

# Smart Voting

---

## Software Requirements Analysis and Design

Version 1.0

2021 November 10

## REVISION HISTORY

Revision	Date	Author	Summary of Changes

## DOCUMENT APPROVAL LIST

Version	Approved By	Signature	Date
1.0	Matt Campbell Stephen Davis Satabdi Sangma Michael Sirna	<a href="mailto:matthew.campbell@georgebrown.ca">matthew.campbell@georgebrown.ca</a> <a href="mailto:stephen.davis@georgebrown.ca">stephen.davis@georgebrown.ca</a> <a href="mailto:satabdi.sangma@georgebrown.ca">satabdi.sangma@georgebrown.ca</a> <a href="mailto:michael.sirna@georgebrown.ca">michael.sirna@georgebrown.ca</a>	2021-11-10

## DOCUMENT DISTRIBUTION LIST

Version	Name of the Receiver / Group	Date
1.0	Matt Campbell Stephen Davis Satabdi Sangma Anjana Shah Michael Sirna	2021-11-10

## **TABLE OF CONTENTS**

<b>DOCUMENT APPROVAL LIST</b>	<b>2</b>
<b>DOCUMENT DISTRIBUTION LIST</b>	<b>2</b>
<b>TABLE OF CONTENTS</b>	<b>2</b>
<b>1 INTRODUCTION</b>	<b>4</b>
1.1 PURPOSE	4
1.2 SCOPE	4
<b>2 SYSTEM OVERVIEW</b>	<b>5</b>
2.1 PROJECT PERSPECTIVE	5
2.2 SYSTEM CONTEXT	5
2.3 GENERAL CONSTRAINTS	6
2.4 ASSUMPTIONS AND DEPENDENCIES	6
<b>3 FUNCTIONAL REQUIREMENTS</b>	<b>7</b>
3.1.1 EASE OF ACCESS	7
3.1.2 VOTING SECURITY	7
3.1.3 ELECTION RESULTS	7
3.1.4 PARTY INFORMATION	7
3.2 USE CASES	7
3.2.1 BENEFITS FOR SOME GROUPS	7
3.2.2 ANALYTICAL TOOLS, RESULTS, AND CANDIDATE PROFILES	8
3.2.3 UNFORESEEN NATURAL OR ARTIFICIAL DISASTERS	8
3.3 DATA MODELLING AND ANALYSIS	8
<b>4 NON-FUNCTIONAL REQUIREMENTS</b>	<b>15</b>
<b>5 LOGICAL DATABASE REQUIREMENTS</b>	<b>15</b>
<b>6 OTHER REQUIREMENTS</b>	<b>16</b>
<b>7 APPROVAL</b>	<b>16</b>

# 1 INTRODUCTION

## 1.1 PURPOSE

The purpose of this document is to outline the high level software requirements for the Smart Voting System; an accessible and easy to use online web application that allows users to vote with comfort and security in their elections. The system will be outfitted with a ledger database to store user votes in a secure manner. On top of this, the system will have analysis tools for elections so voters can view results with little down-time. The tools also have the added benefit of helping candidates in their campaign to strategize effectively. With this, users will be able to view candidate pages and information to help in making more informed decisions about their votes.

## 1.2 SCOPE

### In Scope (What It Will Do)

- Be an **alternative** voting system to today's current system
  - Easier voting capabilities for people with health risks and disabilities.
  - Specifically designed to combat unforeseen disasters or events (i.e a global pandemic)
  - Beneficial for those who do not have time or the capability to get to a voting station during the allotted time frame.
  - Allows secure voting globally.
  - Possible thanks to moving voting on to an online web application (Internet connection required)
  - Ledger Database used for storing votes and who has voted (Two separate, non cross identifiable tables)
  - Mail out identification system will be used to verify security before voting
  - Faster and more efficient then the current voting system
- Display analytics for users so there is no waiting and candidates may strategise their campaign around results
  - Provide users with up-to-date information about Candidates allowing the users to stay up to date and make voting decisions they can feel safely informed on.

