DEPARTMENT OF COMPUTER SCIENCE,

FACULTY OF ARTS AND SCIENCE,

American University of Science and Technology,

Beirut, Ashrfieh, Lebanon

CSI 308: Report

LOUAY SAFWAN KHADDAJ

MAZEN JAMAL ABOU HASSAN

A DOCUMENTATION OF MY CSI 308 PROJECT REPORT TITLED:

"FINGERPRINT VOTING APPLICATION"

JANUARY 2021

TABLE OF CONTENT

ABSTRACT

The word “vote” means to choose from a list, to elect or to determine. The main goal of voting is to come up with leaders of the people’s choice. Most countries have problems when it comes to voting. Some of the problems involved include: ridging votes during election, insecure or inaccessible polling stations, inadequate polling materials and also inexperienced personnel. This online voting/polling system seeks to address the above issues.

In this application a voter can poll his vote easily. In the database server all voters’ information was stored to register in this system, the voter should fill a registration form with the help of a user id and password given by the admin. This information will be checked by the database server of the government.

This system is helpful to the voters for it decreases the time of voting process and it is more secure due to the fact that fingerprint is an important identity of the user. Further, Fingerprint Voting System is user-friendly, has a simple architecture, reduces the polling time and provide easy and accurate counting without any troubles.

It should be noted that with this system in place, the users, citizens in this case shall be given ample time during the voting period. They shall also be trained on how to vote online before the election time.

CHAPTER ONE

INTRODUCTION

1.1 –BACKGROUND OF THE STUDY

To begin with, Fingerprint voting system is a web based application due to the fact everything is becoming web oriented nowadays. By web oriented, we mean that you can access and manipulate the application. In addition to that, with web comes all the factors such as speed and security.

This web driven application paved the way to online voting which is an electronic way of choosing leaders. This machine uses human biometric system where it reduced the polling time from paper voting system. According to the traditional system, where it operates by issuing paper ballots to eligible voters who present themselves at the polling unit on the day of the election, votes could be counted manually so that there is more opportunity for occurring error, such a duplicates counting and completely missed counting. Sometimes votes were even manipulated and motivate by political parties which lead to inaccurate vote and it will distort the results of an election in favor of certain candidates. One of the essential activities in the election process is to count. If the counting failed this could lead affect the people attitude towards the current government so that the election counting should be transparent, accurate and reliable. For achieving such a purpose, this application accomplished security, confidentiality, reliability and accuracy where these are the heart of computerized e-voting system. The election data was recorded, stored and processed as digital information in the government database. Online voting process authentication can be done with fingerprint sensing at the time of voting.

For each state district, valid voters would have their fingerprint name and other infor mation on the government database server. This would also ensure that only valid users can cast their votes with the aid of a specific identification card number and fingerprint scanner.

1.2 – AIM/OBJECTIVES

The aim is to develop an application that seeks to use various stages of security authentication carrying these objectives:

- To create a secured online voting platform where authenticity of votes and voters are ensured with the use of mechanisms such as fingerprints
- To improve Voter's identification since biometric features cannot be shared
- To ease the problem of queuing during voting period on elections

1.3 – ADVANTAGES OF FINGERPRINT BASED VOTING SYSTEM

1.3.1 **Planned approach towards working:** Organized and planned approach will take place in the working organization. All data will be stored in a governmental database in a sound form. This information can be retrieved anytime its requested.

1.3.2 **Accuracy**: A higher level of accuracy is obtained. It makes sure that the operations are done correctly and it ensures that whatever information is coming from the center is accurate.

1.3.3 **Reliability:** The reliability of the proposed system will be high due to the above stated reasons. The reason for the increased reliability of the system is that now there would be proper storage of information.

1.3.4 **No Redundancy:** In the proposed system utmost care would be that no information is repeated anywhere, in storage or otherwise. This would assure economic use of storage space and consistency in the data stored.

1.3.5 **Immediate retrieval of information:** Immediate and efficient retrieval of information is the main objective of such a proposed system.

1.3.6 **Immediate storage of information:** In manual system there are many problems to store the largest amount of information.

1.3.7 **Easy to Operate:** The system should be easy to operate and should be such that it can be developed within a short period of time and fit the budget of government.

1.4 – PRODUCTS OF SOFTWARE

Products of a software can be customized (Bespoke) or generic (COTs). In generic products the specification of what the software should do is owned by the software developer and decisions on software change are made by the developer. While, in customized products the customer owns the specification of what the software should do plus the customers makes decisions on software changes. Customized software is more compatible for such an application because its maintained as long as you require and you can do anything you wish with it. In addition, a bespoke software is a smart investment in the long run. Custom software on the other hand is more secure and harder for hackers to infiltrate. Hackers see no point attacking bespoke software when they can access software shared by multiple companies.

