Mobile Voting Application for Local, State, and National Elections

By: Alec Metzger, Adrian Mitchell, and Cory Jackson

iVote

Submitted to
the Faculty of the School of Information Technology
in Partial Fulfillment of the Requirements for
the Degree of Bachelors of Science
in Information Technology

Alec Metzger _____ April 11, 2017

Adrian Mitchell _____ April 11, 2017

Cory Jackson _____ April 11, 2017

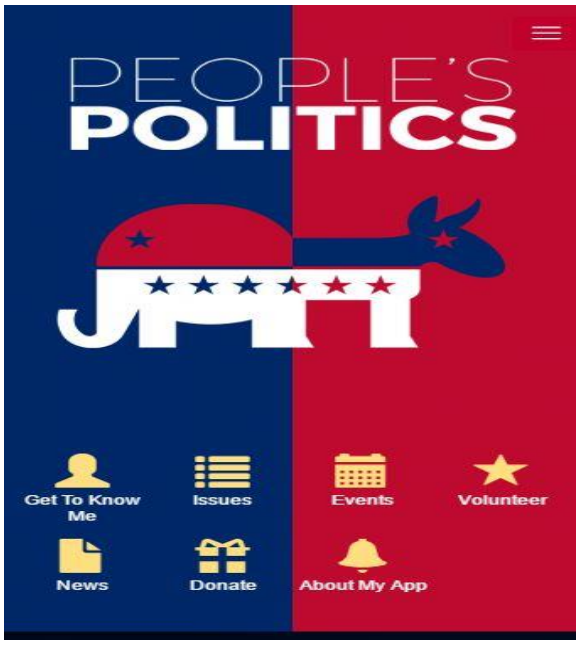
Jim Scott & Bogdan Vykhovanyuk _____ April 11, 2017
Insert Faculty Advisor's Name Here

University of Cincinnati
College of
Education, Criminal Justice, and Human Services
April 11, 2017

Mobile Voting Application

for

iVote



Contents

Tables

Abstract

Every election many American's do not vote for the government officials due to their location, time, constraints, and lack of interest and/or disabilities. We have created and designed an application that will allow American citizens to vote via their smartphone. iVote changes the way people vote, access political information, and communicate with other constituents. Not only does our application allow voting to be more convenient, but also more secure. Our application is using Blockchain technology to ensure all users their votes are being processed securely, and uses a pdf127 scan to verify their identity. Every American has the right to vote, and our application will allow more Americans to exercise that right.

Problem Statement

1.1 Introduction

Every four years all American Adult Citizens (18 & up) are eligible to cast a vote on their personal preference on who they believe should be the President of the United States. The candidate that receives the most votes becomes the President of the United States for the next four years, which is equivalent to one full-term. Presidents are allowed to serve two full-terms at most, but they have to be re-elected for the second term.

1.2 Project Description

We will be designing an application that will allow all American Citizens to cast their Presidential Vote via their smartphone with the security of Blockchain technology built into the application. Our application will not only make it more convenient for citizens to vote, but also more secure. We are aware that some American citizens are unable to vote because of location, time constraints, and disabilities. With the creation of this application we hope to eliminate these problems and give each user a more convenient tool to enable them to cast their vote for the Presidential Election.

1.3 Problem

The problem we are having with the Presidential Election is that more people are voting for American Idol and other TV shows because they allow their viewers to vote via their mobile device. This creates a more convenient way for their viewers to vote, because they do not have to physically go to a certain location, but instead can vote straight from their mobile device. This method is also a lot less time consuming for the user and enables them to continue their daily routine without any major

adjustments. With the creation of our application it will solve the problem the United States has with voting when it comes to the Presidential Election, because American citizens will be able to cast their votes from their mobile devices. This will allow voting to be more convenient and less time consuming, while also being more secure.

1.4 User Profile

1.4.1 Application Title

iVote

1.4.2 Potential Users

Students (18 years or older)
Adults
System Administrators

1.4.3 Software and Interface Experience

Students and adults understand the basics about using mobile applications with either Android or iOS devices. The System Administrator will have the skills to consolidate the different user's operating system on the other end and transmit their votes that are being directed to the appropriate location.