### Out of Scope (What It Will Not Do)

- We won't be distributing the voter cards. That would be the job of the government / postage system
- The app will not fill out or update candidate information / party information. That is the duty of the party itself
- General Election information is handled by election officials.
- This app **IS NOT** a social media app. The only users who can create and edit accounts are political candidates / their parties. Voter accounts will be pre-created by government officials. Voters will not be able to create or personalize their accounts.

## **2 SYSTEM OVERVIEW**

### **2.1 PROJECT PERSPECTIVE**

The Smart Voting System is a follow-on member to current voting systems that are in place. It is meant to work side by side with these systems, and by no means to replace them as some people still must rely on those other systems, and some prefer personal interaction. To add to this, some are less adept at using technology, but given this it is important that the system is made easy to use and accessible to anyone.

As stated previously, the origin of the system was to be an alternative to the current system; specifically for those who may not be able to go out for reasons from health issues, to time management. The idea rose from the October 2020 Federal Election that occurred during a global pandemic. During this, fears of election fraud and overall paranoia were arguably at an all time high, especially due to elections that occurred in the United States, and information in the news and on the internet. This app would be an excellent contribution to hopefully reduce this paranoia, and overall stigma with elections on top of helping those who need this service to vote.

### **2.2 SYSTEM CONTEXT**

#### **2.2.1 WHAT HAS TO BE DEVELOPED?**

An online voting web application that allows users to vote anywhere at any time given that they have valid identification and their voter cards. The system will use a ledger database to store votes so they are secure and cannot be tampered with after voting. The web application will also be outfitted with analysis tools so candidates can view and strategise their campaigns and users are able to view results and information on their ridings as it is updated.

#### **2.2.2 WHAT IMPACTS THE DEVELOPMENT PROCESS?**

A major portion of the development process is funding as without funding projects cannot exist, and developers cannot be paid. In this case, of course, funding will not be an issue as the Smart Voting system is part of the George Brown College Capstone Program. In any other context, though, funding would be a primary issue of a project team.

Another big issue is one that was experienced recently and the reason this idea came to fruition; unforeseen natural disasters. Since this problem is still currently happening today, it poses little risk to the project as we will not have to halt or slow production because of switching to an online workflow. Again, though, if there were no disaster or pandemic, this would be a large issue that would definitely halt production on many projects, as we have seen with COVID-19.

### **2.2.3 WHAT CAN BE DISREGARDED?**

The main priority of the system would be to securely store voter information including voter's cast ballots. Without that information, the system would be useless. Because of this, in the case of a halt in production, or a deadline needing to be met, the analytics system and results system would have to be scrapped. Of course, this is not a huge issue because the Smart Voting system would be an alternative way to vote, and other means of getting analytics and polls would still exist. Ballots would still be counted on the system, but there would not be any way to display that information to users. Along with analytics being scrapped, the political candidate profiles would also probably have to be taken along with it. Essentially, in the scenario that a deadline must be met, the bare minimum would have to be developed; the bare minimum being the number one function and priority of the system.

## **2.3 GENERAL CONSTRAINTS**

Due to the fact that "Smart Voting" is an online web application, there are very few, if at all, constraints on the system itself. All tools needed to create the system are available. While the application will be web based, the authentication measures will feature a voter-unique, printable mailer that will be sent to each voter. With this, it must be mentioned that we are not in charge or in control of Canada Post or other courier services that may cause delays due to unforeseen circumstances in delivering voter registration mailers.

## **2.4 ASSUMPTIONS AND DEPENDENCIES**

It is assumed that all users who use this product would have access to an internet connection and device that they are able to use. This product is meant as an alternative to voting, and by no means implies that it will completely replace current voting systems if implemented. The system is simply an extra option for citizens, especially those who have little time, health issues or are deployed as military or embassy staff overseas. All of this also assumes that a government would approve of the system being used across their countries. The plans and goals set out in this document and others attached are the desired outcomes and by no means represent the final product.

### **3 FUNCTIONAL REQUIREMENTS**

#### **3.1.1 EASE OF ACCESS**

“Smart Voting” is a web based application that can be accessed anywhere in the world, at any time (provided the user has an established internet connection). The program will provide ease of use to the platform for both candidates and voters, an easily accessible system with a multi-factor authentication system, as well as an updateable candidate and party information stream customizable for each candidate. The more easily accessible the system is for its users, the closer it gets to its intended purpose.