However, we could not forget that such a software costs more than off-the shelf due to amount of time and effort needed to create such a masterpiece. Plus, the waiting time is more because its created from scratch.

1.5 – METHODOLOGY

Rapid Application Development (RAD) is the methodology of choice here; this is because the objectives of RAD include high speed, high quality and lowered cost. RAD emphasizes the use of special techniques and computer tools to speed up analysis, design and implementation phases. Tools include Computer Assisted Software Engineering (CASE) tools for example: blackboard was doing great but with the COVID-19 the volume of market share has increased exponentially from few millions to 100 million.

The RAD methodology goes through the areas associated with the proposed system which include:

- **Mode of user voting:** Through the unique user ID and password provided by the Administrator, users may log 0into the system. The device tests, upon submission of the login request, whether the specific voter is eligible to vote in the particular election and carries out appropriate actions. The device

shows a corresponding message to the voter (staff) if he/she is unable to vote, that he/she is not qualified to vote.

- **Mode of voter authentication:** By checking the unique identification information they have through their fingerprint; voters will be authenticated. This fulfills our mission of ensuring a safe method of validating voters before allowing them to cast their votes.
- **Mode of data collection and verification:** There is a need for pre-recorded data in the memory, i.e. a database of the system, before voters can be checked. Consequently, in order to allow verification, the data of any prospective voter (staff) needs to be collected.
- **Mode of data communication:** The proposed system would operate on an internet. This allows all voters to have access regardless of location as well as increase their ability to use a range of devices to cast their votes.

1.6 – PROJECT OUTLINE

The work is divided into six Chapters:

Chapter 1- It gives an introduction of the report where it explains the case description from the aim to products and advantages of such an application.

Chapter 2- It talks about software process Model from agile methods combined with plan-driven to Scrum process to the importance of including minimal documentation.

Chapter 3- It tackles the requirements specification and the

Chapter 4- it shows the system modeling and drawings of diagrams from activity to use case and sequence diagrams.

Chapter 5- System testing

Chapter 6- Conclusion of the entire report and Future Work

1.7 –CHAPTER SUMMARY

This chapter mainly focuses about the introduction of the report from tackling the aims and objectives of the application and advantages of it. This chapter also include the methodology being used during the development process as well as the products of the software.

CHAPTER TWO
SOFTWARE PROCESS MODEL

2.1-DEFINITION

- Software process: its steps needed to take a software from requirements to full implementation. In addition, a process is steps you're going to develop or take a certain idea from specification to the final product. We shouldn't be in a situation where we should reverse engineer the system. And by reverse engineer the system we mean backtracking, reading the code of the application and reading the initial documentation to reveal its design, architecture and code. It is to understand, to maintain and do extra reading and to embed new requirements and update the application.
- A process improvement should be a mix of 2(waterfall and incremental) to be able to satisfy quality requirements or end user requirements through an agile approach. When using agile we should understand the existing model and change it to increase the product quality thus, reducing cost of development.
- Agile approach is starting with demanded feature and add on the top of what exists until the product is complete. In agile methods, it focuses on code rather than the design and are intended to deliver a working software quickly and evolve quickly to meet changing requirement and they are based on an iterative approach.
- Moreover, agile is using any kind of process development to develop the software even if I'm short in time. For example, if you are short in time and a certain requirement was given to you to implement you shouldn't be waiting till the end of the project you can do it any time. The backlog will change dynamically and if it's an urgent change you'll see it in the next sprint. Unlike plan-driven that function according to a predefined plan.

2.2-INCREMENTAL MODEL

We cannot just use the waterfall model or the incremental model alone it should be a mix of both. Every sprint can be a plan-driven sprint and every waterfall is responsible for a release, but the overall process of different sprints is an agile process. When requesting this type of application, the client in this case the government, requests to see working increments. When you use waterfall the main drawback of the waterfall is the difficulty of accommodating change after the process is underway. In principle, a phase has to be complete before moving onto the next phase. In such a case if the client doesn't like your application you should restart the work and do it all again. To avoid such problems, you use the incremental model embedding in each sprint the waterfall model.

Furthermore, you should take into consideration that you don't have the whole time in the world to develop the application. Plus, documentation in waterfall may take a lot of time that you don't have. On the contrary, the customer needs his application in a short time and this is why the waterfall may not be the perfect fit.

While using the incremental model you start with the most important, you prioritize things. This creates satisfaction from an end user perspective, where most commonly used features are done first, the customer has something in hand. So, it's easier to get customer feedback on the development work that has been done.

2.3-ATTRIBUTES OF THE SOFTWARE

Maintainable Software: The software can be modified and ready to cope with change according to the client needs. As a software engineer you should fix it if there are any bugs, add business requirements so that the output of application will remain according to the clients' expectation (System is up and running).

