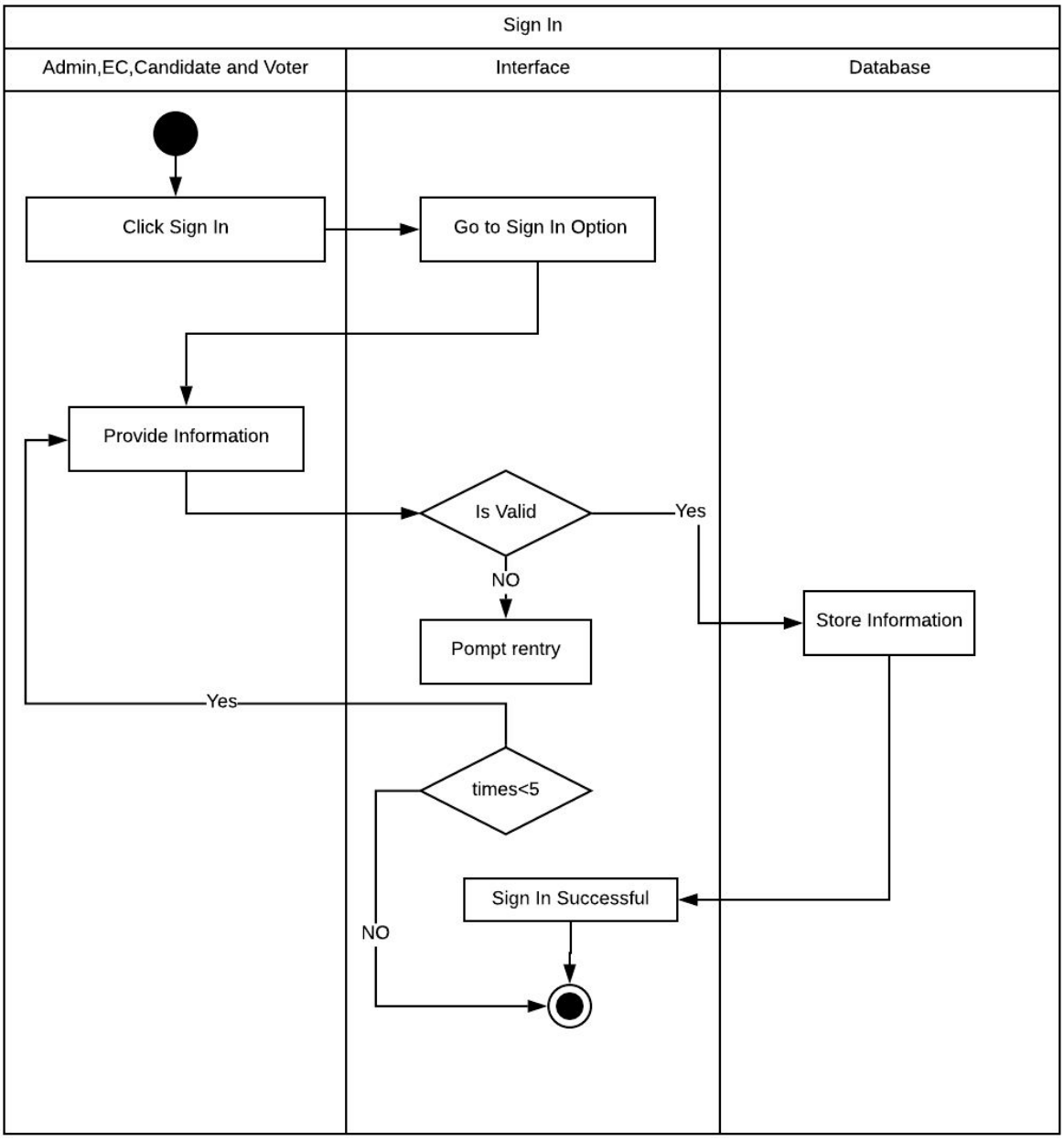
4.17 SWIMLANE DIAGRAM



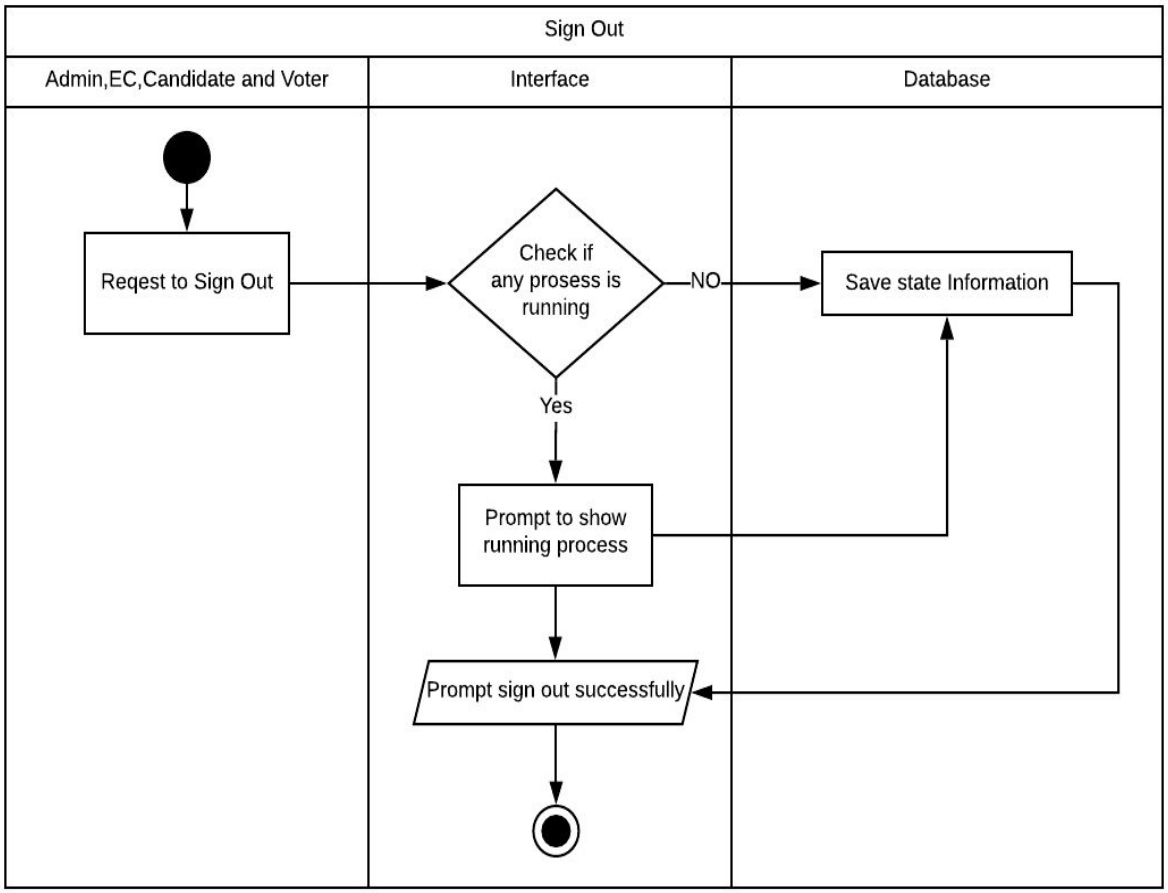
Swim Lane Diagram-1:Registration(Sign Up as EC)



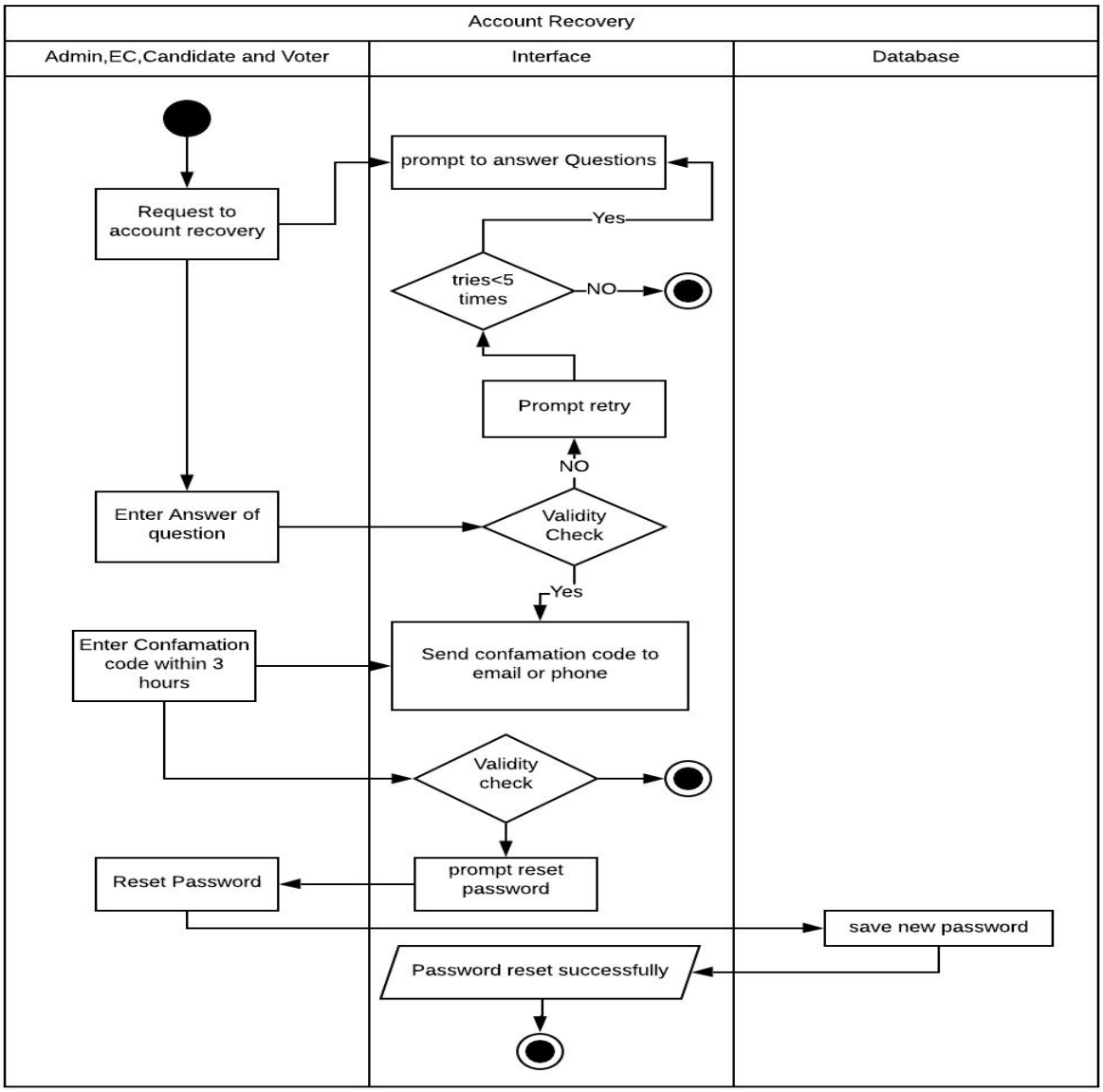
Swim Lane Diagram-2:Registration(Sign Up as Candidate and Voter)