#### **3.1.2 VOTING SECURITY**

In order to provide voter security, we will provide multi-factor authentication prior to voting. Every voter will receive a unique mailer with a scannable QR code and typeable code. Alongside said mailer the voter will use their unique login information and identification number (such as a Social Insurance Number, Driver's License Number, or other valid identification) to be able to access the voting section of the app and allow them to provide a quick and easy selection of their votes.

#### **3.1.3 ELECTION RESULTS**

On the backend of the program, votes will be tallied by two tables; a non-cross-identifiable ledger that will register the votes, and another that will record who has successfully voted. This will prevent users from voting more than once while also upholding the security of all user's ballots and keeping user's anonymity. From a front-end standpoint the application will display up-to-date voting results throughout the election period and provide voters and candidates with an accurate view of the current standings.

#### **3.1.4 PARTY INFORMATION**

On the front end of the system, people who are viewing the application will be able to see information from every party and its candidates in one spot. When a voter accesses the system, the voter will be able to create a bespoke feed keeping them up-to-date on parties and candidates of their choice. Candidates will also be able to provide updates and changes to their information on their own runnings as well as allowing party leaders to make changes to their party pages information.

### **3.2 USE CASES**

#### **3.2.1 BENEFITS FOR SOME GROUPS**

The main purpose of the Smart Voting System is to aid people in voting; especially those who can not due to health concerns and issues, those who are deployed or work in other countries temporarily, and those who simply have no time to plan a trip out to vote. All concerns are addressed and solved with the addition of this system as people would be able to vote from anywhere in the world at any time they want to without hassle. To add to this, voting would be instantaneous, and voters would not have to wait in long lines in order to vote.

### 3.2.2 ANALYTICAL TOOLS, RESULTS, AND CANDIDATE PROFILES

Analytical tools and results are a major benefit to having an online system as votes and results would be able to be viewable instantly from anywhere without having to wait for votes to be tallied. With this, user's would be up to date with all information in an election, and candidates would be able to strategise their campaigns around the results.

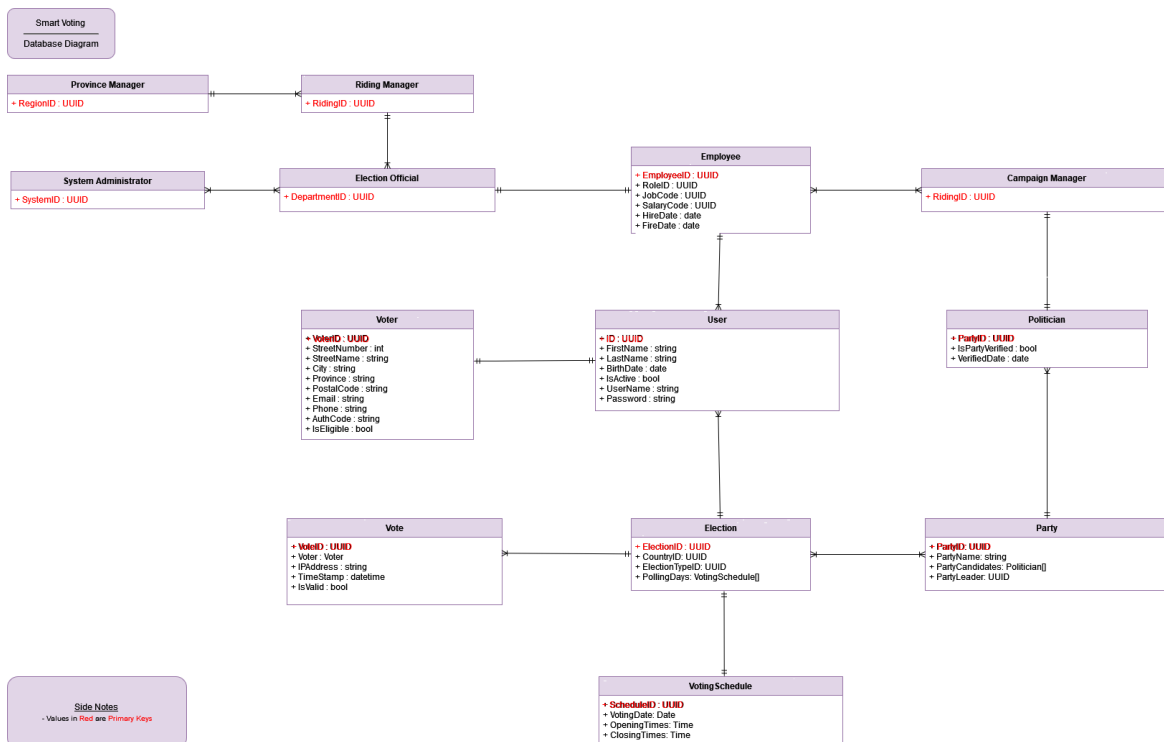
To add to candidates, they would be able to have their profile and political views listed on their profiles so users can better understand their objectives and perhaps make more informed decisions. This would be especially beneficial for those who do not have time to look over candidate information and make decisions before going out to polling stations.