Dependability and security: Improve voter's identification since it biometric features cannot be shared.

Efficiency: Optimized software this means minimum hardware specifications and resources, no need to change the OS or the server. Also, less loops you create, the better code

thus it can run under minimum set of specifications. With an electronic voting system, results of an election can be counted and displayed immediately. And it is particularly useful in the event that there is a very large amount of votes to be counted.

Acceptability: User-friendly, economic and easy to manipulate.

2.4-IMPORTANCE OF ADOPTING TO A MODEL

Maintenance is much longer than developing the software, it's better to pay at the beginning and a adopt to a certain process model and it would agile in this case rather than developing the software quickly without going through process models so you are stuck at the maintenance and evolution of the system. This will cost you more since you're obligated to do the work again thus, it will become expensive and not practical. Yet, the majority of software costs in large companies, come from maintaining their existing software systems.

2.5-CHAPTER SUMMARY:

This chapter summarizes the definition of the two models and generally overviews that it should be a mix of both incremental and waterfall processes plus major attributes of the software which include security, efficiency and maintainability.

CHAPTER THREE
REQUIREMENTS SPECIFICATIONS

3.1-AGILE PRINCIPLES

Customer Involvement: Customer and in this case the government should be involved permanently unlike plan-driven, the government is involved at the end, it should here test the model, chunk by chunk what the programmer is releasing. The company will have to designate a representative of the end user of the application where he will directly be in touch with the government.

Incremental delivery: The software is developed in increments with the customer specifying the requirements to be included in each increment.

People not process: Check whether the government is satisfied in the work. People here will change their mind at any time.

Embrace change: Change is always a factor while developing such an application, always there should a way to embed these changes and embrace them.

Maintain simplicity: With a simple code embedded in the application, you will be able to track, trace errors or bugs found in the code.

3.2-SCRUM PROCESS

- Scrum is a typical agile method it manages iterative development and it's a loop you keep iterating until you're done it consists of three phases:

- **Initial phase:** you will be planning all the phases where they are interleaved through increments and releases to be added to the initial product later on. Before in Plan-driven phases were specification, coding, maintenance, testing and implementation. Here every waterfall is responsible for a release where releases can be added to initial products until the final product is obtained.

- **Second phase:** is a sprint cycle that could correspond to 1 or many tasks that we just broke down.
- **Third phase:** project closure where you complete required documentation that consists of embedding comments in your code that explain what you've accomplished.
  Taking into consideration, when implementing agile, minimal documentation should be included as a hard copy. If it wasn't part of the code, many issues will be faced in the future, so there will be no trace of underlying logic when it comes to maintaining the code.

Domain Requirements:

The general requirements of the system must be taken from the customer which is in this case the government and those requirements can be as the following.

The System must have the hardware compatible with the software for fingerprint recognition , and the first steps are the fingerprints being captured from the voters (the data of the features of the fingerprints) which will be part of their sign up process (which will also be compared with the government fingerprint database to check if the voter is registered as a citizen in the country or region he is voting in), and store them in a secure database for the system to use to verify if the voter is eligible to vote or authorized to vote by just comparing his fingerprint to the fingerprints in the database. And after all the authorization is done he will be able to vote to whoever he wants, and every candidate for the position in speak will have a brief description about his campaign, objectives and his accomplishments which will help undecided voters. Additionally, there will be tracking of the votes to check if everything is going as planned and no complications are happening and nobody will know who voted for who etc.

Functional Requirements:

Functional requirements are the requirements that show the potential or the capabilities of the system. In the case of the Fingerprint Voting Application the functional requirements are as following:

- The System must explain and give information accurately and clearly to help the voter in decision making of who to vote for.

- Each change to the system from the governmental side, developers side or from the people responsible of adding information to the system through their accounts on the system must be recorded and verified completely to make sure that no sabotage or cheating can take place in these elections.
- The System should always check for bugs and errors and be able to send feedback giving the error code, and the voter can also give feedback to every error he encounters in case any happen.
- The system must be user friendly and easy to use.
- The system must have a pop up message that asks the user to confirm his decision and that is for the probability of users pressing the wrong button or choosing the wrong candidate by mistake.
- Only authorized voters, who are registered, should be able to vote, no voter should be able to vote more than once and no voter must be able to choose more than one candidate.
- The data must be accurate (i.e. the votes should be counted and updated) this means the system must provide data integrity checks to ensure data remains consistent and updated.

Non-Functional Requirements:

Non-Functional Requirements are requirements that are limited to the quality of the Functional Requirements and to the services or functions offered by the system:

- The voting process must not take more than one day.
- Mobile phones or Computers or any device with fingerprint recognition can install and use the application.
- The efficiency of the System must be at its peak performance during the voting process from beginning to the end.
- The system must adapt and agree with the government requirements.
- Backup data and the option to restore should be available at all times.