1.4.4 Experience with Similar Applications

Students, adults, and the System Administrator will understand the layout and functionality of the application, because our application is very user-friendly. It allows users of all ages to be able to understand the process of

validating their identity and selecting a candidate, which will then be submitted by the users and directed to the correct location by the System Administrator.

1.4.5 Task Experience

Users will be able to vote on who they want to be the President of the United States. They will also be able to view other candidates and learn information about them. Find up-to-date news instantly and also view the election polls.

System administrators will be able to process all of the votes submitted and send them out to their correct location, while they also will be able to authenticate the votes.

1.4.6 Frequency of Use

This application will be used for American Citizens to cast votes for the upcoming Presidential Election and also any future elections. They will also be able to gather up-to-date information about all the candidates and find out more of the general information about them.

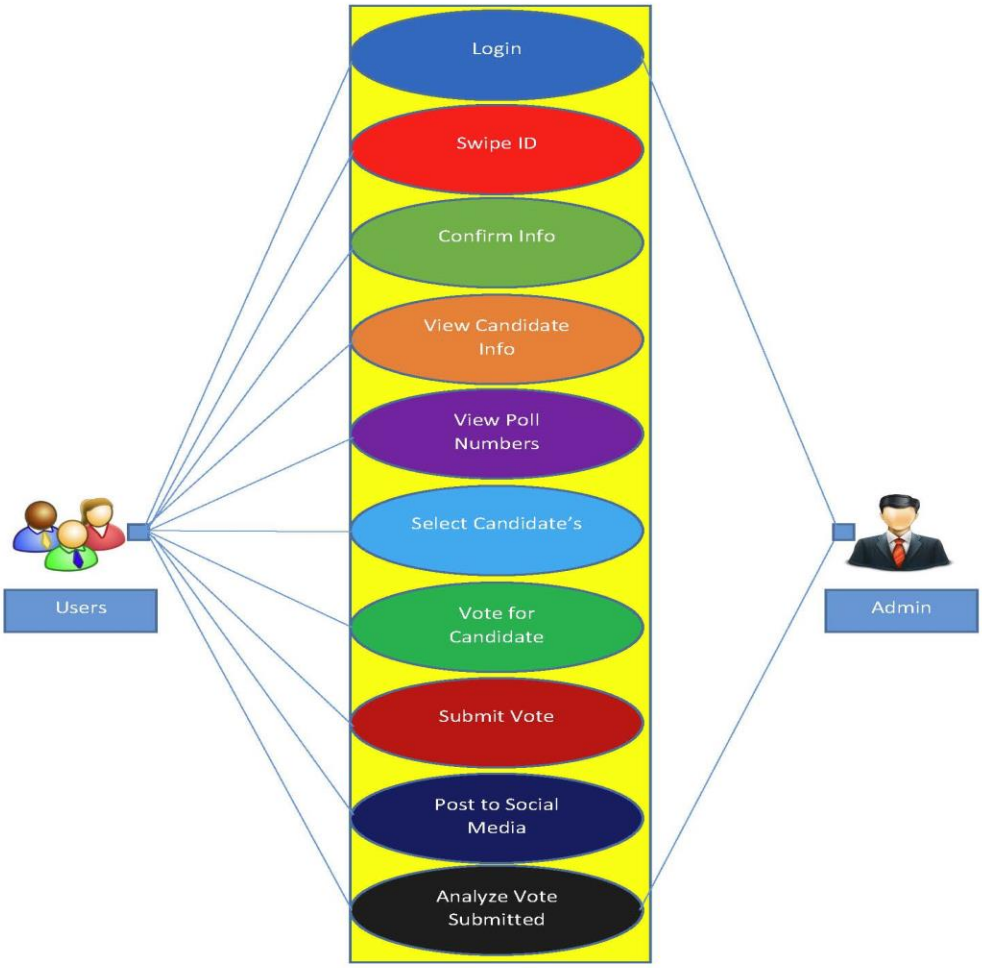
1.4.7 Key Interface Design Requirements that the Profile Suggests

Easy and user-friendly to all users with smartphones. Our application will also be able to be downloaded on computers and tablets, for those who do not have smartphones. This will allow a variety of people to use our application, instead of strictly smartphone owners.

1.5 Use Case Diagram

The use case diagram is designed to show the viewers the different roles each person has, whether they be a user or an administrator. Users create their own profile which they use to log into the application. Once they have logged in to their profile, they will swipe their ID, and then are able to cast their votes. However on the administrative side, the user receives all the votes that are being submitted, which they manage and direct to the correct location.