### 3.2.3 UNFORESEEN NATURAL OR ARTIFICIAL DISASTERS

All of the above use cases were all outlined in elections that happened throughout the world from March 2020 to the present day because of a global pandemic; COVID-19 to be specific. These natural and unforeseen global disasters are a very real and possible obstacle that can stop elections and all their plans in their tracks. With an online voting system, all of this could be bypassed. In the case of a global pandemic; there would be no human interaction and therefore no way transmission could happen between people, which solves the entire problem and allows elections to continue seamlessly.

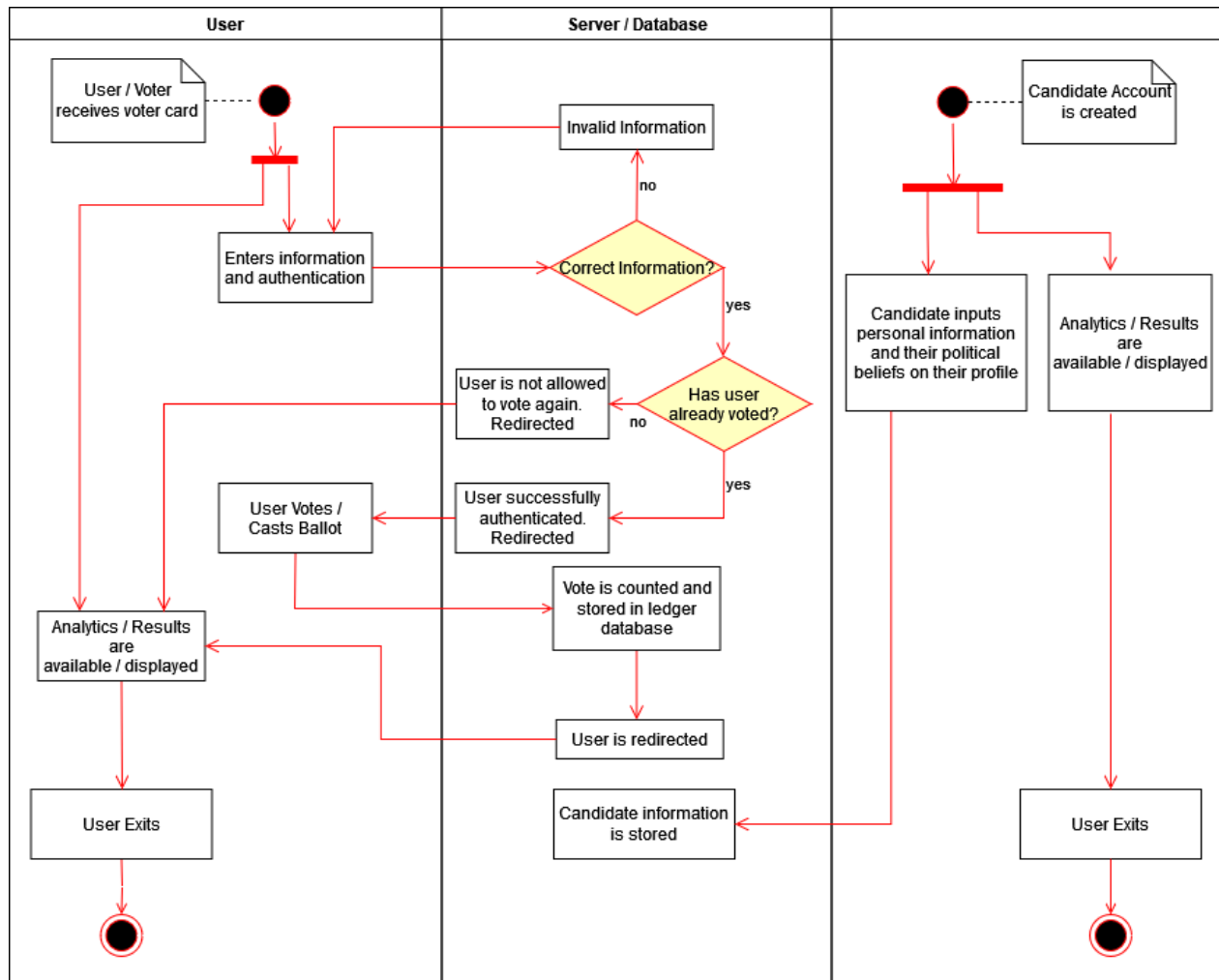