Security Requirements:

- The votes that the system gets should not be associated with the identity of the voter.
- The system must be protected against any DOS attacks and must be available for anyone that can use it.
- Every operation done on the system must be tracked and recorded.
- Accurate time and date settings must be provided by the system.
- Unauthorized voters must not be able to vote or even access the system.
- The scheme should not allow unauthorized users to download votes to decide how voters voted.
- The mechanism should provide a way to safeguard and secure recounts of ballots cast in elections.

CHAPTER FOUR
ARCHITECTURAL DESIGN

Architectural Design is the architecture of the application, its closer to implementation and how you are going to implement, what architecture to use and how the app is subdivided into chunks models and functions. Architecture should be optimized and the structuring of the application must be based on performance, security and functionality.

The best architectural design of such a system is using Model View Controller(MVC) embedding in it layered architecture.

The MVC is one of the most commonly used patterns that are used with web applications, it is used when there are multiple ways to view and interact with data. Also applied when the future requirements for interaction and presentation of data are unknown such in our case of voting application. It consists of three logical components:

Model: manages system data and associated operations on that data.

View: how the voter can view the application and how it is presented to the user.

Controller: it controls the input and the output, and what you're allowed to do in terms of data entry and data access.

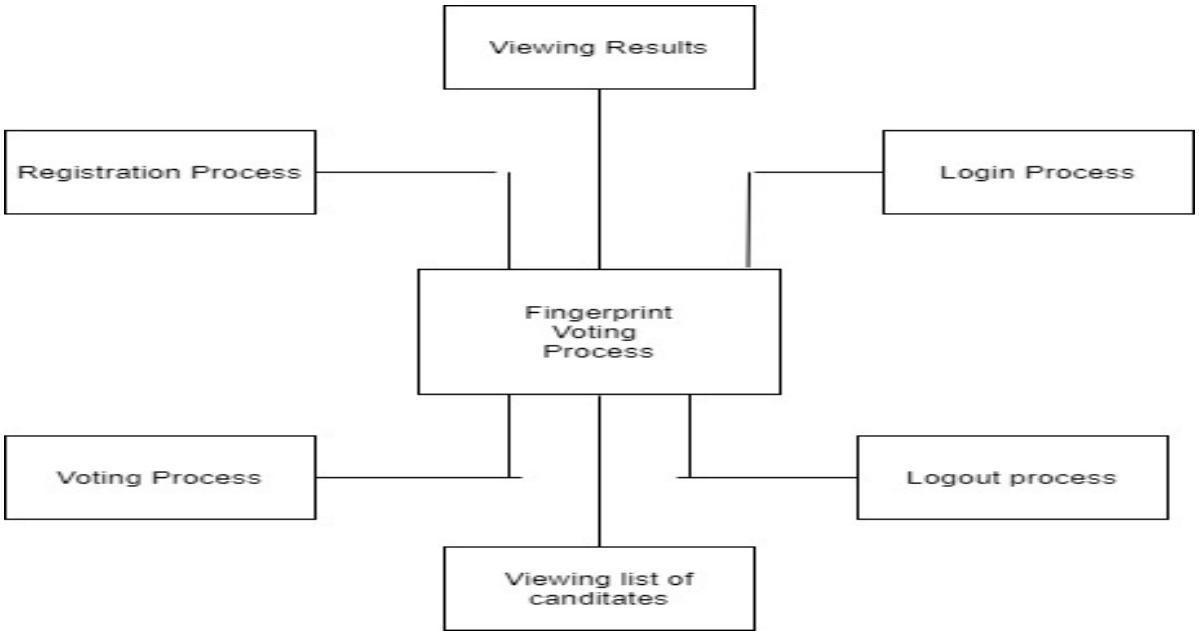

Figure 1: Context Diagram:

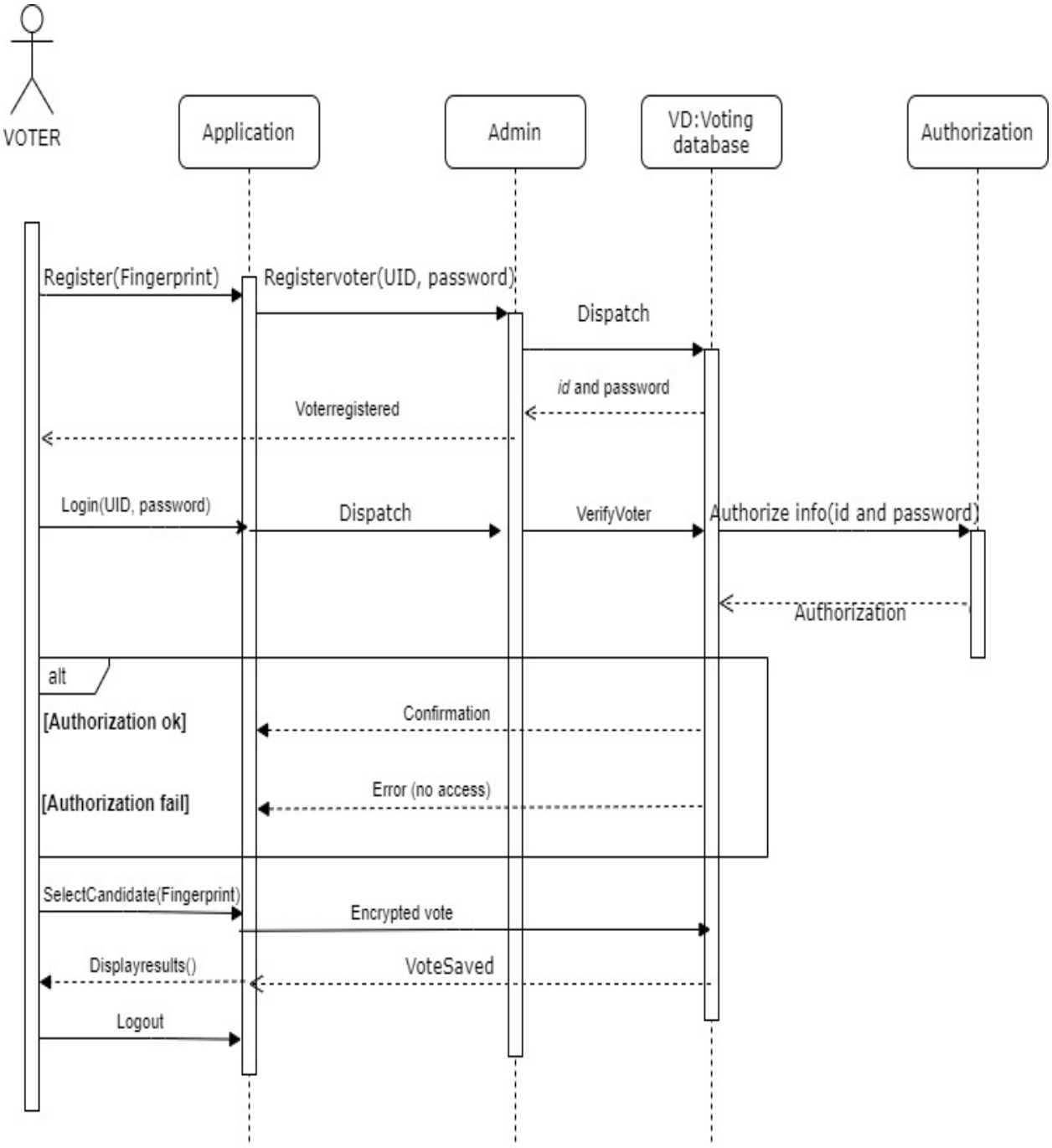

Figure 2: Sequence diagram

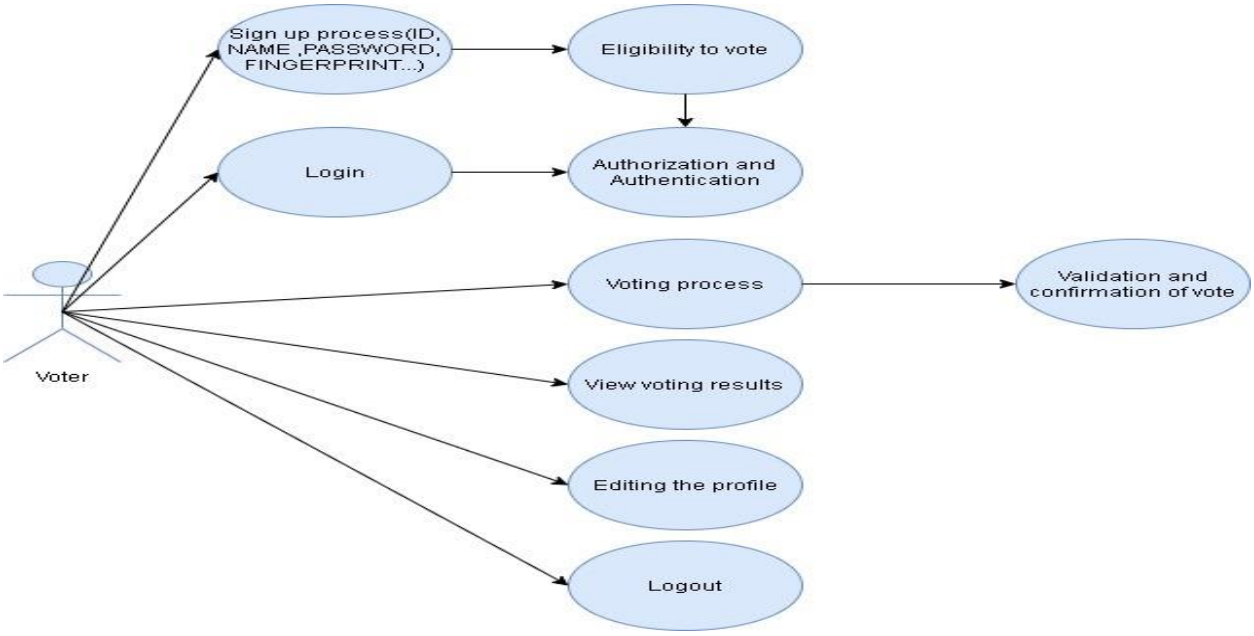

Figure 3:Use case Diagram For Voter

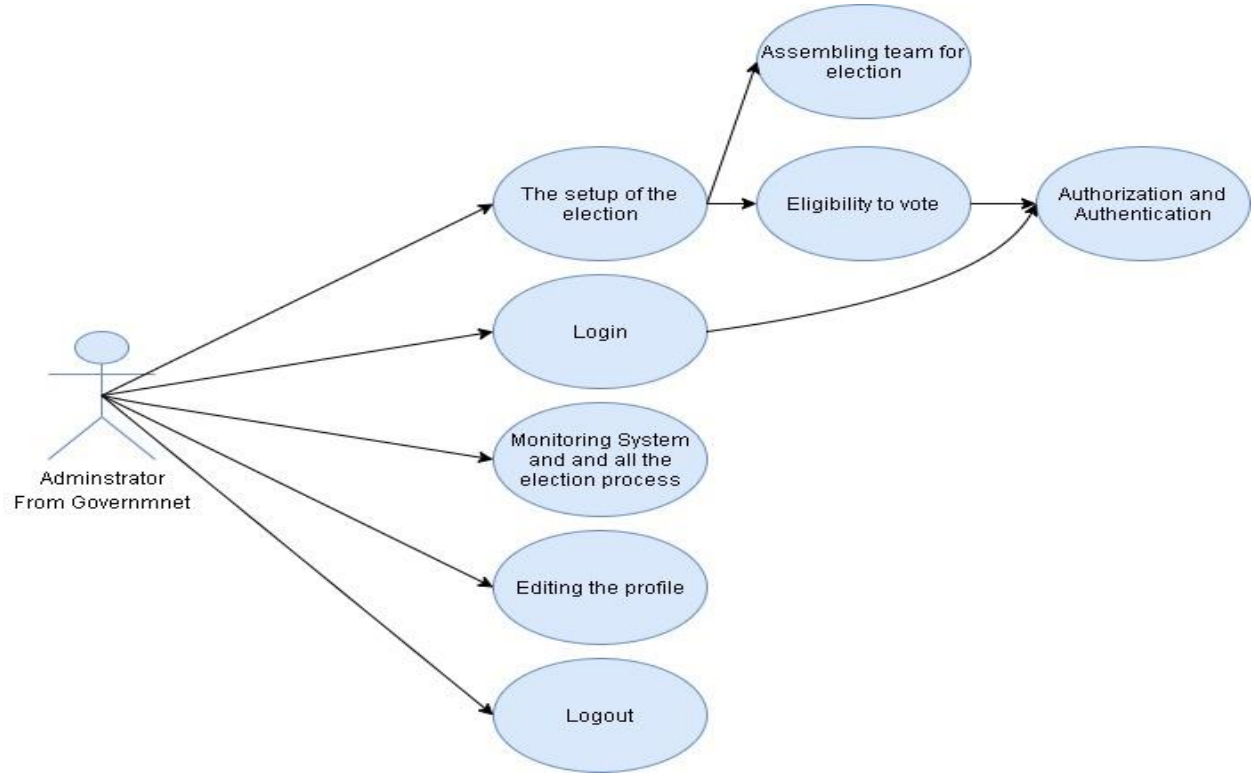

Figure 4- Use Case Diagram for Admin

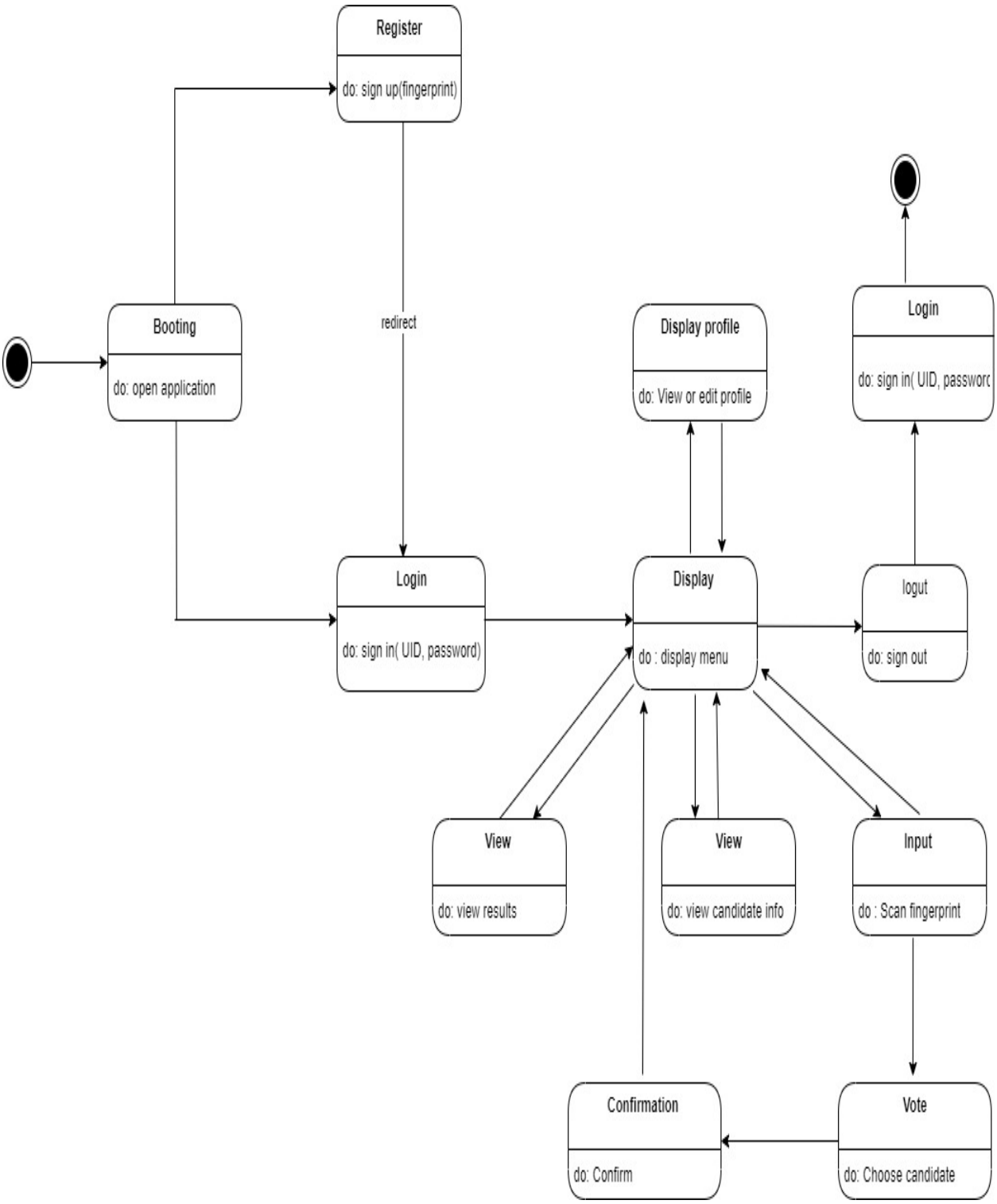

Figure 5: State Diagram

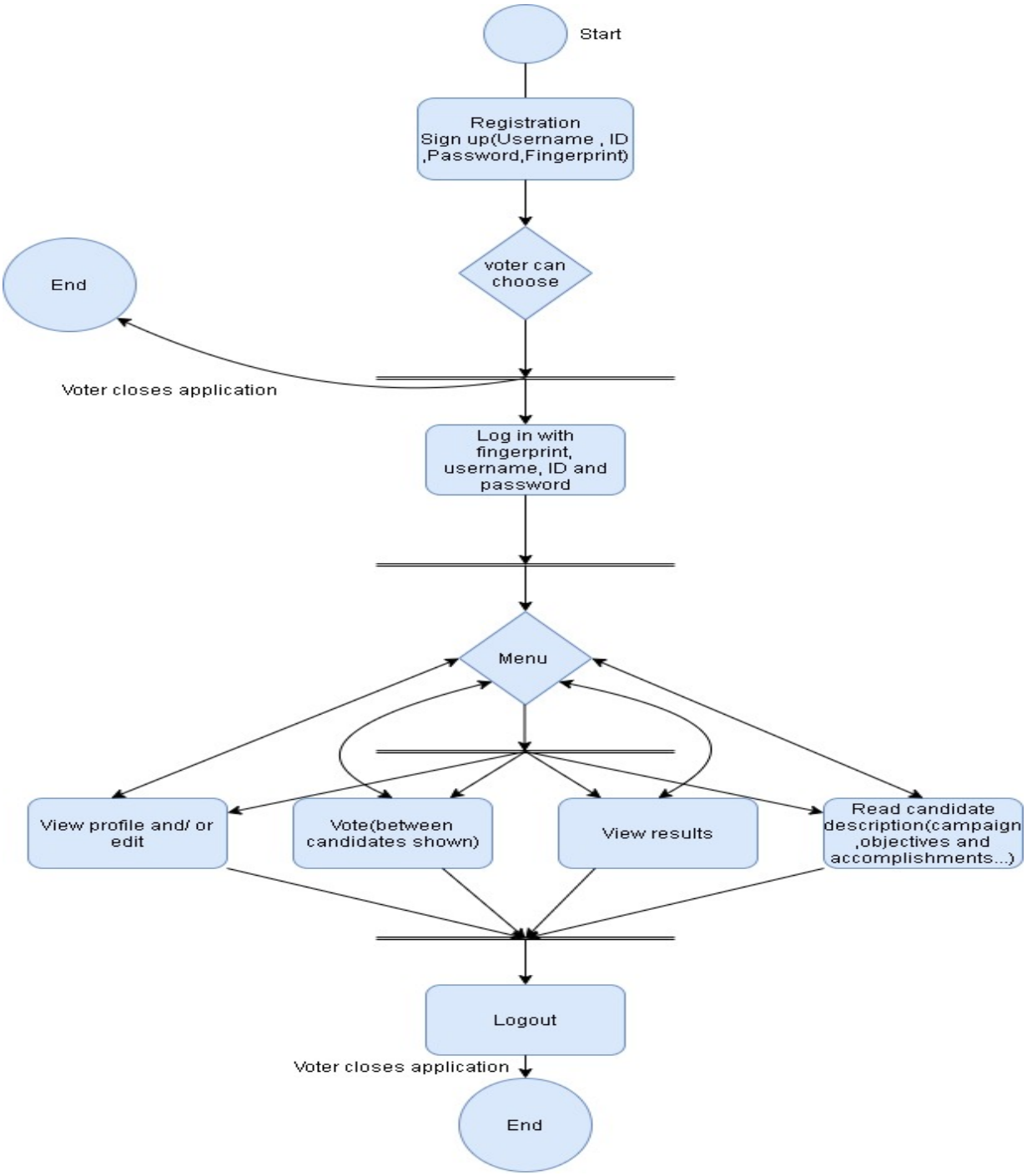

Figure 6: Activity Diagram

CHAPTER FIVE

System Testing

System testing is one of the most important parts of making a system since it checks if there is any errors or bugs that can cause failure in the system in any way, which is very important because if there was any error or bug the developers will find out and will solve them immediately to create a system that is of high quality. For example: as a development company debugging the program will