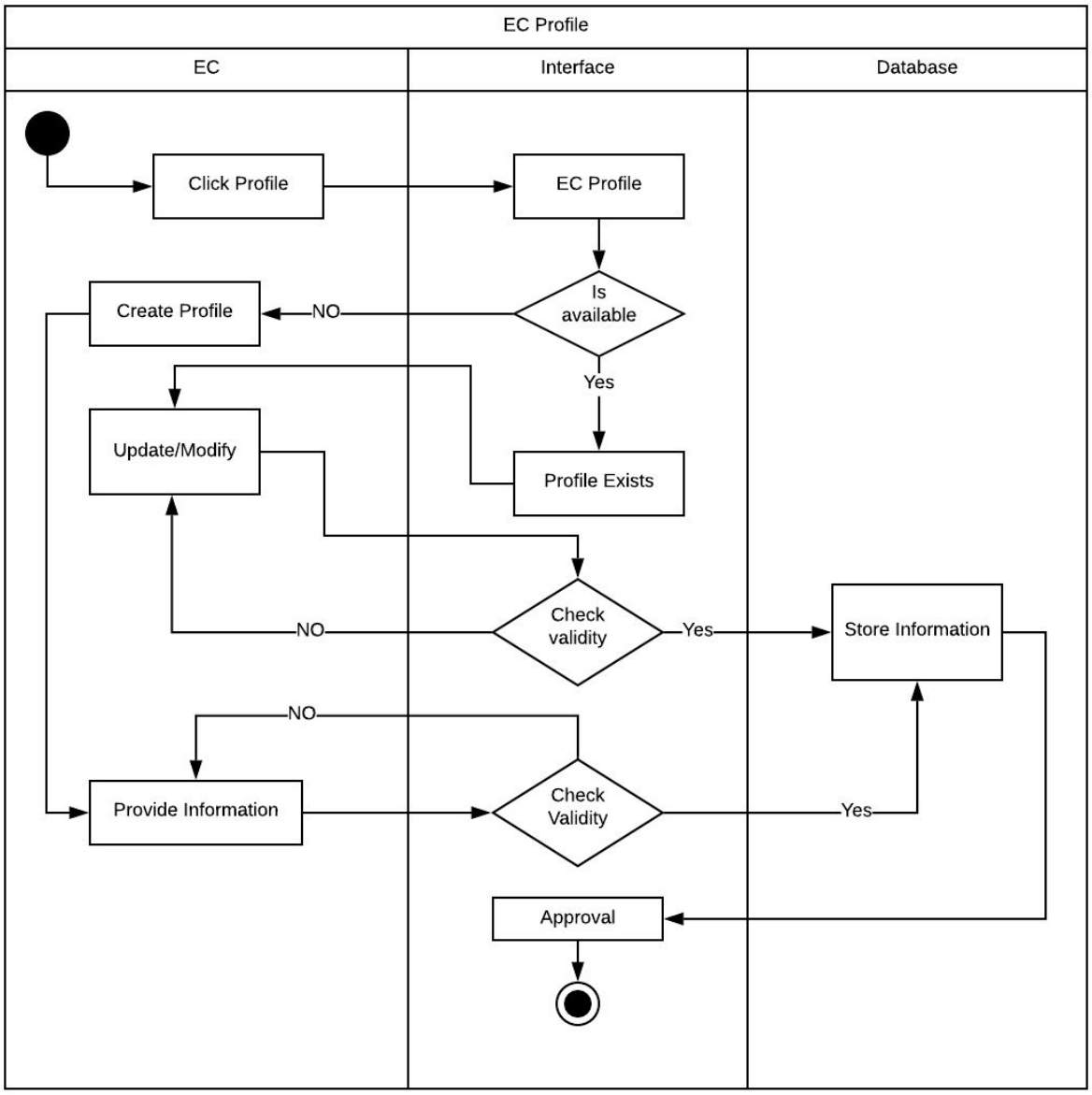
Swim Lane Diagram-3:Sign In



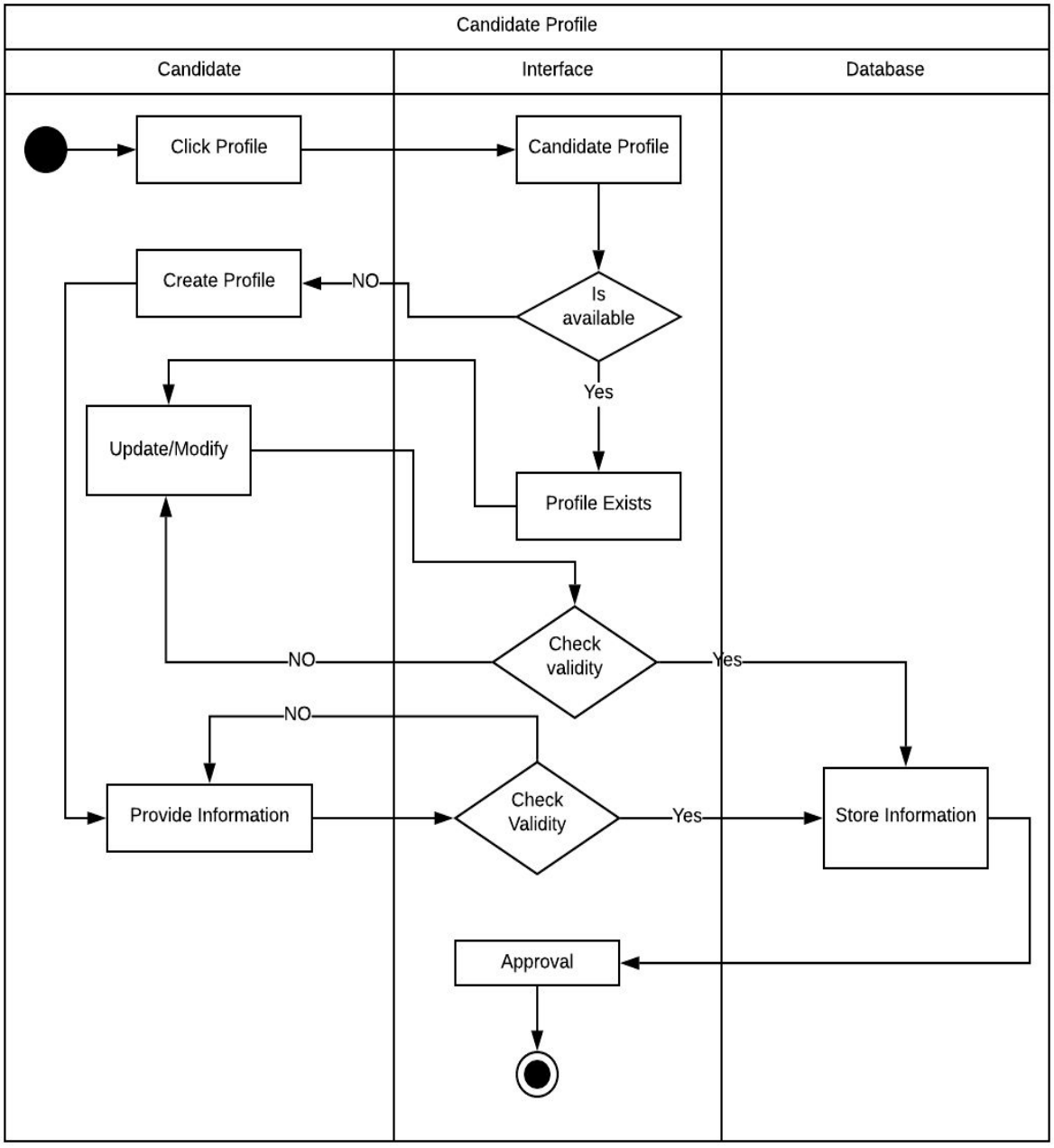
Swim Lane Diagram-4:Sign Out



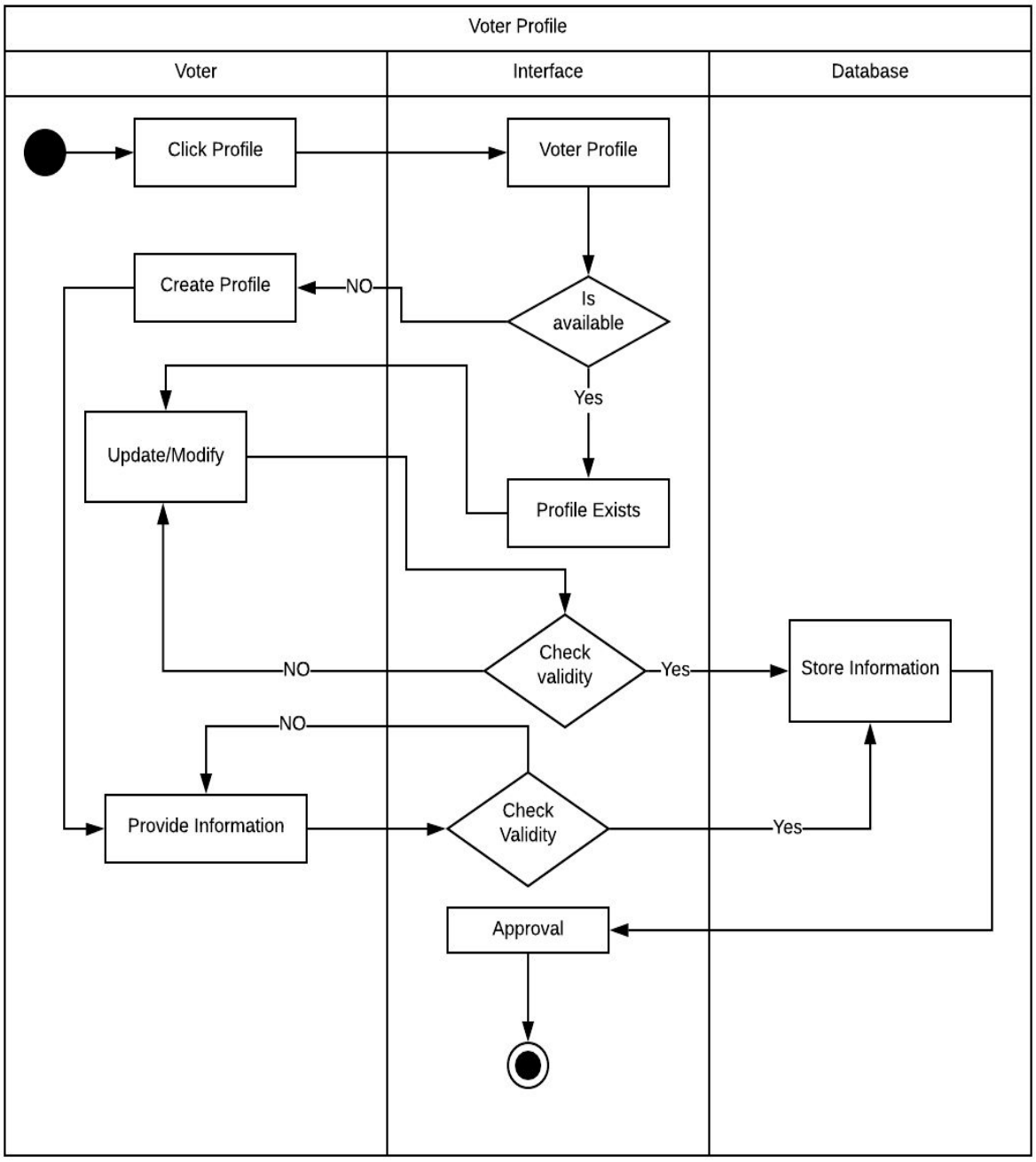
Swim Lane Diagram-5:Account recovery



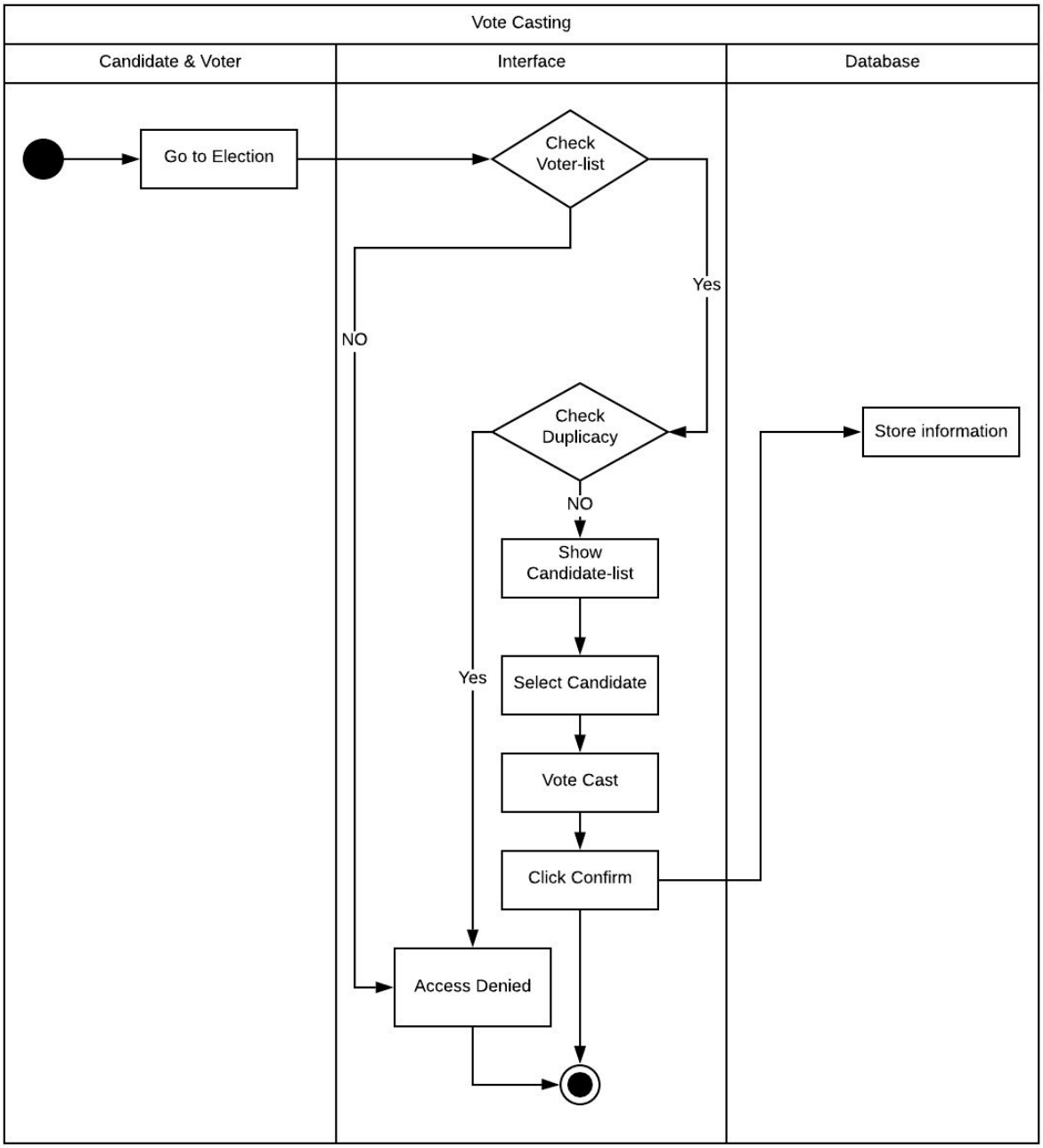
Swim Lane Diagram-6:EC Profile



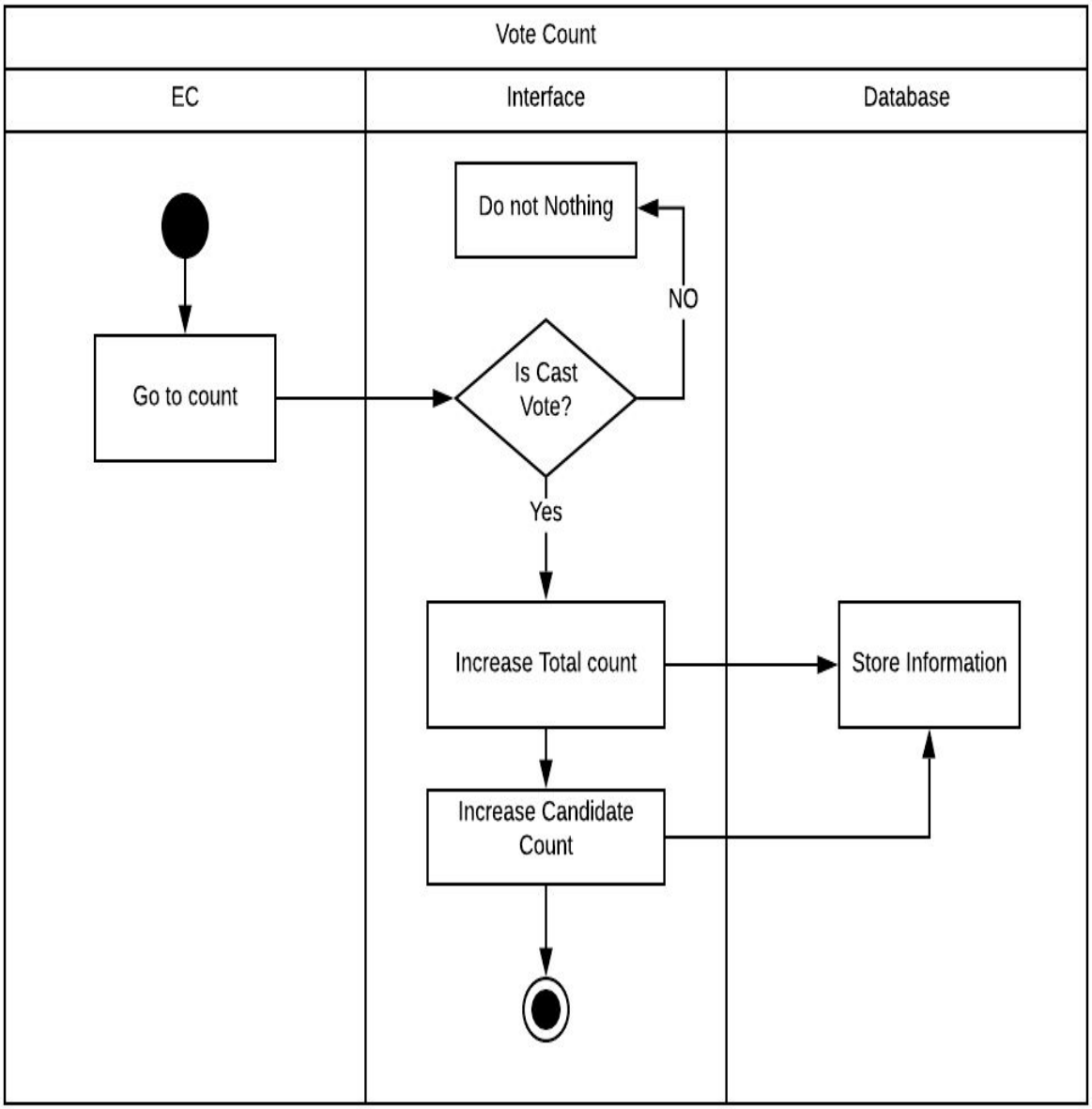
Swim Lane Diagram-7:Candidate Profile



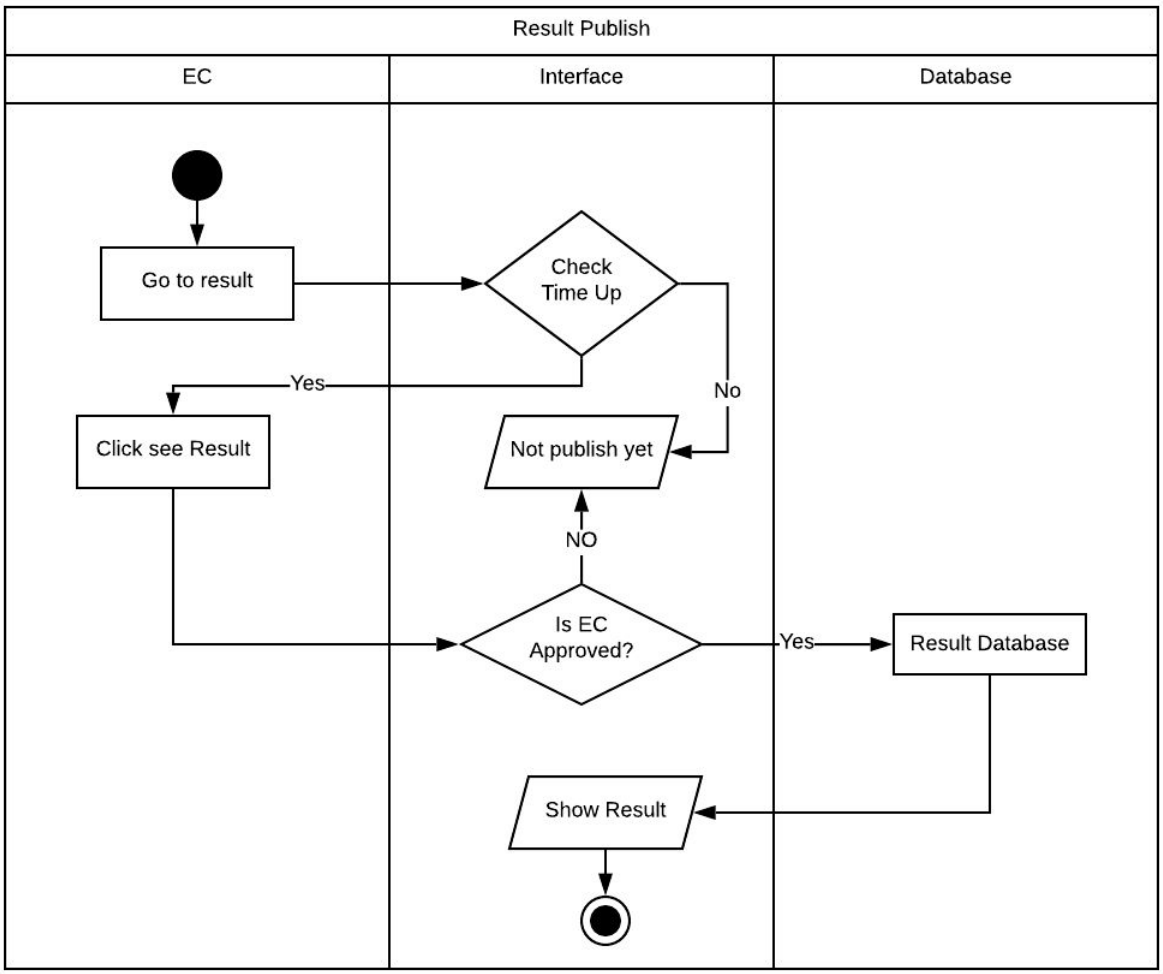
Swim Lane Diagram-8:Voter Profile



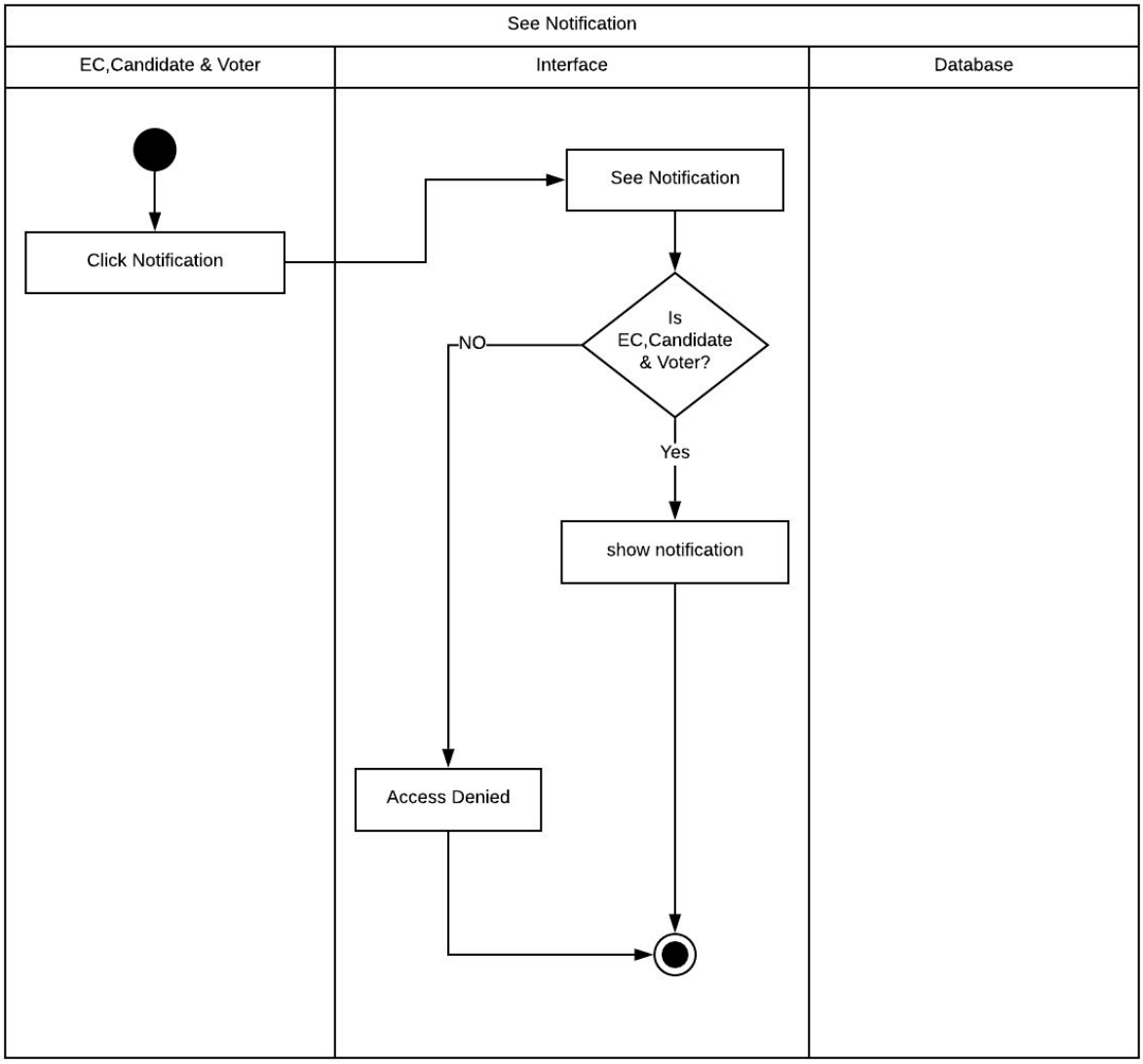
Swim Lane Diagram-9:Vote Casting



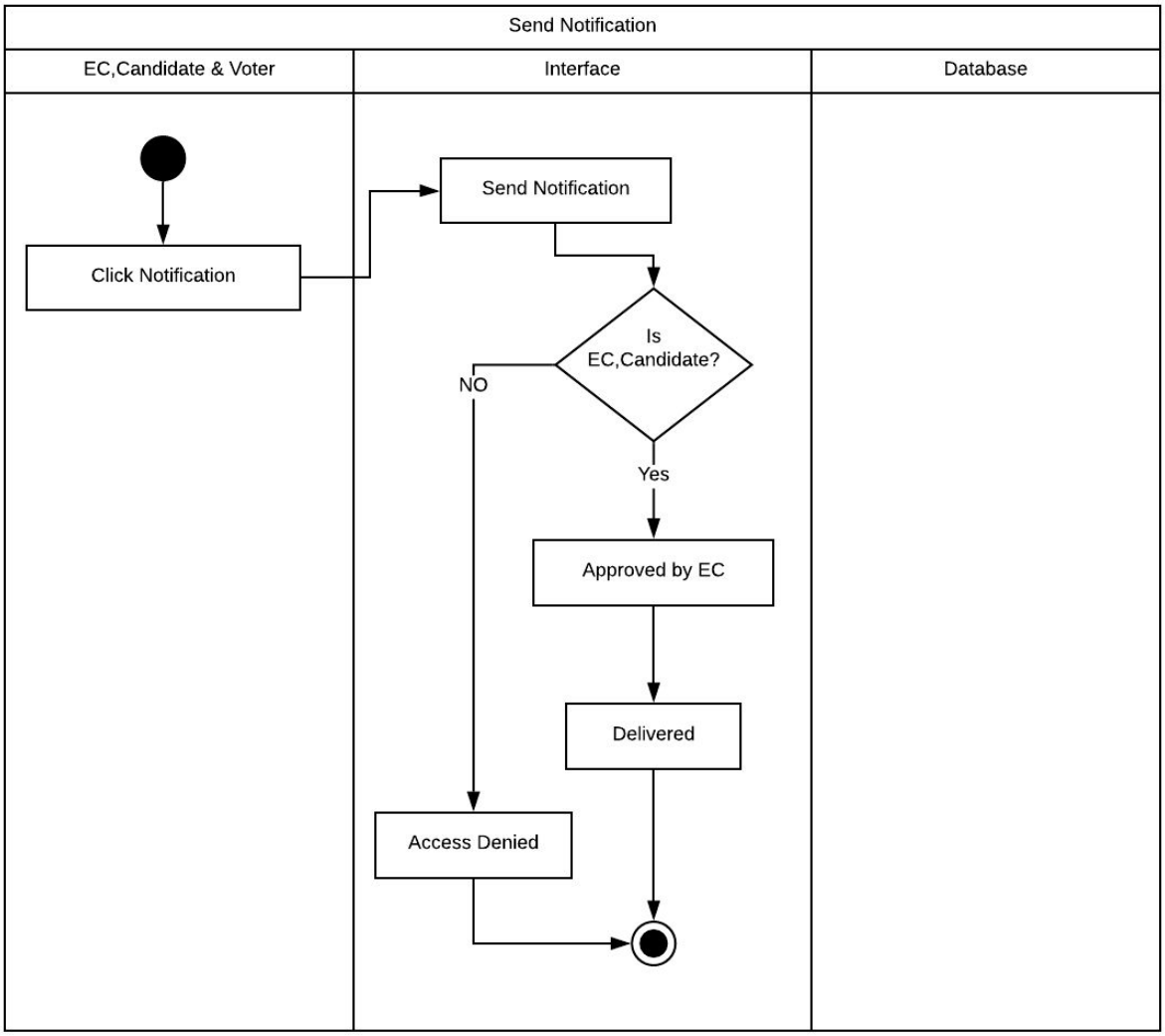
Swim Lane Diagram-10:Vote Count



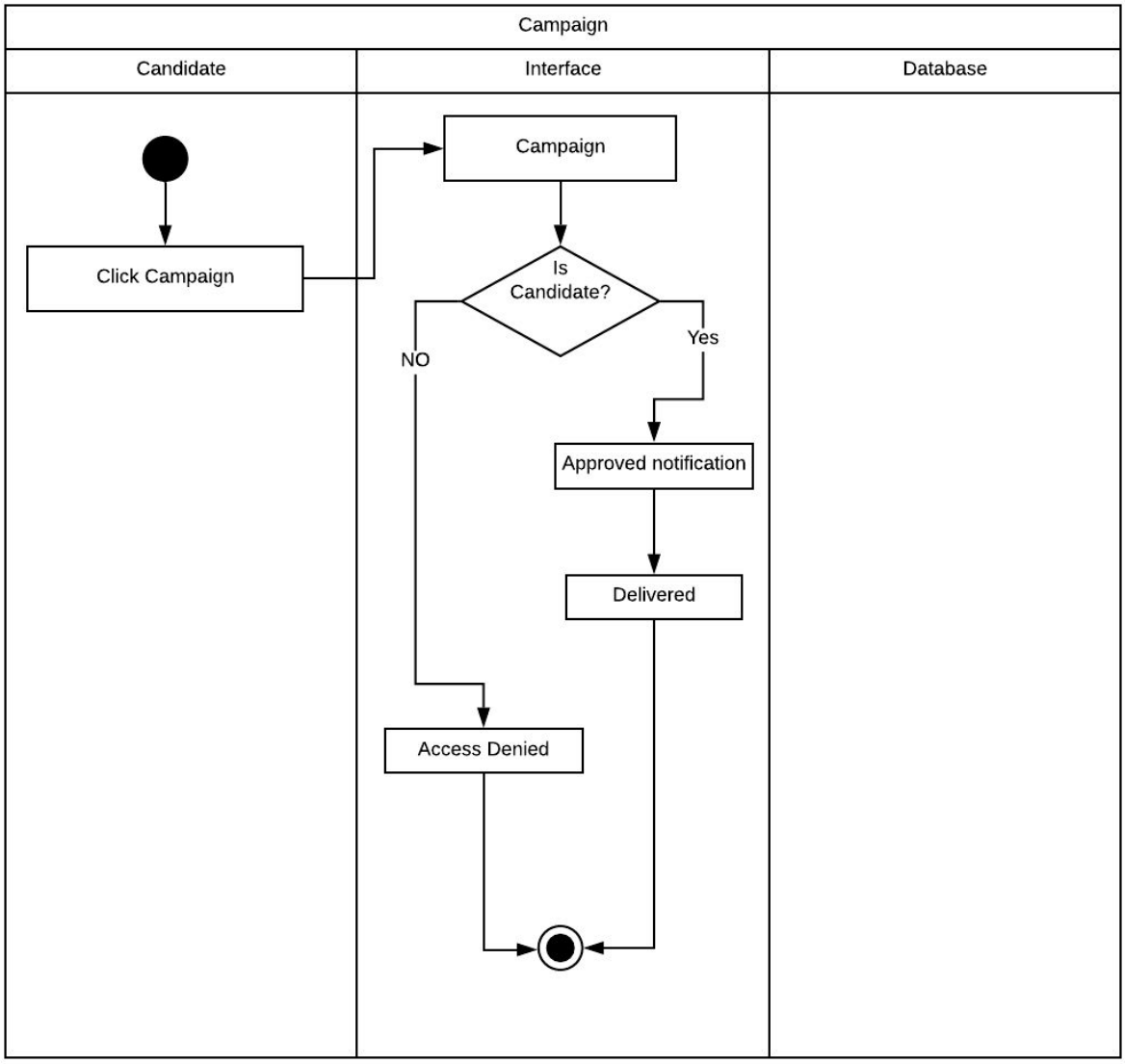
Swim Lane Diagram-11:Result Publish



Swim Lane Diagram-12:See Notification



Swim Lane Diagram-13:Send Notification



Swim Lane Diagram-14:Campaign

CHAPTER 5

DATA MODELING OF ONLINE VOTING SYSTEM

5.1 DATA MODELING CONCEPT

If software requirements include the necessity to create, extend or interact with a database or complex data structures need to be constructed and manipulated, then the software team chooses to create data models as part of overall requirements modeling. The entity-relationship diagram (ERD) defines all data objects that are processed within the system, the relationships between the data objects and the information about how the data objects are entered, stored, transformed and produced within the system.

5.2 DATA OBJECTS

A data object is a representation of composite information that must be understood by the software. Here, composite information means an information that has a number of different properties or attributes. A data object can be an external entity, a thing, an occurrence, a role, an organizational unit, a place or a structure.

5.3 NOUN IDENTIFICATION

We identified all the nouns whether they are in problem space or in solution space from our story.