## 3.3 DATA MODELLING AND ANALYSIS

### 3.3.1 DATABASE MODEL (NORMALIZED DATA MODEL)



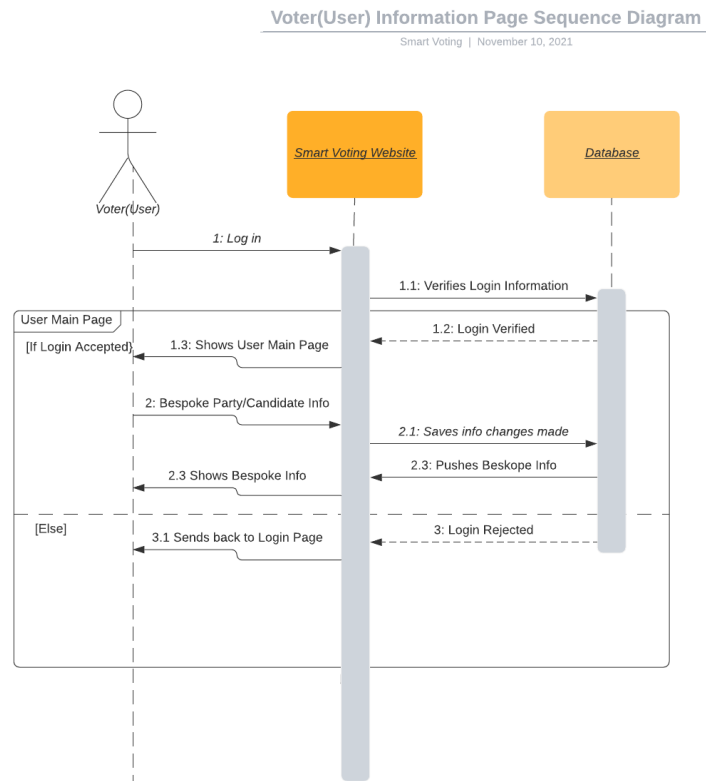


### 3.3.2 ACTIVITY DIAGRAM

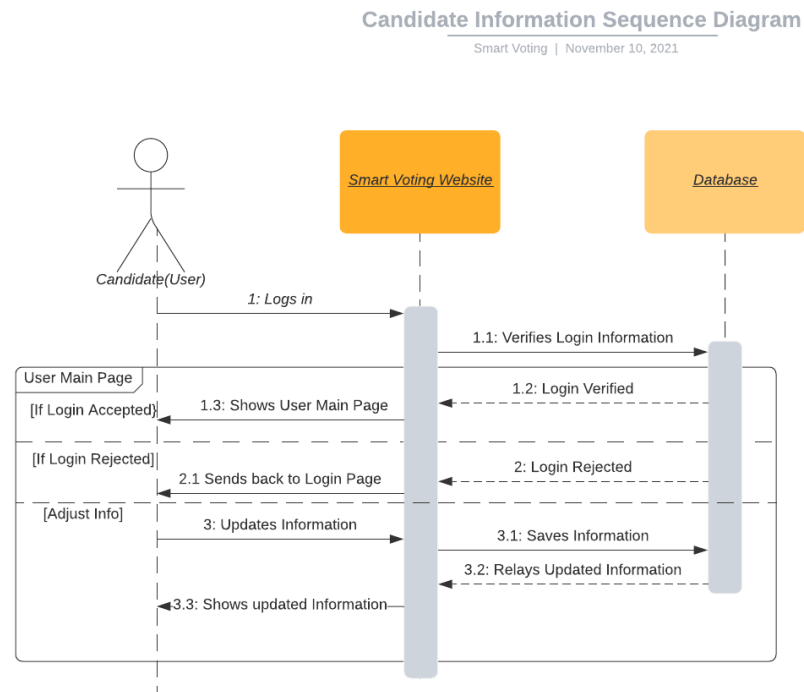


### 3.3.3 SEQUENCE DIAGRAMS

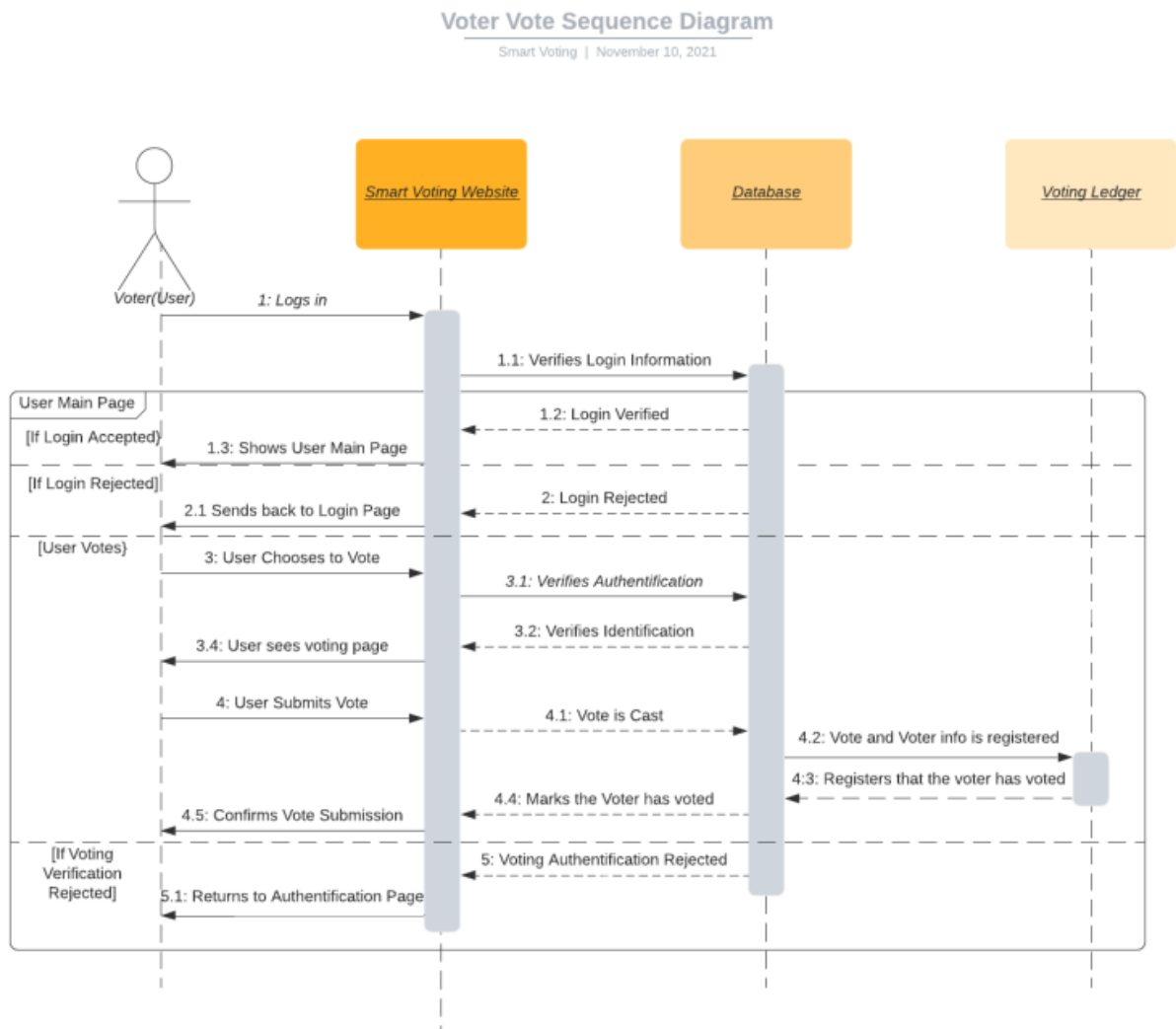
#### Sequence Diagram 1: Voter (User) Information Page