1.5.1 Figure 1: Mobile Voting Application



1.6 Technical Elements

1.6.1 Mobile Application

iVote will be accessed and downloaded through a mobile application store. There will be several informational links and one secure link. Data will be scanned in by the phone camera. The camera will use the 2D barcode scanner to upload license information.

1.6.2 Database

Uploaded data from the cell phone will be checked against voter information stored in the database. If data from the cell phone is correct, the user will be able to vote. The database will also check the address to assure that the correct ballot is being downloaded.

1.6.3 Blockchain

To ensure that the user vote is secure, Blockchain will be used to create a ledger of the vote and make the vote secure. This will create an electronic “paper” trail of the vote and make hacking into the voter process very difficult. Blockchain in this instance is used from a template, rather than creating a custom chain. In a circumstance where iVote were to become used on a wide level, such as throughout a whole state or nation. A custom built Blockchain would be necessary to take on a larger load of transactions. Provided below, is the template in which Blockchain is used in with iVote:

https://github.com/Azure/azure-quickstart-templates/tree/master/ethereum-consortium-blockchain-network

1.7 Objectives/Deliverables

Objectives and deliverables are the main points of our project that our group uses to help plan and implement each task. It's a list that we created that will help each group member stay on track and understand the need for each object. In **Table 1: Objectives/Deliverables** you can see the different objects for each section on our application and all of the deliverables that are needed to complete each object.

1.7.1Table 1: Objectives/Deliverables

Programming	• Program the application to communicate with the database. • Program the application to communicate and read the barcode scanner.
Backup	• Make sure all the information is being backed up. • Log all activity that occurs within the application. • Log transactions and votes.
Storage	• Find a place to store all data. • Make sure all changes in the application are saved.
Connectivity	• Test to make sure everything is connected and communicating.
Software	• Barcode scanner software. • Make sure it is bug free. • Test on all users. • Make sure the application is user-friendly and easy to use.
Application	• Create application interface. • Make sure application is user-friendly. • Enter application information. • Testing the application.
Security	• Research Blockchain. • Implement Blockchain into our application. • Testing Blockchain.
Database	• Create users to link to the application. • Create users in the database.

	• Create ID numbers assigned to the users.

1.8 Budget

After working months on creating our application and other components, we have created and demonstrated in the **Table 2: Budget** below the amount of money it would cost a person to create an application. There is a lot more to an application than just creating it, you also provide storage, allow testing, gather testing devices, and find software that is compatible. As you can see in the table below, everything starts to add up fast.

1.8.1 Table 2: Budget

No.	Item	Unit, Each	Unit Price	Total
1	Storage	Virtual Server	$0.03 per gig	$30.00
2	Supplies	10 Test Phones	$200 each	$2000.00
3	Programming	1 Programmer	$75 per hour	$30,000.00
4	Software	1 Software	Free	$0.00
5	Labor	3 Employees	$80 per hour	$153,600.00
6	Security	1 Program	$20 per month	$240.00
			Total:	**$186,200.00**

1.9 Testing

We have created a testing plan that allows our group to set time frames in the order that we are able to complete each task. We have created these testing plans for two semesters, and we broke them up into weekly intervals. This helps our group stay on track and allows us to give weekly progress updates. Below in tables 3 & 4, you can find our weekly assignments.

1.9.1 Table 3: Fall Semester Testing Plan

Week 1	• Determine group project. • Determine group leader. • Determine group project name.
Week 2	• Create logo for group project. • Research information for Blockchain.
Week 3	• Research data on how to create an application. • Research data on how to create a website. • Brainstorm how to implement Blockchain within an application or website.
Week 4	• Start beginning the creation of the application prototype. • Find some of the problems we are going to be facing. • Estimate the costs of the project. • Brainstorm how to improve our project.
Week 5	• Start implementing the logo within the application prototype. • Continue to research Blockchain.
Week 6	• Brainstorm creation of database users. • Continue to work on the app prototype. • Continue research of Blockchain.
Week 7	• Implement Blockchain within the application. • Find bugs or errors within the application.
Week 8	• Complete database of users for the application. • Purchase barcode scanner software.
Week 9	• Enable the application to identify users in the database.