Serial No	Noun	Problem/ Solution space	Attributes
1	User	s	20,21,22,23,24,25,26,27,28,29,30,31
2	Authentication	p	
3	Maintenance	p	
4	Automation	s	
5	Services	s	

6	Elicitation	s	
7	Account	s	11,13,14,15
8	Result	s	81, 82, 86, 88
9	Notification	s	30, 68
10	Management	p	
11	Admin	s	28,29
12	Election	s	
13	EC	s	20,21,22,23,24,25,26,27,28,29
14	Candidate	s	20-29,30,31,100
15	Voter	s	20,21,22,23,24,25,26,27,28,29
16	Category	s	
17	Registration	p	
18	Sign up	p	
19	Sign in	p	

20	First Name	s	
21	Last Name	s	
22	Gender	s	
23	Birthdate	s	
24	Current Address	s	
25	Permanent Address	s	
26	NID	s	
27	Contact no	s	
28	Email	s	
29	Password	s	

30	Image	s	
31	Symbol	s	
32	Validity	p	
33	Click	p	
34	Store	p	
35	Formate	s	
36	Verification	p	
37	Profile	s	13, 14, 15, 73, 74, 75
38	Database	s	
39	Collection	s	

40	System	s	
41	Creation	p	
42	Casting	p	
43	Vote	s	88, 89, 75
44	Factor	s	
45	Count	p	
46	Access	p	
47	Recovery	p	
48	Updation	p	
49	Confirmation	p	
50	Sign out	p	
51	Modification	p	
52	Approval	p	
53	Information	s	87, 95

54	Ages	s	
55	Area	s	
56	Proof	s	
57	Duplication	p	
58	Ten times		

59	Question	s	
60	Answer	s	
61	Data	s	
62	Status	s	
63	Characteristics	p	
64	Number	s	
65	Unsave	s	
66	Commissioner	s	
67	Date	s	
68	File	s	
69	Campaign	p	
70	Upload	p	
71	Website	s	
72	Attribute	s	
73	Login time	s	
74	Logout time	s	
75	Cast Status	s	
76	Identification	p	
77	Voter	s	

78	Voter list	s	14, 15
79	Condition	s	
80	Candidate list	s	14, 83
81	Total Count	s	
82	Vote time	s	
83	Hash	s	
84	Encryption	s	
85	Rules	s	
86	Vote date	s	
87	Purpose	p	
88	Election id	s	
89	Time	s	
90	Description	s	
91	Digit	s	
92	Input	p	
93	Criteria	s	
94	Range of time	s	
95	Picture	s	
96	Interface	p	

97	Propaganda	s	
98	Security	p	
99	Capability	s	
100	Ballot no.	s	
101	Service	p	

5.4 POTENTIAL DATA OBJECTS

- ★ User : 20,21,22,23,24,25,26,27,28,29,30,31
- ★ Admin: 20,21,22,23,24,25,26,27,28,29
- ★ EC: 20-29
- ★ Voter: 20-29
- ★ Candidate: 20-29, 30, 31
- ★ Information: 87, 95
- ★ Vote: 75, 88, 89
- ★ Notification: 30, 68, 89
- ★ Account: 13, 14, 15
- ★ Result: 81, 82, 86, 88
- ★ Profile: 13, 14, 15, 73, 74, 75

5.5 ANALYSIS FOR FINALISING DATA OBJECTS

- Admin, EC, Candidate, Voter have some common attributes. So their common attributes can be stored as User.

- Vote tables contain the necessary info and vote casting and counting.

- User receives and send Notification. System generates notification. So, messages needs to be saved and all information must be stored.

- Profile keeps information of the user profile. As account and profile seems same meaning, so we can keep a table only to work with.

- Details of users is stored in EC, Candidate and Voter respectively for updating information and tracking his activities.

- Result database contains all the info of an election for future use.

5.6 FINAL DATA OBJECTS

Final Data Objects Table

1	**User**: First name, last name, Gender, Birthdate, Present Address, Permanent address, Designation, NID, Email, Password, Contact no.
2	**Admin**: Admin id, User id, Full name, Permanent address, Current address, Designation, NID, Email, Password
3	**Profile**: Username, User id, Email, Password, Login_time, Logout_time
4	**EC:** User id, Email, password, election_id
5	**Voter:** NID, Email, Password, Voting_Status
6	**Candidate:** Symbol, image, candidate_id, NID
7	**Vote:** NID, cast_status, voting_time, NID
8	**Result:** election_id,EC_id, candidate_id, Candidate count
9	**Notification:** Time, Notification id, Receiver User Id, Type, File/Description

5.7 DATA OBJECT RELATIONSHIPS

Data objects are connected to one another in different ways. They are shown below.

Admin sets EC

EC publish Result

EC, Candidate and Voter are a user

EC and Candidate send Notification

Voter casts vote

Candidate is also a voter

5.8 ER DIAGRAM

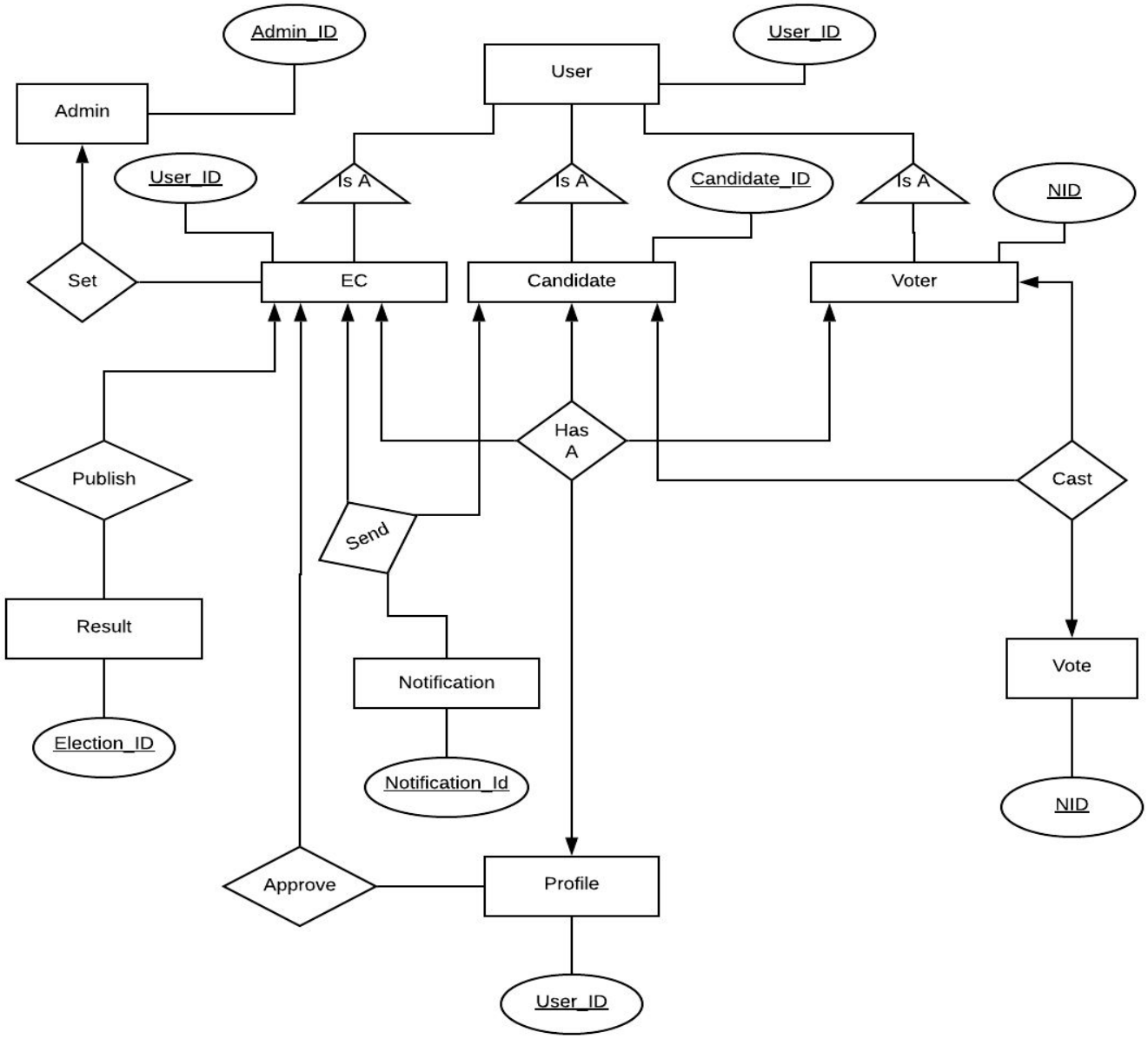


Figure:ER Diagram

5.9 SCHEMA DIAGRAM

USER

Attributes	Type	Size
First Name	Varchar2	80
Last Name	Varchar2	80
Gender	Varchar2	80
Birthdate	Varchar2	80
NID	Varchar2	80
Contact no	Varchar2	80
Email	Varchar2	80
Password	Varchar2	80
Current Address	Varchar2	80
Permanent Address	Varchar2	80

ADMIN

Attributes	Type	Size
Username Admin_id Designation Backup question Answer	Varchar2 Varchar2 Varchar2 Varchar2 Varchar2	80 80 80 80 80

EC

Attributes	Type	Size
User id Admin_id Email Password election_id	Number Number Varchar2 Varchar2 Number	10 10 80 10 10

CANDIDATE

Attributes	Type	Size
Candidate_id User_id NID	Number Number Number	10 10 10

Symbol Image	BLOB BLOB	1MB 1MB

VOTER

Attributes	Type	Size
NID User_id Email Password Voting_Status	Number Number Varchar2 Varchar2 Varchar2	10 10 80 10 10

VOTE

Attributes	Type	Size
NID Candidate_id voting_time	Number Number Varchar2	10 10 40

RESULT

Attributes	Type	Size
Election_id EC_id Candidate_id	Number Number Number	10 10 10

Candidate_Count	Varchar2 Number	40 10

NOTIFICATION

Attributes	Type	Size
Time User_id Notification id Type File/Description	Varchar2 Number Number Varchar2 Varchar2/Blob	40 10 10 40 1MB

PROFILE

Attributes	Type	Size
Username User id Email Password Login_time Logout_time	Varchar2 Number Varchar2 Varchar2 Varchar2 Varchar2	40 10 80 10 40 40

CHAPTER 6

CLASS-BASED MODELING

.

6.1 CLASS BASED MODELING CONCEPT

We did the Class-based modeling to represent the objects that will be manipulated by the system, relationships between the objects and the collaborations that occur between the classes that are defined and the operations that will be applied to the objects.

6.2 GENERAL CLASSIFICATION

As we need to identify the potential classes, here, we have first selected the nouns from the solution space of the story. The nouns were then characterized in seven general classifications. The seven general characteristics are :

1. External entities
2. Things
3. Events
4. Roles
5. Organizational units
6. Places
7. Structures

The specifications of the nouns according to the general classifications are given below:

Serial No	Noun	General Classification
1	System	2,4,7
2	Authentication	3,5
3	Maintenance	3,5
4	Automation	5,7

5	Services	3,4,5
6	Elicitation	4,5
7	Profile	4,5
8	Result	5,7
9	Notification	2,3,5,7
10	Management	5,7
11	Admin	4,5,7
12	Election	3,6,7
13	Commissioner	4,5,7
14	Candidate	4,5,7
15	Voter	4,5,7
16	Category	5,7
17	Registration	3
18	Sign up	3
19	Sign in	3

20	First Name	
21	Last Name	
22	Gender	
23	Birthdate	
24	Current Address	
25	Permanent Address	
26	NID	
27	Contact no	
28	Email	

29	Password	
30	Image	
31	Symbol	
32	Validity	
33	Click	
34	Store	
35	Format	
36	Verification	
37	Account	4,5
38	Database	2,4,7
39	Collection	

40	User	4,5,7
41	Creation	
42	Casting	4
43	Vote	2
44	Factor	
45	Count	
46	Access	4
47	Recovery	3
48	Updation	
49	Confirmation	
50	Sign out	3
51	Modification	
52	Approval	3,4

53	Information	
54	Ages	
55	Area	
56	Proof	
57	Duplication	3
58	Ten times	

59	Question	
60	Answer	
61	Data	1
62	Individual	
63	Characteristics	
64	Number	
65	Unsave	
66	EC	4,5,7
67	Date	
68	File	
69	Campaign	3
70	Upload	3
71	Website	
72	Attribute	
73	Login time	
74	Logout time	
75	Additional info	
76	Identification	3

77	Voter	4,5,7

78	Voter list	2,5
79	Condition	
80	Candidate list	2,5
81	Total Count	3,7
82	Vote time	
83	Hash	
84	Encryption	
85	Rules	
86	Vote date	
87	Purpose	
88	Election id	
89	Time	
90	Description	
91	Digit	
92	Input	
93	Criteria	
94	Range of time	
95	Picture	
96	Interface	2,4,7

97	Propaganda	
98	Security	5
99	Capability	5,6

100	Ballot no.	2
101	Service	3,5

Table: General Classification

6.3 SELECTION CRITERIA

A potential class becomes a class if it fulfills all six characteristics stated below :

- Retain information
- Needed services
- Multiple attributes
- Common attributes
- Common operations
- Essential requirements

Here, we have selected the potential classes by following these selection criteria.

No	Noun	Selection Criteria
1	System	1
2	Authentication	3
3	Services	2,5
4	Result	1
5	Notification	3,4,5
6	Admin	1-5
7	Election	5,6
8	EC	1-5
9	Candidate	1-5
10	Voter	1-5
11	User	1-5
12	Interface	1
13	Database	1
14	Account	1-5

15	Profile	1-5
16	Approval	1,6

6.4 ASSOCIATE NOUN AND VERB IDENTIFICATION

We now identify the nouns and verbs associated with the potential classes to better find out the attributes and methods of each class.

No	Potential Class	Noun	Verb
1	User	First name,Last name, password, email address, NID, Gender,Birth date,Contact No ,Current Address,Permanent Address	Updating user information(full name,current address, NID, contact etc..), log out, See notification
2	Admin	User name ,email address, Password, NID, Contact	Approving account ,Adding EC,Removing EC, Updating EC information
3	EC	User name ,email address, Password, NID, Contact	Approving account ,Adding EC,Removing EC, Updating EC information,Adding candidate,Removing candidate,sending Notification,Updating Profile,See Notification,Adding Voter-list,Publishing result,See result
4	Candidate	User name ,email address, Password, NID, Contact	See Notification,Sending Notification,Cast vote,Updating Profile,Campaigning,

			See result
5	Voter	User name ,email address, Password, NID, Contact	See Notification,Cast vote,Updating Profile,See result
6	Result	Election ID,vote count,Total Count	Count Vote, Publishing result
7	Profile	Username,Password,User ID	Updating contact, Generating User ID,Reset password, Matching password, Changing password
8	Authentication	User,Database,Interface, system	Login, Signup, Recovery account
9	System	Database, Interface	Delete, Update, Store, Access Database, Authentication, Verify information
10	Database	Table name,Status	Insert,Update,Delete,Search
11	Notification	Notification ID, Time	Generate Notification ID, Send Notification, Receive Notification
12	Vote	Election ID,Candidate-list	Cast vote,Counting vote,Check Duplicacy
13	Interface		

6.5 ATTRIBUTE SELECTION

After identifying the classes, we have specified their attributes and methods.

No	Name	Attributes
1	User	• First name • Last name • Password • Email address • NID • Gender • Birth date • Contact No • Current Address • Permanent Address
2	Admin	• User name • email address • Password • NID • Contact
3	EC	• User name • email address • Password • NID • Contact
4	Candidate	• User name • email address • Password • NID • Contact • Symbol • Image • Candidate Id
5	Voter	• User name

		• email address • Password • NID • Contact
6	Result	• Election ID • Candidate count[] • Total Count
7	Profile	• Username • Password • User ID
8	Authentication	• User • database • Interface
9	System	• Database • Interface
10	Database	• Table name • Status
11	Notification	• Notification ID • Time
12	Vote	• Election ID • Candidate-list • Total count • Candidate count[ ]
13	Interface	• User

table:Attribute selection

6.6 METHOD IDENTIFICATION

After identifying the classes, we have specified their methods.