#### Sequence Diagram 2: Voter (User) Information Page

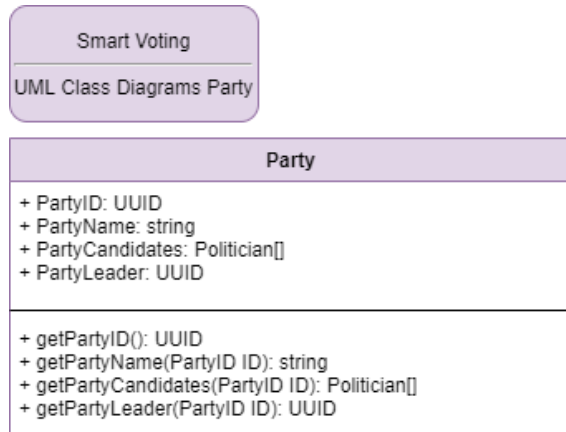


### Sequence Diagram 3: Voter Votes

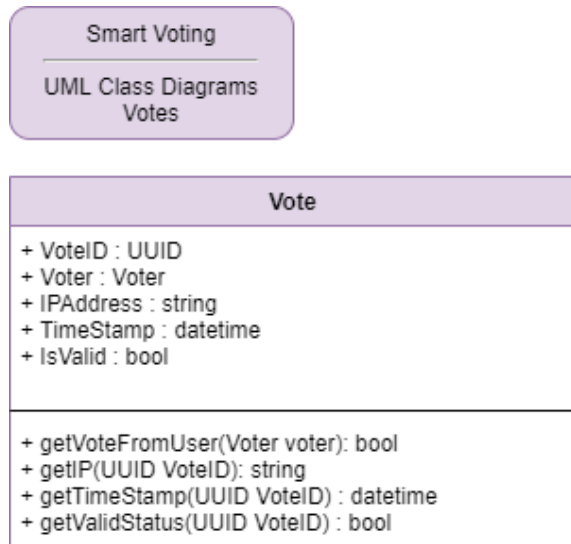