help in finding solutions for issues that the system encounters after receiving the feedback from the users or while making the system parts.

Some techniques of system testing are:

Integration Testing: Which is the testing that makes sure that the communication of the different parts of the system is working properly where the inputs and the outputs of the system are transported are being used efficiently for the best quality of the functions of the system to be achieved.

Unit Testing: Which is the testing of the different parts of the system separately to check if each part is working correctly and doing what it is supposed to do efficiently according to its function in the system.

User Testing: Which is the testing of the system as a whole after the system is implemented and the testing is done by people from the company or real users to check if there are any more bugs or if the system is missing any important details, and that is done when the application is still in its beta version. Additionally, in this testing the system is checked to see if it is up to the quality standards of the customer.

Chapter Six
Conclusion

This project tackled the process of developing a fingerprint voting application where the software process model and the requirements needed for such an application whether it was functional, non- functional or security requirements were presented. Moreover, diagrams from sequence to context to state diagrams were also presented and how to maintain and debug the system. The future of voting and the election process will be bright using these types of applications since voting will become more efficient, more secure, free of fraud, economic and trustworthy. It allows the voters to vote from any device at any time during the election day. The application will have 24/7 maintenance and will be subjected to evolution over time. By using such types of applications, the percentage of voting will increase, the problem of queuing will be eased and the cost and time of voting will also be decreased.

References:

SOMMERVILLE. Software Engineering, 10th Edition, Pearson, 2016.

https://www.slideshare.net/nitinbhasin3/online-voting-system-project-file

https://www.slideshare.net/subhashisdas376/online-voting-system-project-51189399

https://www.cs.jhu.edu/~rubin/courses/sp03/groupreports/group4/group4_requirements.pdf