No	Potential Class	Verb
1	User	• validateNID() • SetFirst name() • GetFirst name() • GetLast name() • GetLast name() • SetPassword() • GetPassword() • SetEmail address() • GetEmail address() • SetGender() • GetGender() • SetBirth date() • GetBirth date() • SetContact No() • GetContact No() • SetCurrent Address() • GetCurrent Address() • SetPermanent Address() • GetPermanent Address() • setProfile() • getProfile() • getNID() • setNID() • createNotification()
2	Admin	• Approve EC() • Remove EC() • Update EC information()
3	EC	• ApproveProfile() • AddEC()

		• RemoveEC() • UpdateECinformation() • Addcandidate() • Removecandidate() • sendNotification() • UpdateProfile() • SeeNotification() • AddVoter-list() • Publishresult() • Seeresult()
4	Candidate	• See Notification() • Cast vote() • Update Profile() • Campaign() • See result()
5	Voter	• See Notification() • Cast vote() • Update Profile() • See result()
6	Result	• getCount Vote() • Publish result()
7	Profile	• Update contact() • Generate User ID() • Reset password() • Match password() • Change password()
8	Authentication	• Signin() • Signup() • Recovery account() • validateinput() • SignOut()
9	System	• Delete() • Store() • Access Database() • Authentication()
10	Database	• Insert() • Update()

		• Delete() • Search()
11	Notification	• Generate Notification ID() • Send Notification() • Receive Notification()
12	Vote	• Cast vote() • Count vote() • Check Duplicacy ()
13	Interface	• showAuthenticationAction() • showMenu() • getInput() • getAction() • showNotification() • ShowResult()

Table : Method Identification

6.7 FINALIZING CLASSES

To identify final classes we need to first check that if there can be any hierarchies or merges. These are given below:

User: is a parent class of the users Admin, EC, Candidate and Voter. We are bound to make it a super class of the child classes with some of their common attributes.

Admin: Admin is the first user of our website. So, we need to make an admin class according to its functionalities.

EC: Election Commissioners are the head of an election. They control everything of the election. So it's our duty to make a class called EC.

Candidate: Candidates are the people for whom we arrange an election to select the right candidates according to people's choice.

Voter: Voter the man who select candidates to make successful election. Voter profile and function is enough to make a class.

Vote: Vote casting, Vote count and necessary info make vote a class. Besides vote is the main keyword of an election process.

Authentication: Authentication includes the registration process and the access account functions. That's why we create a class called authentication.

System: All the automated task will be done by system. So, so system should be a class.

Interface: Interface class includes the general methods those will be implemented by other classes.

Database: Database class will have the responsibility to insert, update, delete databases. So, we need to make a class called database.

6.8 CLASS CARDS

After identifying our final classes we have generated the following class cards.

System	
Attributes	**Method**
• Database • Interface • Users • Authentication	• Delete() • Store() • Access Database()
Responsibilities	**Collaborators**
• Handling database • Creating interface • Authenticate	• Authentication • Interface • Database • User

Table : Class Card for System Class

Interface	
Attributes	**Method**

• User [ ] • Result • Notification[ ]	• showAuthenticationAction() • showMenu() • getInput() • getAction() • showNotification() • showResult()
Responsibilities	**Collaborators**
• Creating interface • Authentication	• System • Authentication • Notification • Result

Table : Class Card for Interface

Authentication	
Attributes	**Method**
• User • Interface • Database	• Signup() • validateInput() • login() • Recovery account() • SignOut()
Responsibilities	**Collaborators**
• getting input for signup/login • Set user info to database	• User • System

Table : Class Card for authentication

User	
Attributes	**Method**
• First name • Last name • Password • Email address • NID • Gender • Birth date • Contact No • Current Address • Permanent Address	• validateNID() • SetFirst name() • GetFirst name() • GetLast name() • GetLast name() • SetPassword() • GetPassword() • SetEmail address() • GetEmail address() • SetGender() • GetGender() • SetBirth date() • GetBirth date() • SetContact No() • GetContact No() • SetCurrent Address() • GetCurrent Address() • SetPermanent Address() • GetPermanent Address() • setProfile() • getProfile() • getNID() • setNID() • createNotification()
Responsibilities	**Collaborators**
• providing user data • creating Profile • View Notification	• Authentication • System • Candidate • Voter • EC • Admin • Notification

Table : Class Card for User

Admin	
Attributes	**Method**
• User name • email address • Password • NID • Contact	• ApproveEC() • RemoveEC() • UpdateECInformation()
Responsibilities	**Collaborators**
• EC Create • Approving Request for election	• System • EC

Table : Class Card for Admin

Notification	
Attributes	**Method**
• Notification ID • Time	• Generate Notification ID() • Send Notification() • Receive Notification()
Responsibilities	**Collaborators**
• Managing notification	• EC • Candidate • Voter • Interface

Table : Class Card for Notification

Database	
Attributes	**Method**
• Table name • Status	• Insert() • Update() • Delete() • Search()
Responsibilities	**Collaborators**
• Insert element • Update element • Delete element • Search element	• System

Table : Class Card for Database

EC	
Attribute	**Method**
• User name • Email address • Password • NID • Contact	• ApproveProfile() • AddEC() • RemoveEC() • UpdateECinformation() • Addcandidate() • Removecandidate() • sendNotification() • UpdateProfile() • SeeNotification() • AddVoter-list() • Publishresult() • Seeresult()
Responsibilities	**Collaborators**
• verifying Candidate & voter data • creating profile • Publishing result • Notification	• System • Notification • Voter • Candidate

Table : Class Card for EC

Candidate	
Attribute	**Method**
• User name • Email address • Password • NID • Contact • Symbol • Image • Candidate Id	• See Notification() • Cast vote() • Update Profile() • Campaign() • See result()
Responsibilities	**Collaborators**
• Creating profile • View result • Vote casting • Campaigning • Notification	• User • Notification • Voter • Vote

Table : Class Card for Candidate

Voter	
Attribute	**Method**
• User name • Email address • Password • NID • Contact	• See Notification() • Cast vote() • Update Profile() • See result()
Responsibilities	**Collaborators**
• Creating profile • View result	• User • Notification

• Vote casting • View Notification	• Vote

Table : Class Card for Voter

Vote	
Attribute	**Method**
• Election ID • Candidate-list • Total count • Candidate count[ ]	• Cast vote() • Count vote() • Check Duplicacy ()
Responsibilities	**Collaborators**
• Casting vote • Counting vote • Checking Duplicacy	• Candidate • Voter • Result

Table : Class Card for Vote

Result	
Attribute	**Method**
• Election ID • Candidate count[] • Total Count	• getCount Vote() • Publish result()
Responsibilities	**Collaborators**
• Get all count • Showing Result	• EC • Vote