	• Test security on the application to make sure it is working correctly.
Week 10	• Test application to make sure is usable and user-friendly. • Test and go over documents.
Week 11	• Find and fix any bugs within the application, database, or security.
Week 12 – 15	• Continue testing

1.9.2 Table 4: Spring Semester Testing Plan

Week 1	• Create and program the application.
Week 2	• Implement the plug and play option and place it within the application to see how it responds.
Week 3	• Create the application interface. • Test the barcode scanner within the application to retrieve user data.
Week 4	• User testing.
Week 5	• Allow database to communicate with the application. • Finish creating the application. • Continue to test the application.
Week 6	• Test security settings.
Week 7	• Workout an existing bugs/problems within the application.
Week 8	• Find any vulnerabilities and fix them.
Week 9	• Finish any last minute touchups and do final testing.
Week 10 – 15	• Finalize all application data. • Finalize all application settings. • Final test. • Prepare to export the application and present at the expo.

1.9.3 Testing Results

Scope

The scope of testing is to identify any bugs/problems that prohibit our application from working successfully on various devices that are running different operating systems. The devices our application supports are Android, Apple, and Windows operating systems, via tablet or smartphone.

Objective

The objective of testing is to ensure our application software is running correctly on different devices, allowing user input, and giving the user the correct feedback. Testing will help us identify the problem, and will help us to determine the solution. Our developers will run test on different devices using different operating systems to help us ensure our application is running smoothly on all devices. We will also have random users of all ages testing the application on the devices as well. This allows our developers to see how user-friendly the application is to people of all age groups.

Reporting and Log Testing

Any bugs that are discovered throughout the process of testing will be documented and the developers will analyze the bugs and will determine

the fix. This process will help keep track of all bugs that are happening to ensure we do not encounter them again in the future, and if we do then we know how to fix it.

Pass/Fail

Our tests will be determined on a pass/fail basis. Human error will not be included in these trials, only system errors.

Testing Devices

The devices our developers and users will be testing on will be on the following devices listed below. We want to test the most up-to-date phones since we have found out that most users will be using the newest products that are being released by the marketplace. We want to deliver our customers the best product we are able to, and by testing with these types of measure we will be able to deliver a great product.

Devices

- Samsung Galaxy 7

- IPhone 7 Plus
- Samsung Tablet
- Surface Pro 4
- IPad Pro

Operating System

- MAC OS
- Android
- Windows

1.9.4 Table 5: Stability Testing

Stability testing is focused on the application running correctly on different devices and operating systems. Below we have testing done by developers and users when trying to use the application on different devices.

Tester	Date	Device	OS	Expected	Pass/Fail	Bug
Developer	11/12/16	IPAD Pro	MAC OS	Runs on the MAC OS smoothly without any bugs or freezing.	Pass	None
Developer	12/01/16	Samsung Tablet	Android	Runs on tablet without	Pass	None

				freezing or any bugs.		
Developer	01/06/17	IPhone 7 Plus	MAC OS	Runs without any bugs or freezing.	Pass	None
Developer	1/12/17	Surface Pro 4	Windows	Runs without any bugs or freezing.	Pass	None
Young User	12/08/16	Samsung Galaxy 7	Android	Runs without any bugs or freezing.	Pass	None
Middle-age User	1/05/17	IPAD Pro	MAC OS	Runs without any bugs or freezing.	Pass	None
Older User	11/25/16	IPhone 7 Plus	MAC OS	Runs without any bugs or freezing.	Pass	None

1.9.5 Table 6: User Interface Testing

User interface testing focuses on the usability and appeal of the application to the user. Each user will grade their performance of the application on how hard/easy it was to use it and will also grade the appeal of the application.

Tester	Date	Device	OS	Expected	Usability	Appeal	Pass/Fail
Young Adult	1/1/17	IPad Pro	MAC OS	Application runs smooth, and all button and	Easy	Great	Pass

				screens are easily managed.			
Young Adult	1/5/17	Samsung Galaxy 7	Android	Application runs smooth, and all button and screens are easily managed.	Easy	Great	Pass
Middle-age Adult	1/12/17	Surface Pro 4	Windows	Application runs smooth, and all button and screens are easily managed.	Usable	Nice	Pass
Middle-age Adult	1/14/17	IPhone 7 Plus	MAC OS	Application runs smooth, and all button and screens are easily managed.	Easy	Good	Pass
Middle-age Adult	1/20/17	Samsung Tablet	Android	Application runs smooth, and all button and screens are easily managed.	Easy	Great	Pass
Older Adult	1/27/17	Samsung Galaxy 7	Android	Application runs smooth, and all button and screens are easily managed.	Hard, User claims they couldn't read the text.	Decent	Pass

Older Adult	2/01/17	IPhone 7 Plus	MAC OS	Application runs smooth, and all button and screens are easily managed.	Easy	Great	Pass

1.9.6 Table 7: Functionality Testing

Functionality testing focuses on the features that are built into the application, and to make sure they are all working properly. We also want to make sure that all of the feature are understood by the users, so they know what they are used for.

Tester	Date	Device	OS	Expected	Usability	Features
Young Adult	1/29/17	IPhone 7 Plus	MAC OS	All features and function are working correctly and user understands what they are used for.	Very usable, and likes the application.	Great features and understands how to use them.
Young Adult	2/01/17	Samsung Galaxy 7	Android	All features and function are working correctly and user understand	Understands how to use the application and all the features.	Loves the features and thinks it was a great idea.

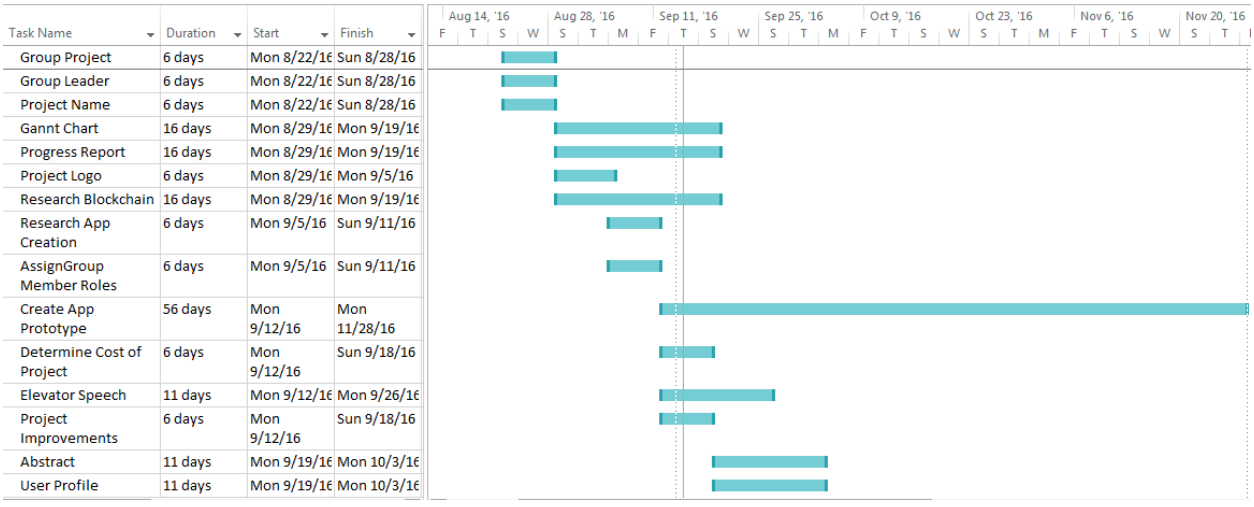
				s what they are used for.		
Young Adult	1/28/17	IPAD Pro	MAC OS	All features and function are working correctly and user understands what they are used for.	Very easy to use and understands the process.	Thinks the application can change the way people vote.
Middle-age Adult	1/03/17	Samsung Tablet	Android	All features and function are working correctly and user understands what they are used for.	Easy to use and loves the application.	Great idea!
Middle-age Adult	1/05/17	Surface Pro 4	Windows	All features and function are working correctly and user understands what they are used for.	Easy to download and struggled slightly finding where to start.	One the features were explained he understood why we added them to the application.
Older Adult	1/15/17	Samsung Galaxy 7	Android	All features and function are working correctly and user	Managed to find their way through the application and	Loves the application and how it makes voting more flexible.

				understands what they are used for.	understood the process.	
Older Adult	1/21/17	IPhone 7 Plus	MAC OS	All features and function are working correctly and user understands what they are used for.	Understood the application and managed his way through the process easily.	Likes the idea of being able to vote from a mobile phone.