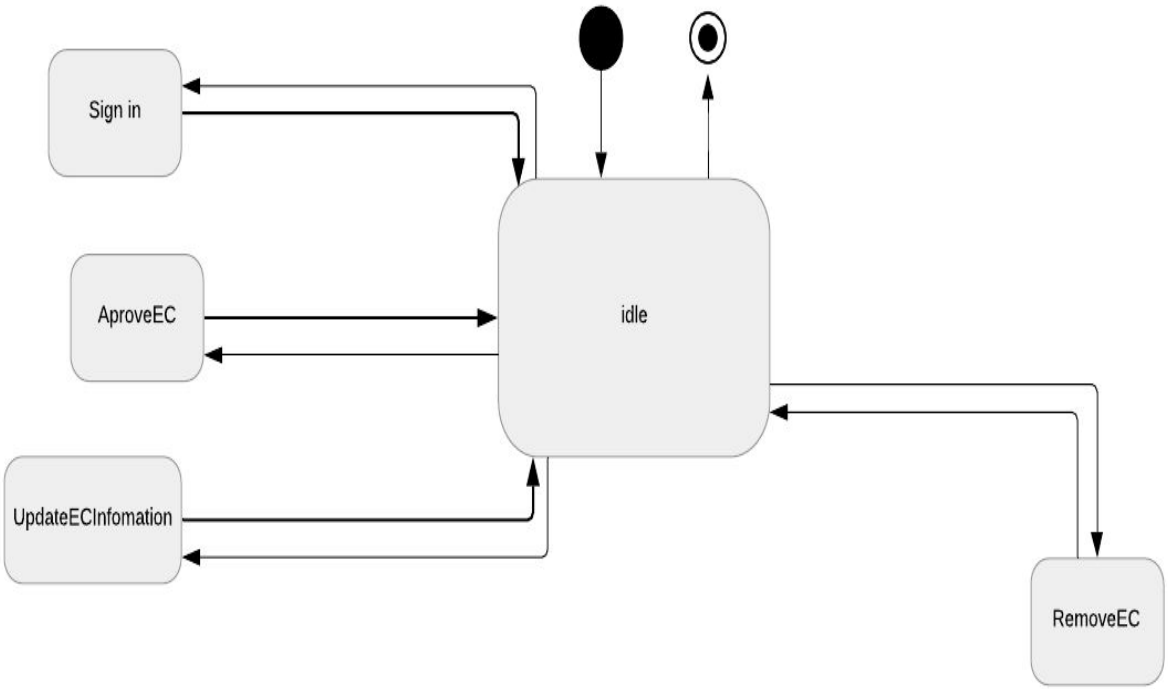
`

Table : Class Card for Result

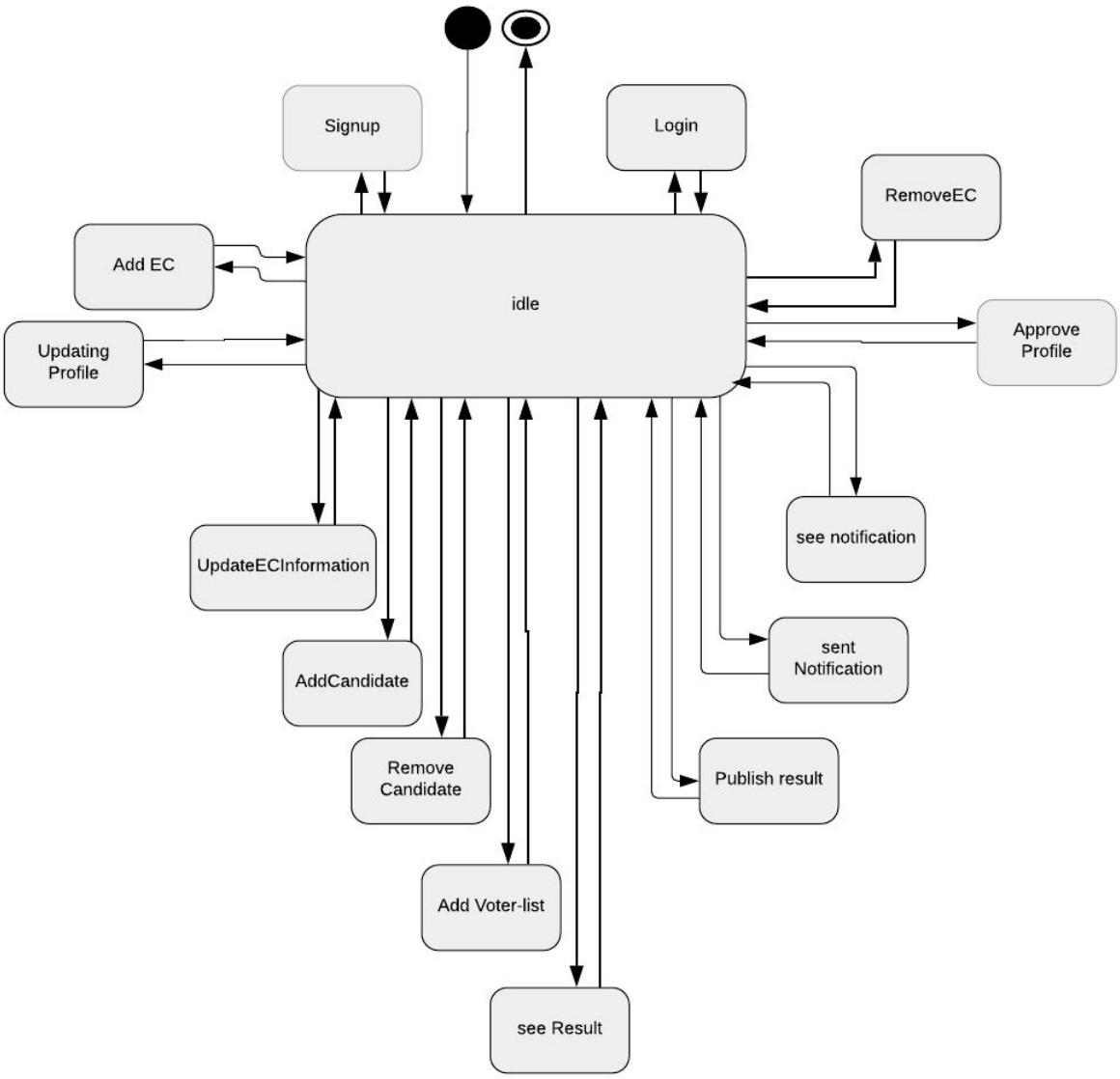
Chapter 7

BEHAVIORAL MODELING OF ONLINE VOTING SYSTEM

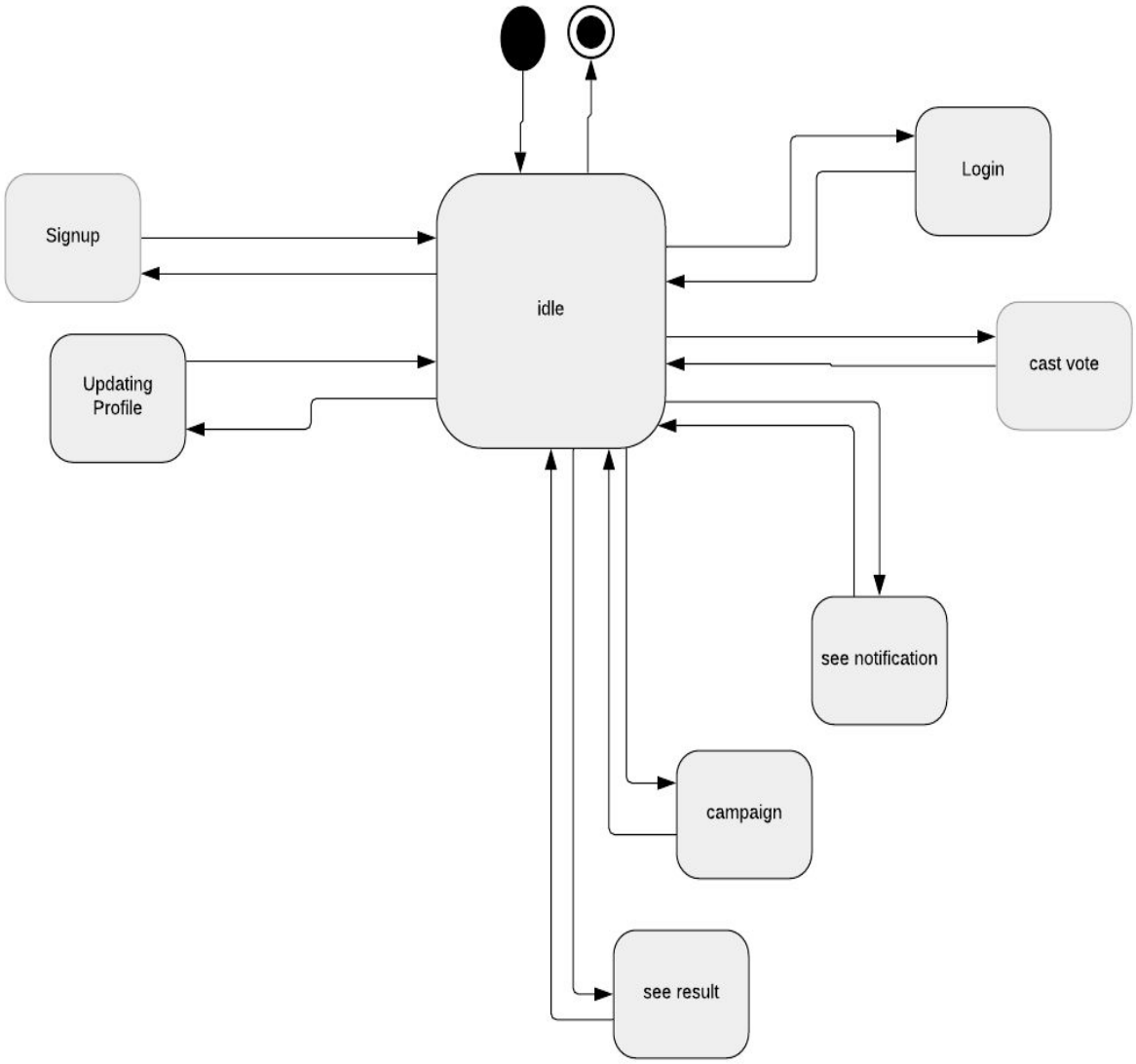
7.1 STATE TRANSITION DIAGRAM



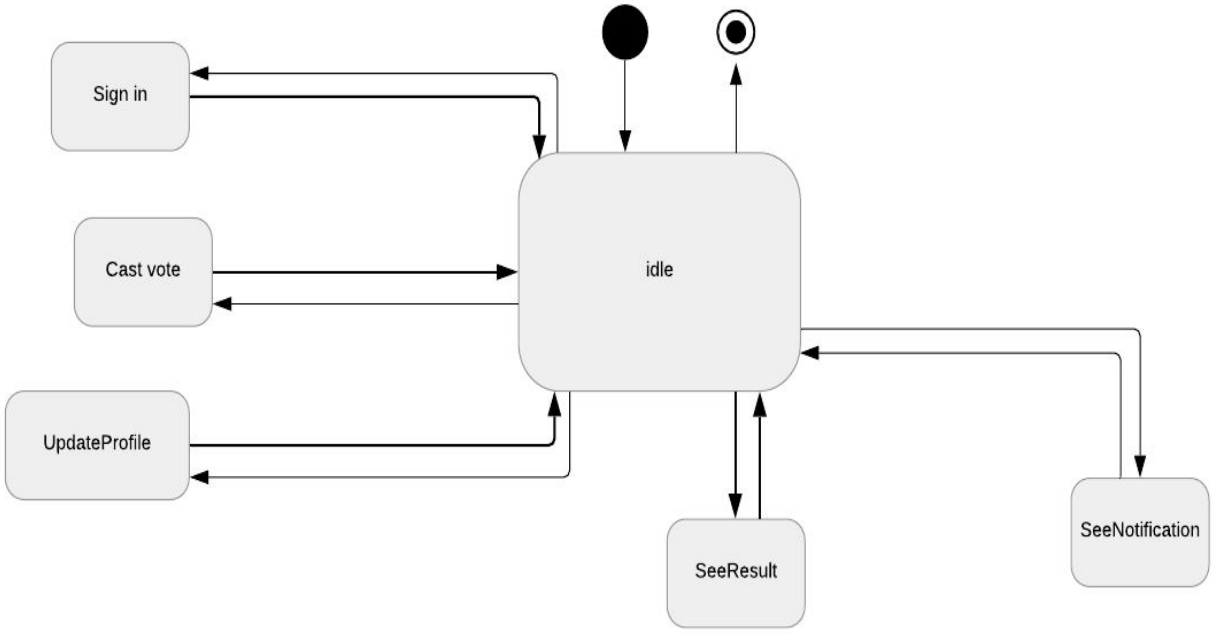
State Diagram:Admin



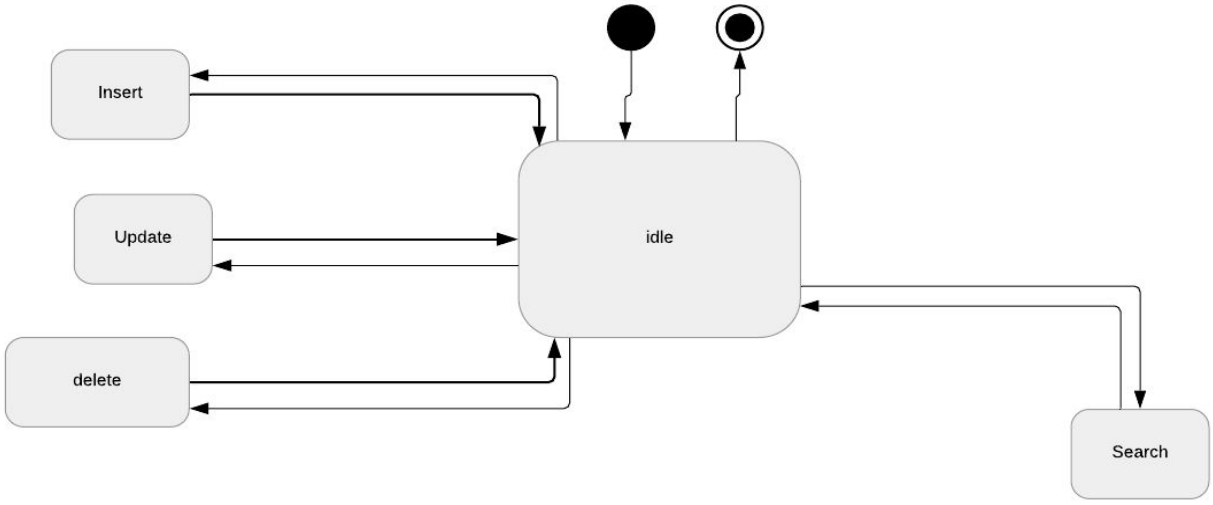
State Diagram:EC



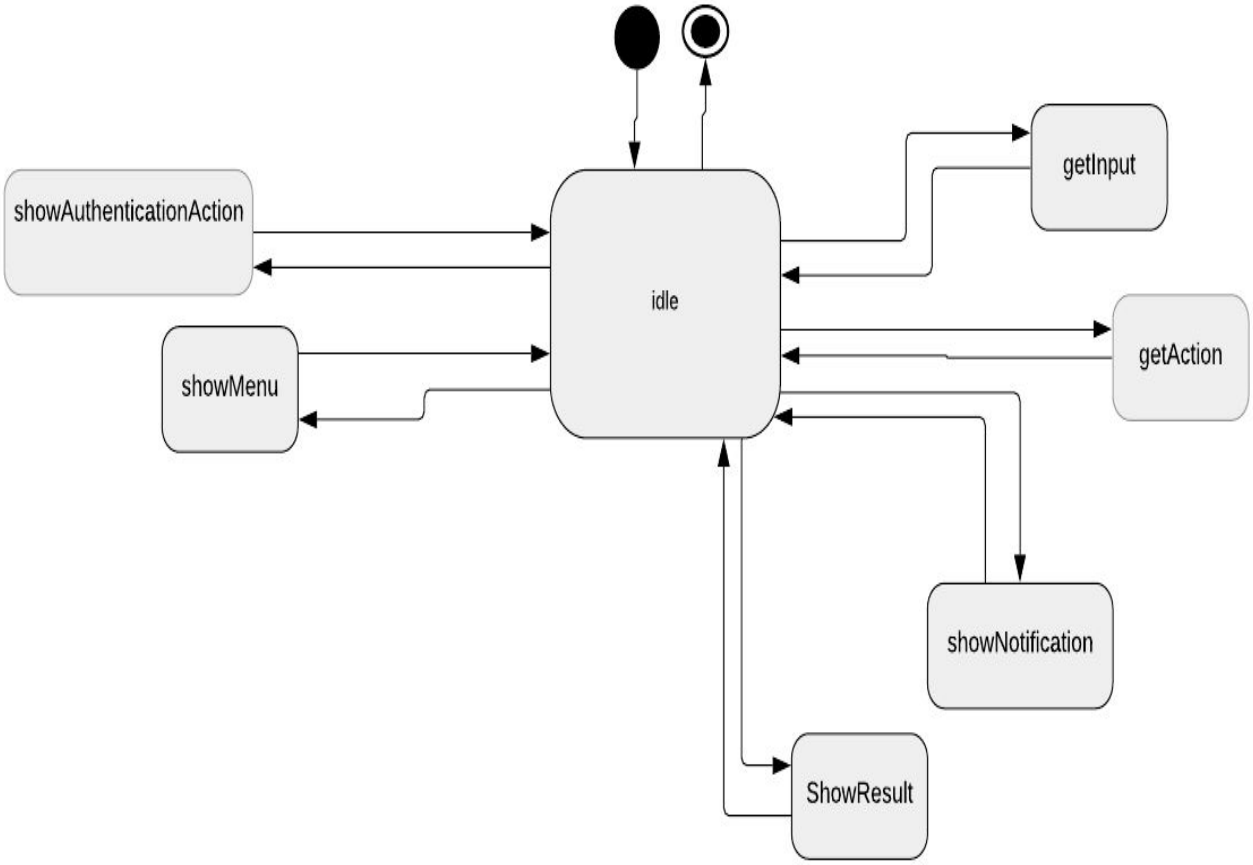
State Diagram:Candidate



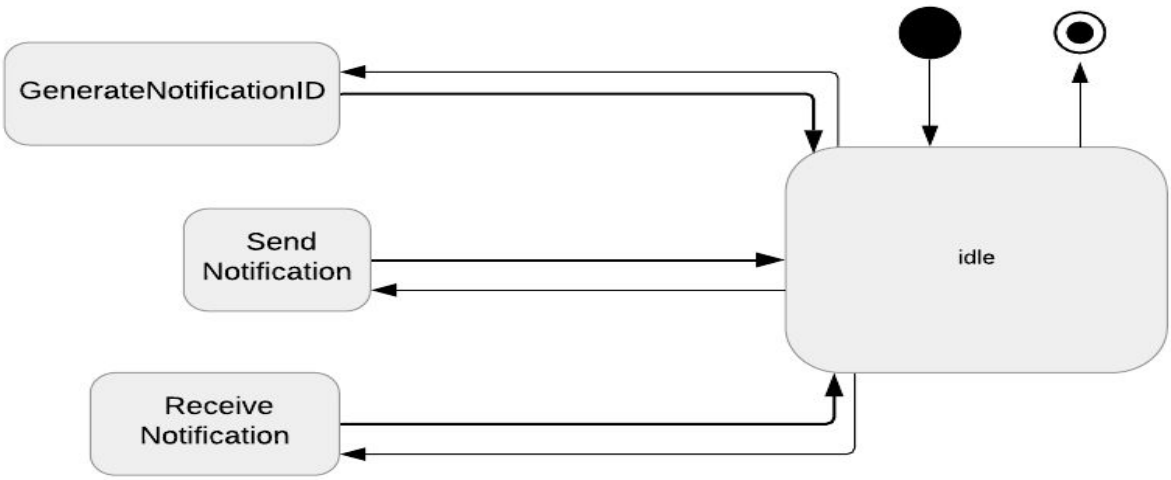
State Diagram:Voter



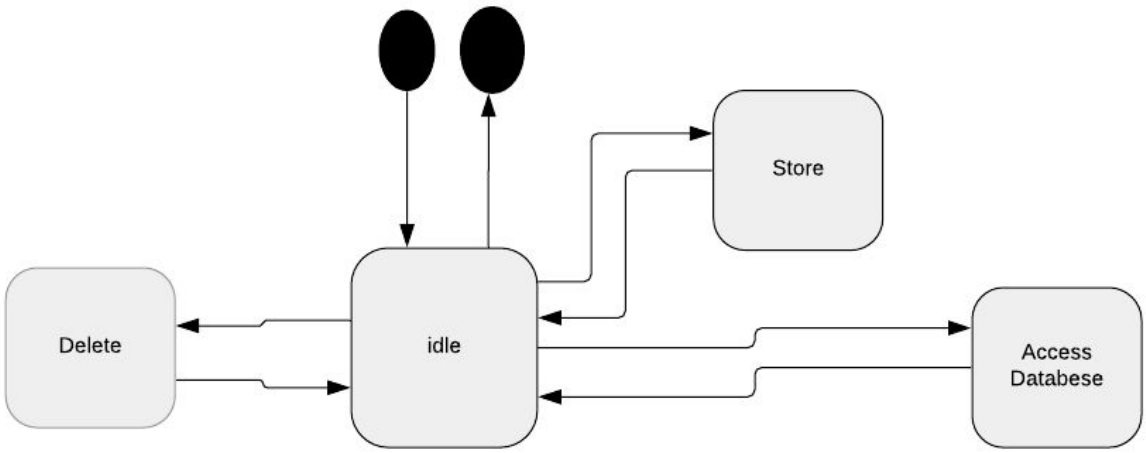
State Diagram:Database



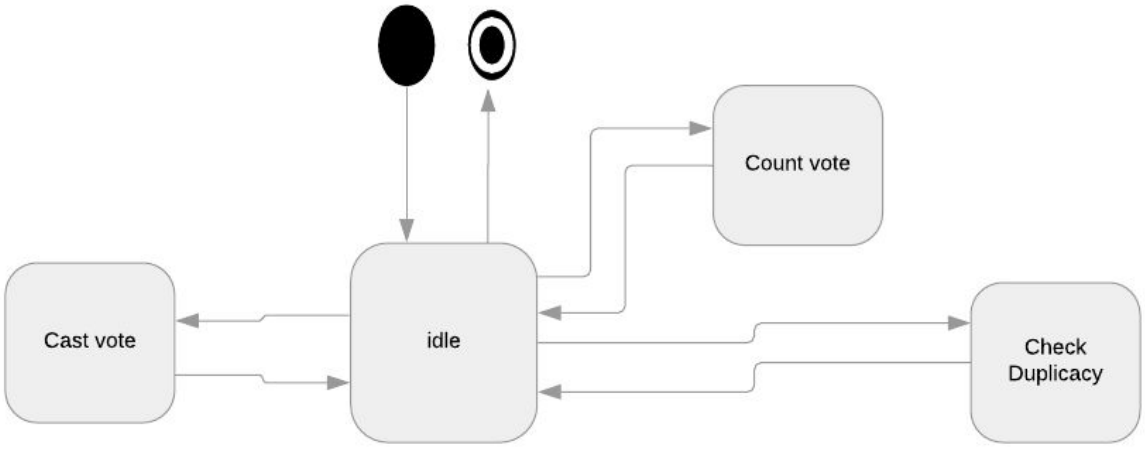
State Diagram:Interface



State Diagram:Notification