1.10 Gantt Chart

A Gantt chart is a visual that helps our group understand the timeframe that our project needs to be completed by. It also helps each group member understand all of the steps that need to be completed for each task. We have provided two visual charts below that will help users understand the step by step process we took during these two semesters.

1.10.1 Figure 2: Gantt Chart Fall Semester

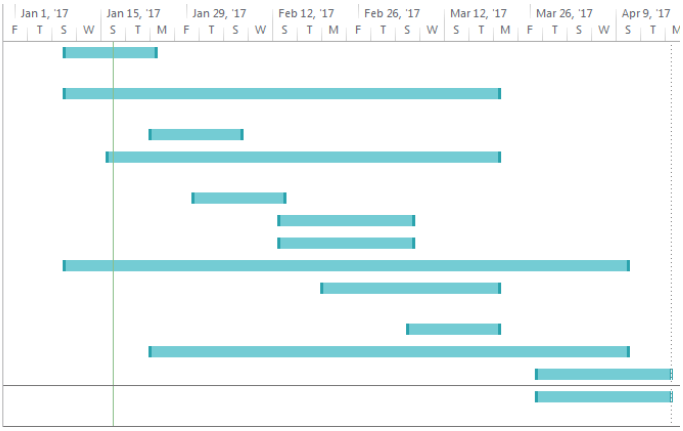


Task Name	Duration	Start	Finish
Group Project	6 days	Mon 8/22/16	Sun 8/28/16
Group Leader	6 days	Mon 8/22/16	Sun 8/28/16
Project Name	6 days	Mon 8/22/16	Sun 8/28/16
Gannt Chart	16 days	Mon 8/29/16	Mon 9/19/16
Progress Report	16 days	Mon 8/29/16	Mon 9/19/16
Project Logo	6 days	Mon 8/29/16	Mon 9/5/16
Research Blockchain	16 days	Mon 8/29/16	Mon 9/19/16
Research App Creation	6 days	Mon 9/5/16	Sun 9/11/16
AssignGroup Member Roles	6 days	Mon 9/5/16	Sun 9/11/16
Create App Prototype	56 days	Mon 9/12/16	Mon 11/28/16
Determine Cost of Project	6 days	Mon 9/12/16	Sun 9/18/16
Elevator Speech	11 days	Mon 9/12/16	Mon 9/26/16
Project Improvements	6 days	Mon 9/12/16	Sun 9/18/16
Abstract	11 days	Mon 9/19/16	Mon 10/3/16
User Profile	11 days	Mon 9/19/16	Mon 10/3/16

User Profile	11 days	Mon 9/19/16	Mon 10/3/16
Research Errors on App Prototype	6 days	Mon 10/3/16	Mon 10/10/16
Problem Statement	6 days	Mon 10/3/16	Mon 10/10/1
Progress Report 2	6 days	Mon 10/10/1	Mon 10/17/1
Use Case Diagram	6 days	Mon 10/10/1	Mon 10/17/1
Final Draft Report	6 days	Mon 10/17/1	Mon 10/24/1
Prepare Final Presentation	11 days	Mon 10/24/16	Mon 11/7/16
Final Presentation	11 days	Mon 11/14/1	Mon 11/28/1
Complete Final Prototype of App	11 days	Mon 11/7/16	Mon 11/21/16