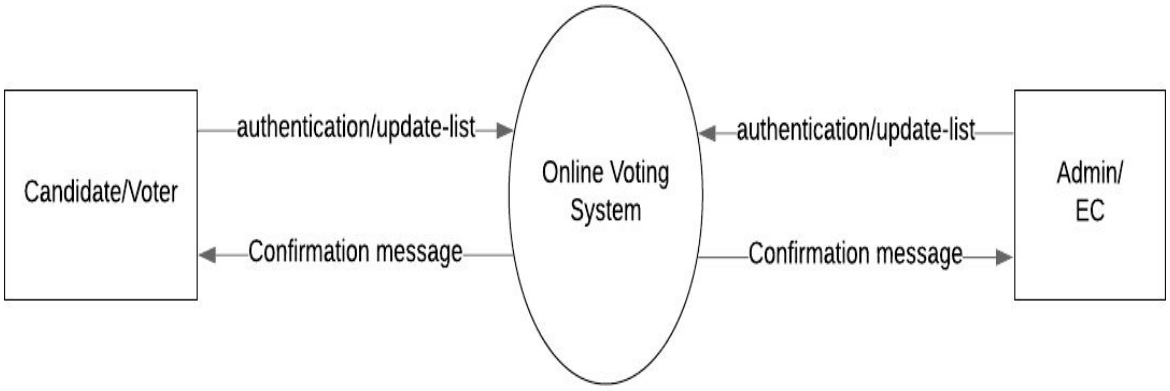


State Diagram:System

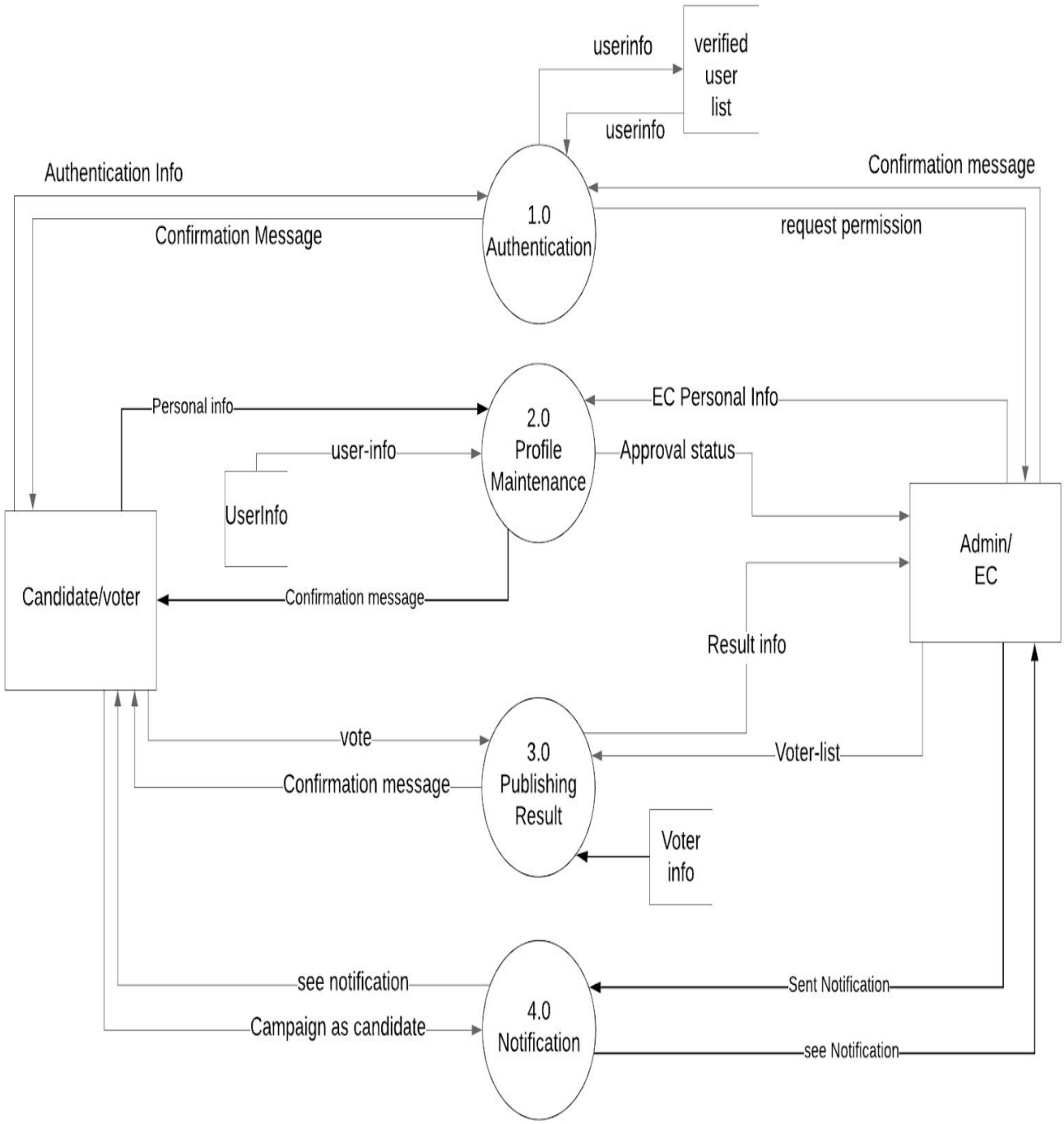


State Diagram:vote

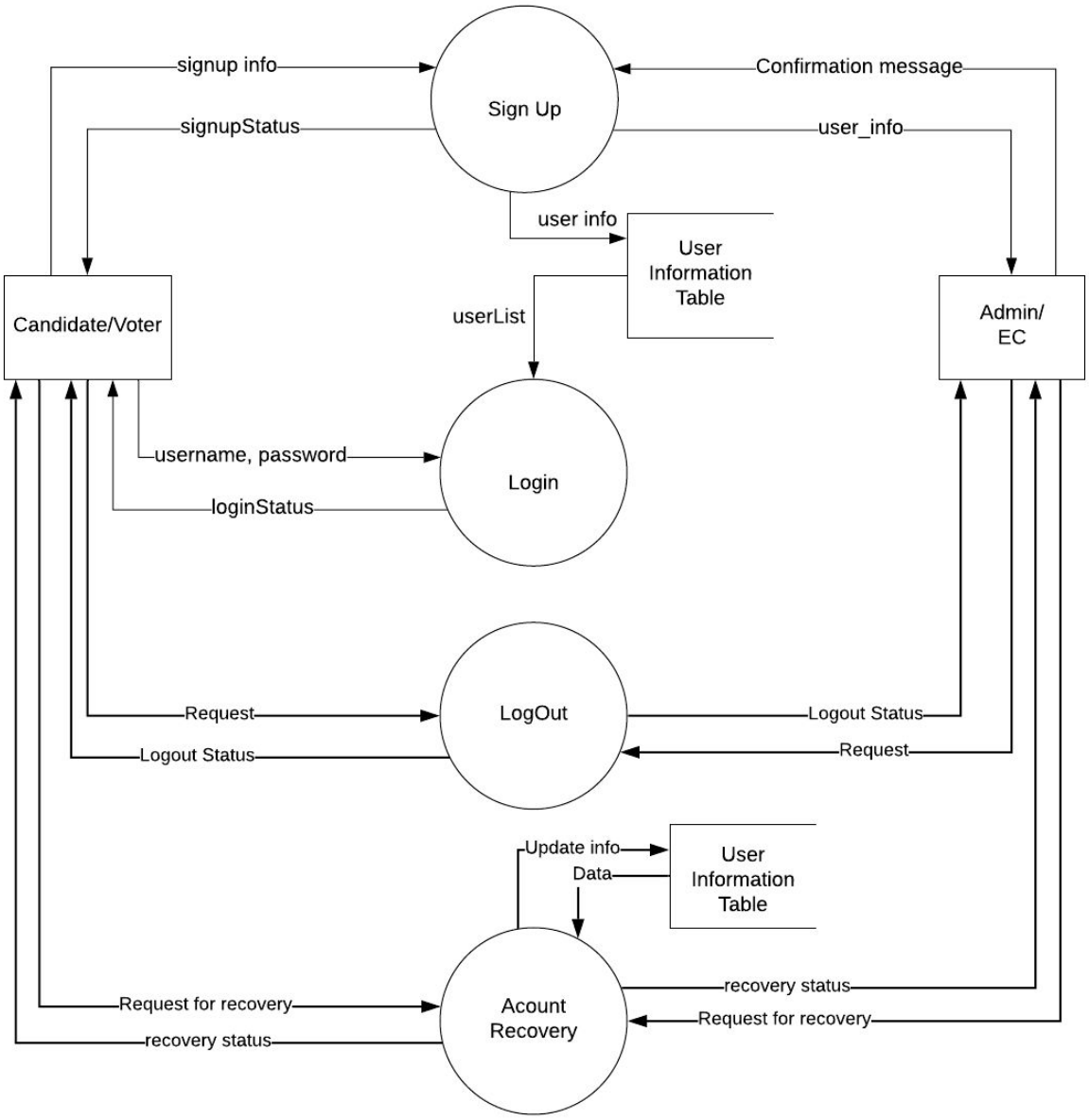
7.2 DATA FLOW DIAGRAM



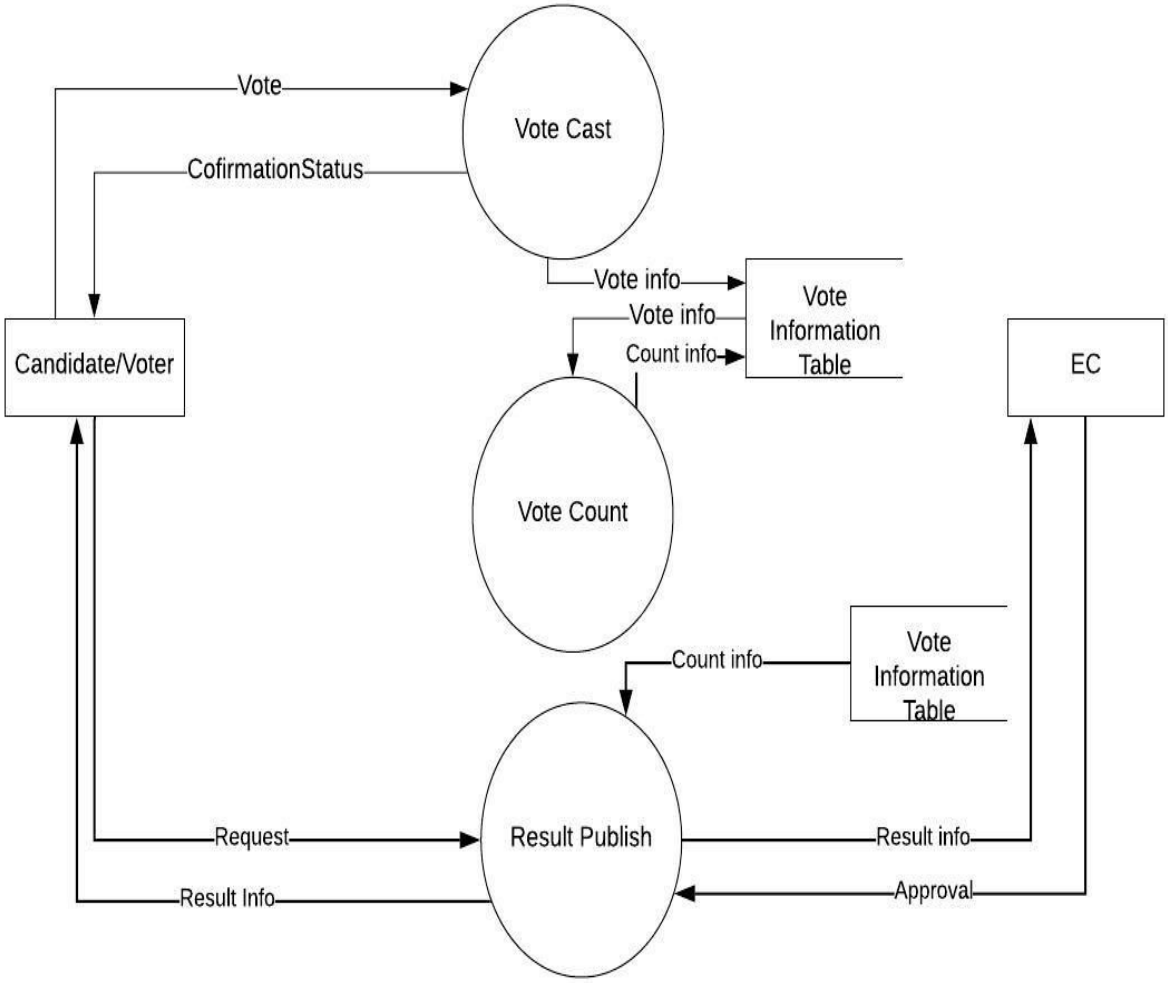
level-0


userinfo
verified user list
userinfo
Authentication Info
Confirmation message
1.0 Authentication
Confirmation Message
request permission
Personal info
EC Personal Info
2.0 Profile Maintenance
user-info
Approval status
UserInfo
Admin/ EC
Candidate/voter
Confirmation message
Result info
vote
3.0 Publishing Result
Voter-list
Confirmation message
Voter info
see notification
Sent Notification
4.0 Notification
Campaign as candidate
see Notification