1.10.2 Figure 3: Gantt Chart Spring Semester

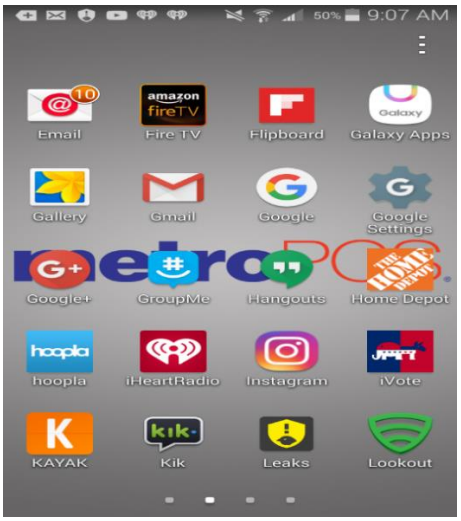
Task Mode	Task Name	Duration	Start	Finish	Predecessors
	Gantt Chart & Deliverables	11 days	Mon 1/9/17	Mon 1/23/17	
	Implement BlockChain	51 days	Mon 1/9/17	Mon 3/20/17	
	Test/Plan Report	11 days	Mon 1/23/17	Mon 2/6/17	
	Prepare for Class Presentations	46 days	Mon 1/16/17	Mon 3/20/17	
	Abstract	11 days	Mon 1/30/17	Mon 2/13/17	
	Draft Tech Poster	16 days	Mon 2/13/17	Mon 3/6/17	
	Draft Report	16 days	Mon 2/13/17	Mon 3/6/17	
	Testing Application	66 days	Mon 1/9/17	Mon 4/10/17	
	Add New Features to Application	21 days	Mon 2/20/17	Mon 3/20/17	
	Final Poster	11 days	Mon 3/6/17	Mon 3/20/17	
	Prepare for Expo	56 days	Mon 1/23/17	Mon 4/10/17	
	Final Report	16 days	Mon 3/27/17	Mon 4/17/17	
	Final Library Copy Final Report	16 days	Mon 3/27/17	Mon 4/17/17	



1.11 Application Prototypes

Below are screen shots and prototypes of the application we have created called iVote. We have provided photos of our application icon, home screen, and the information that will be displayed after scanning your ID. Also below are some examples on the process of Blockchain and how it works within our application.

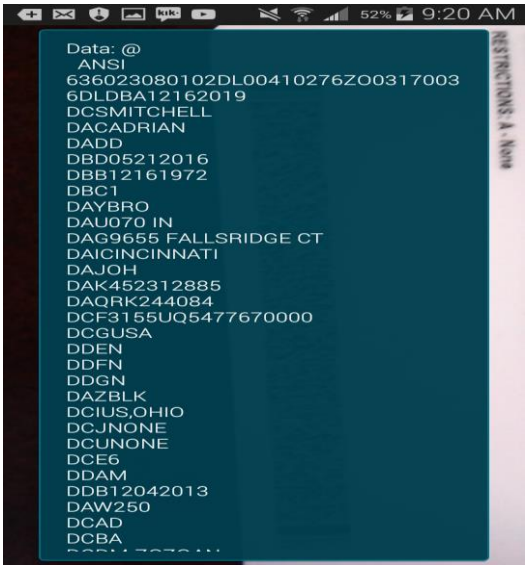
1.11.1 Figure 4: iVote Application Icon



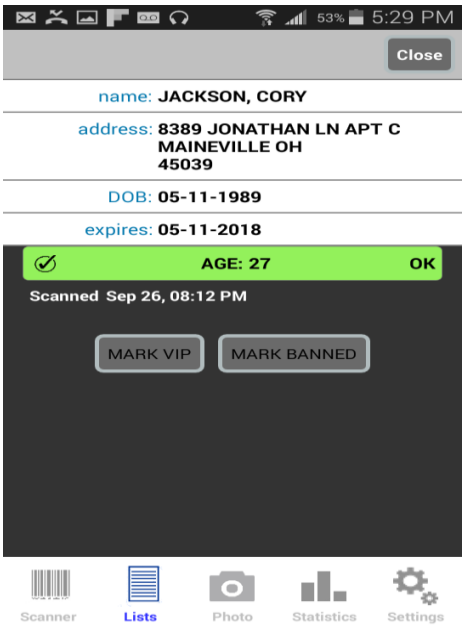
1.11.2 Figure 5: iVote Application Home screen

1.11.3 Figure 6: 2D Barcode Scanner

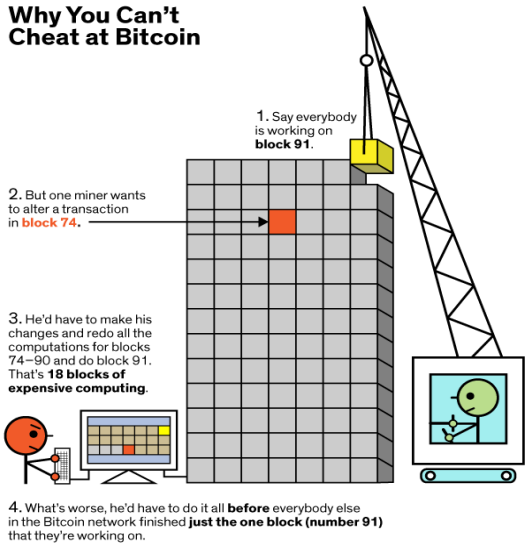
1.11.4 Figure 7: SQL Database User Look-up