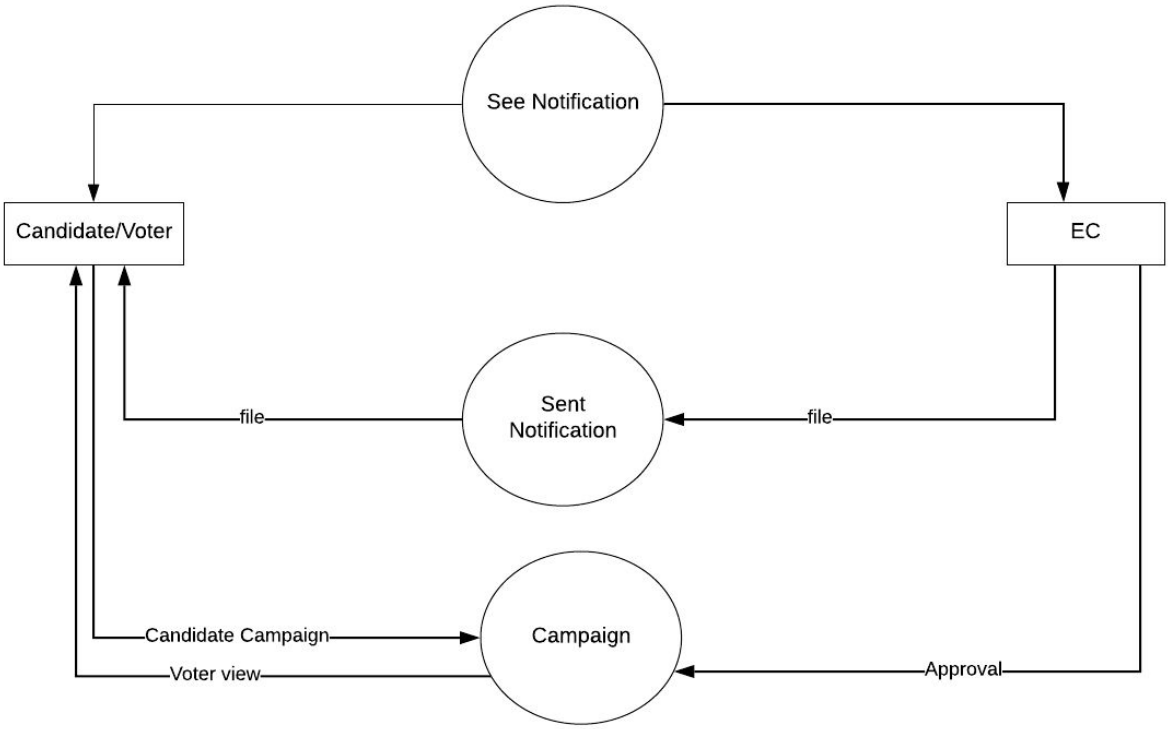
level:1



Authentication

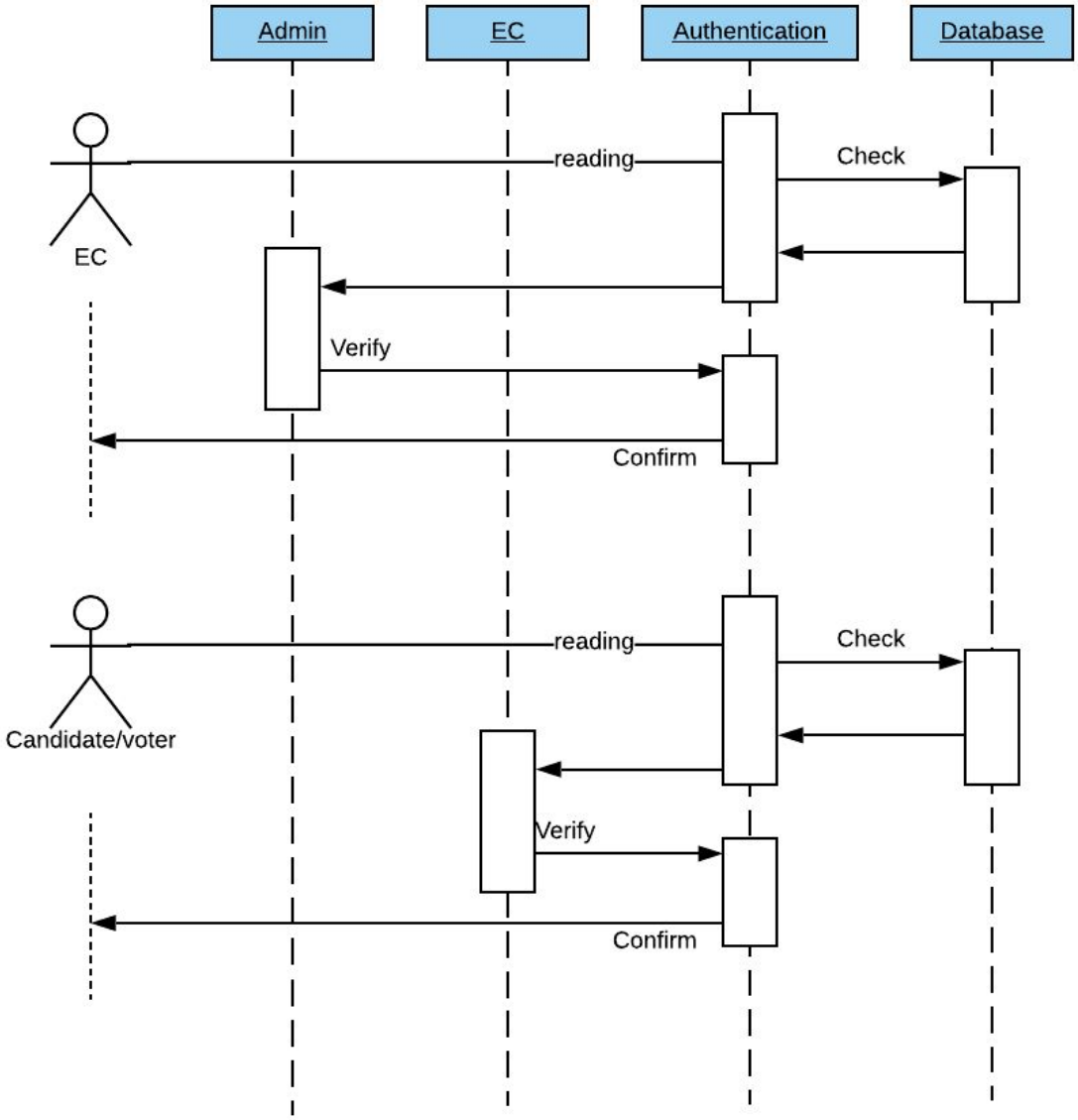


Result Publish

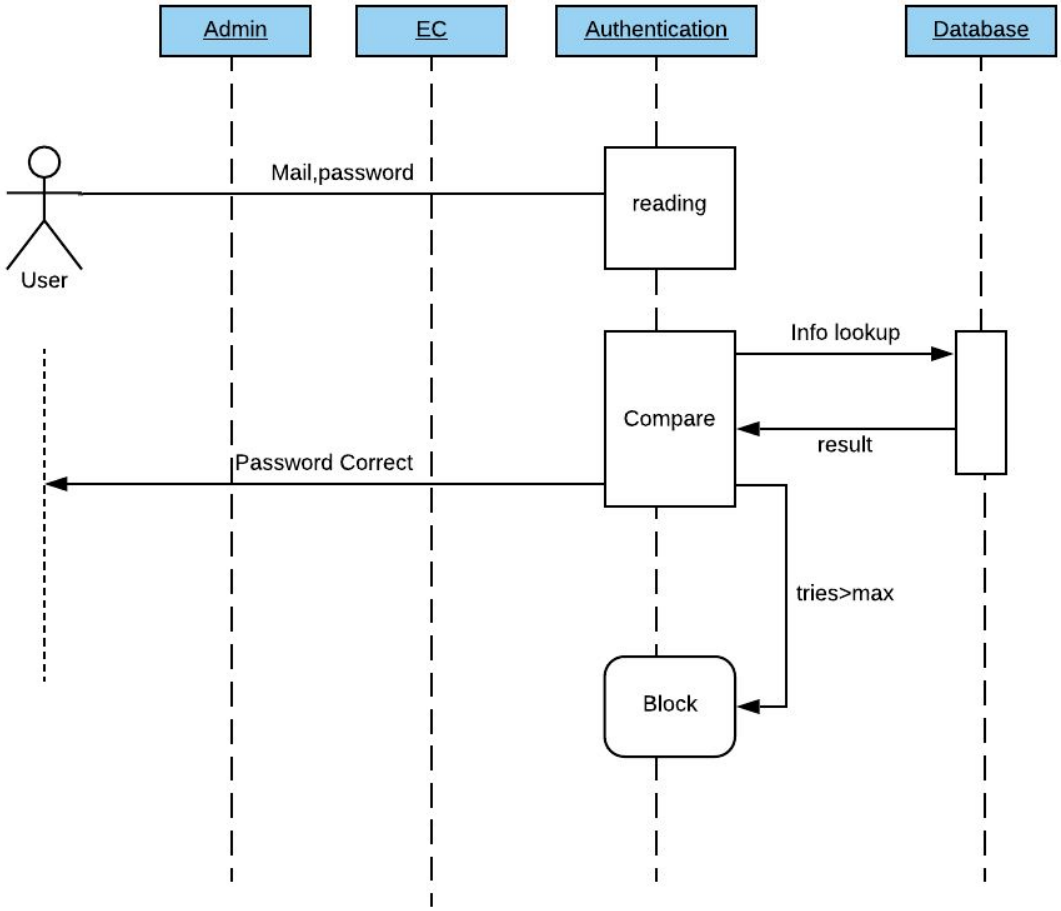


Notification

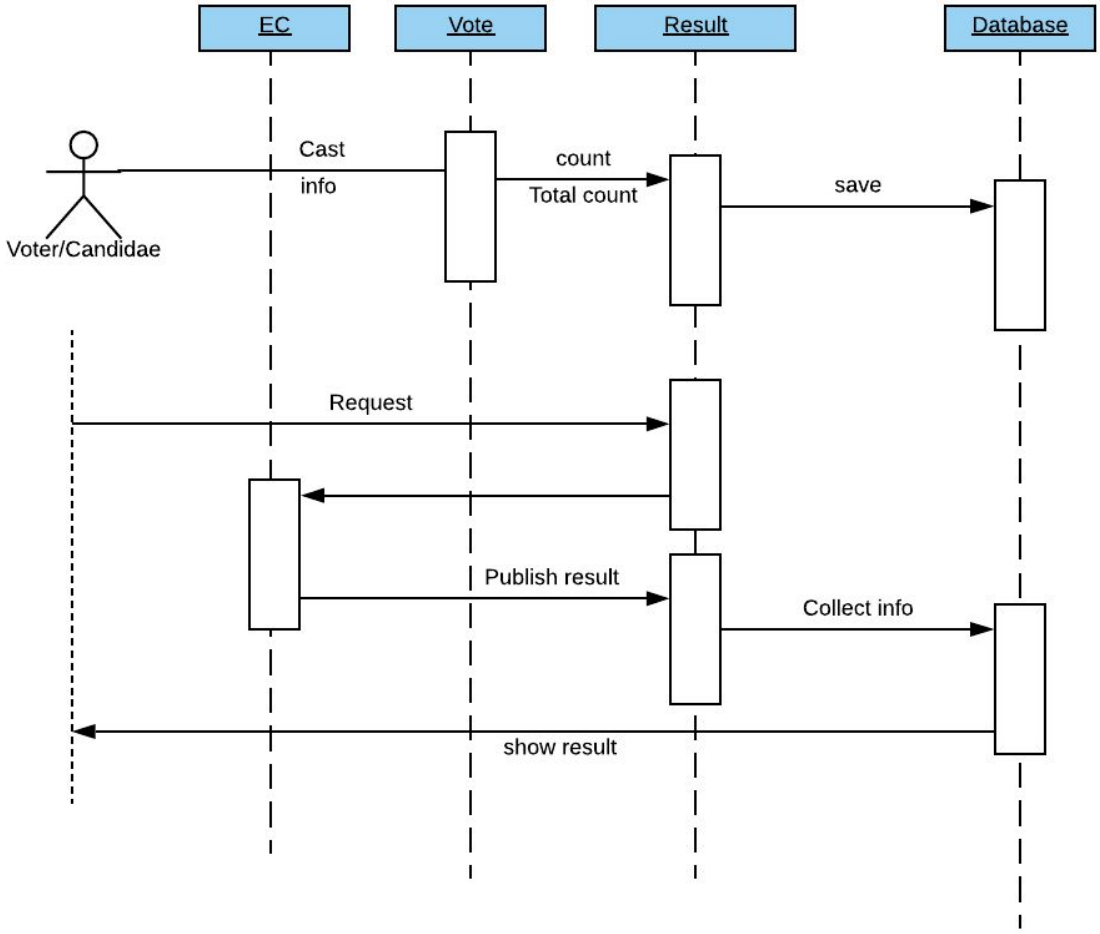
7.3 SEQUENCE DIAGRAM



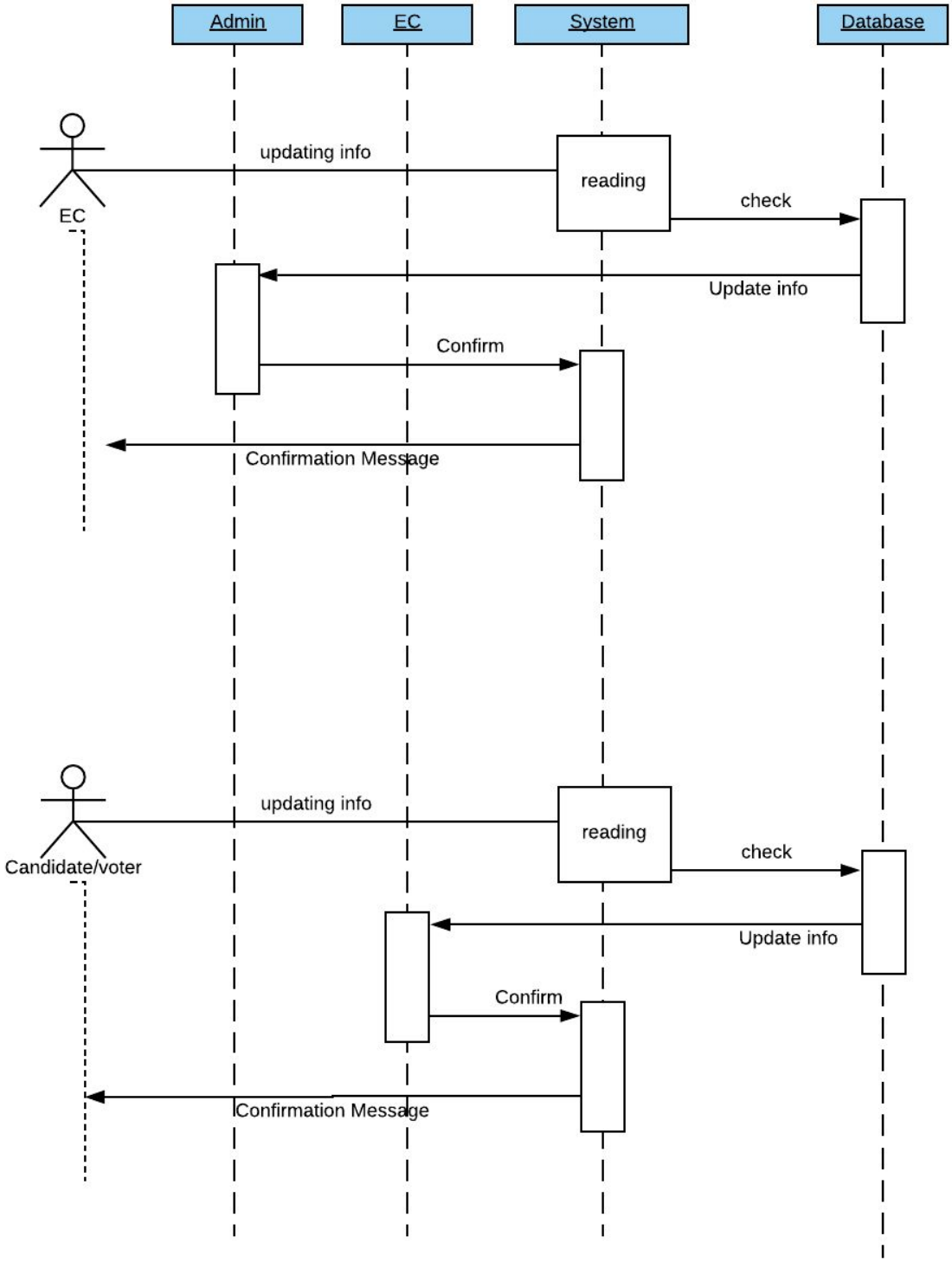
sign up


Admin
EC
Authentication
Database
User
Mail,password
reading
Info lookup
Compare
result
Password Correct
tries>max
Block

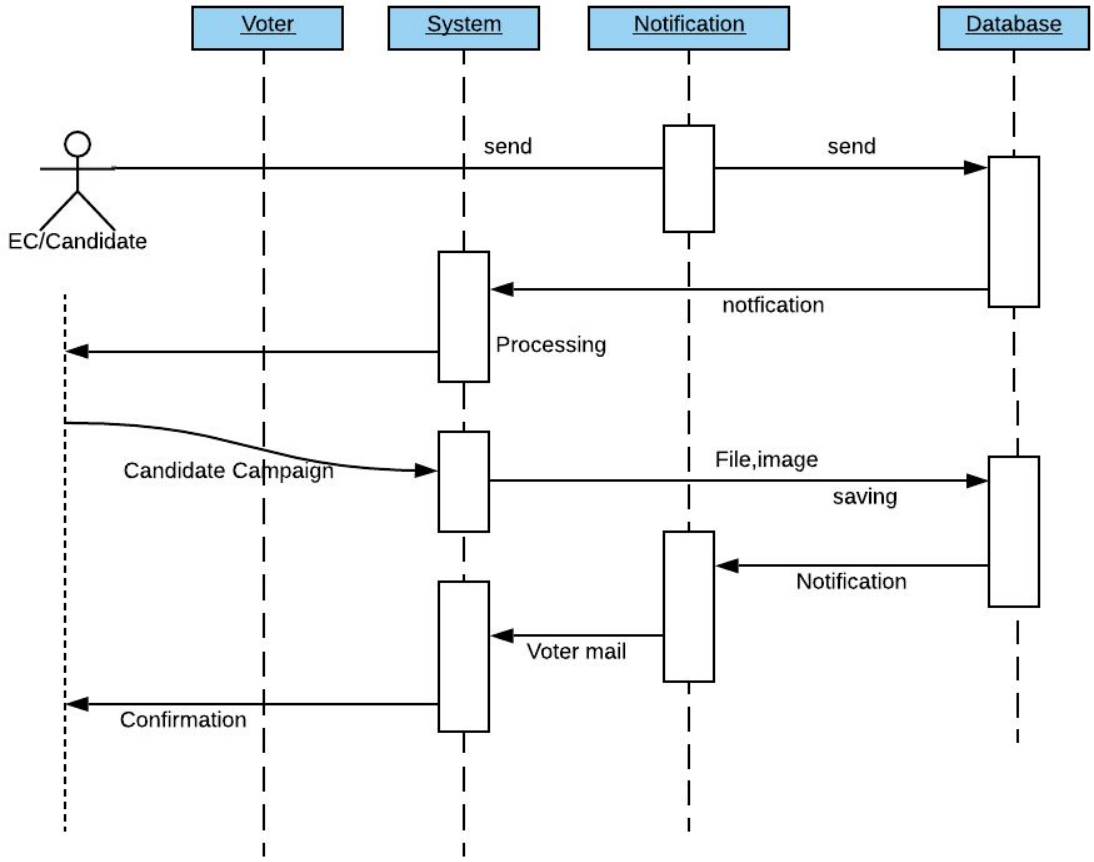
sign in



Publishing Result



Profile Maintenance



Notification