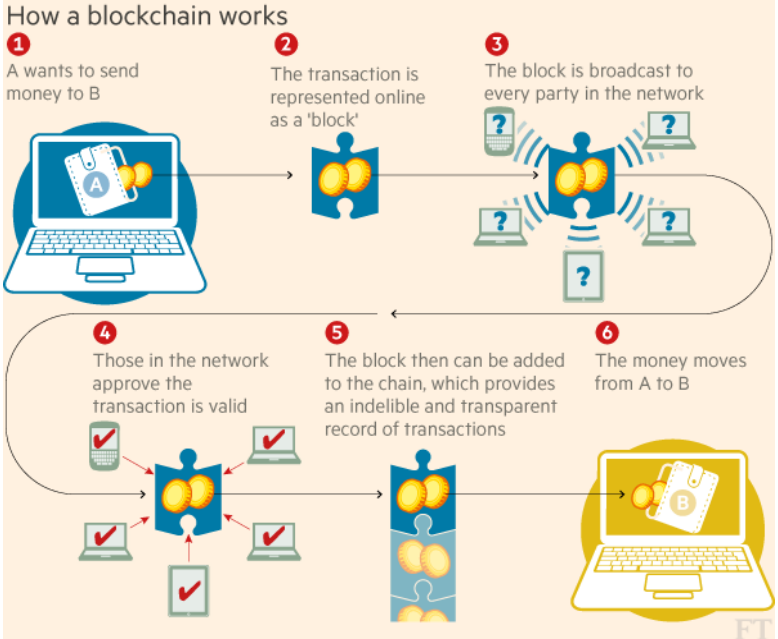
1.11.5 Figure 8: Barcode Scanner User Information



1.11.6 Figure 9: Blockchain Technology



1.11.7 Figure 10: How Blockchain Technology Works



1.12 Conclusion

During the duration of this project our group learned a lot about each other and also gained more knowledge in brainstorming, creating, testing, and implementing. We learned several different areas of IT, including software, networking, and security. Our project was created upon using and learning all three of these areas. The software base was used when we were creating the application for the users to be able to vote via their smartphone. We also created a database with all of the user's information, to be able to verify if they are a registered voter or not. The networking end was used when we were developing a way our database could connect to our application and pull the user's data when their ID was scanned through the phone's camera. The security aspect was used when we were securing all of the votes that were being electronically submitted throughout several different locations using Blockchain technology.

We are preparing to present our prototype to an IT based audience in a few weeks, showing them the progressions we have taken throughout the semester in order to complete this project on time. We have brainstormed great ideas and found a way to implement them all together to create an application that will change the way people are able to vote.

1.13 Bibliography

How Will Blockchain Technology Transform Financial Services? Financial Times, 3 Nov. 2015. Web. 8 Nov. 2016. <https://www.weforum.org/agenda/2015/11/how-will-blockchain-technology-transform-financial-services/>.

""Why You Can't Cheat at Bitcoin" - A Simple Infographic from IEEE Explains Th... | Rebrn.com." *Rerbn*. N.p., n.d. Web. 08 Nov. 2016. <http://rebrn.com/re/why-you-cant-cheat-at-bitcoin-a-simple-infographic-from-ieee-e-152759/>.

By Evan Dashevsky February 23, 2015 10 Comments. "How to Create an App for IOS, Android, or Windows Phone." *PCMAG*. PC, 23 Feb. 2015. Web. 08 Nov. 2016. <http://www.pcmag.com/article2/0,2817,2476480,00.asp>.

Heng, Christopher. "What Is MySQL? What Is a Database? What Is SQL?" *Thesitewizardcom RSS*. N.p., n.d. Web. 08 Nov. 2016. <https://www.thesitewizard.com/faqs/what-is-mysql-database.shtml>.

Moore, Robert. "American Idol Voting 'beats US Presidential Election'" ITV News. Washington Correspondent, 24 May 2012. Web. 01 Nov. 2016. <http://www.itv.com/news/2012-05-24/x/>.