### 3.3.4 UML CLASS DIAGRAMS

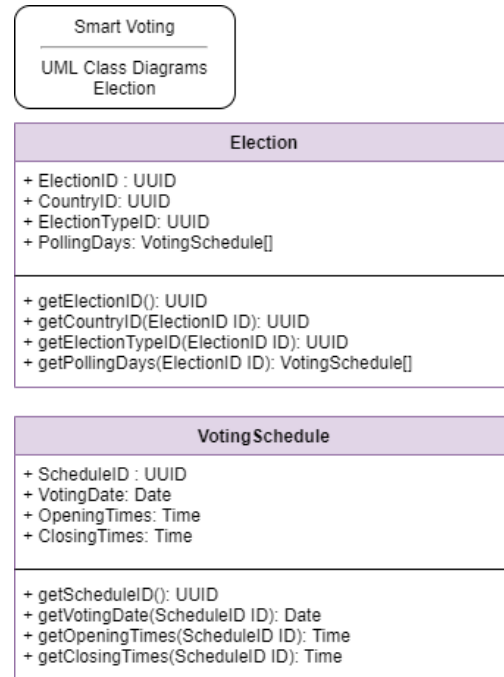
UML Diagram - Party



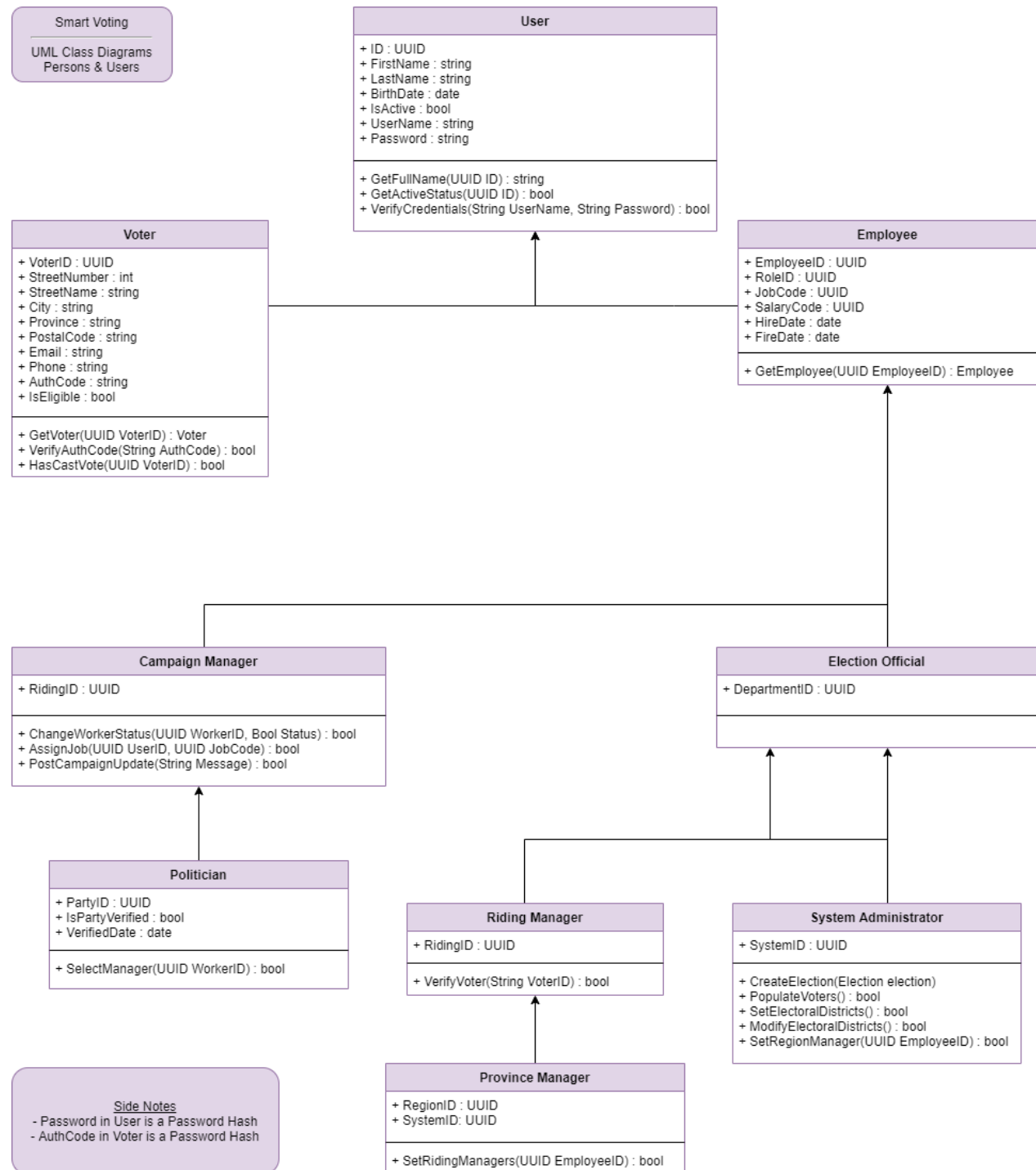
UML Diagram - Vote



UML Diagram - Election

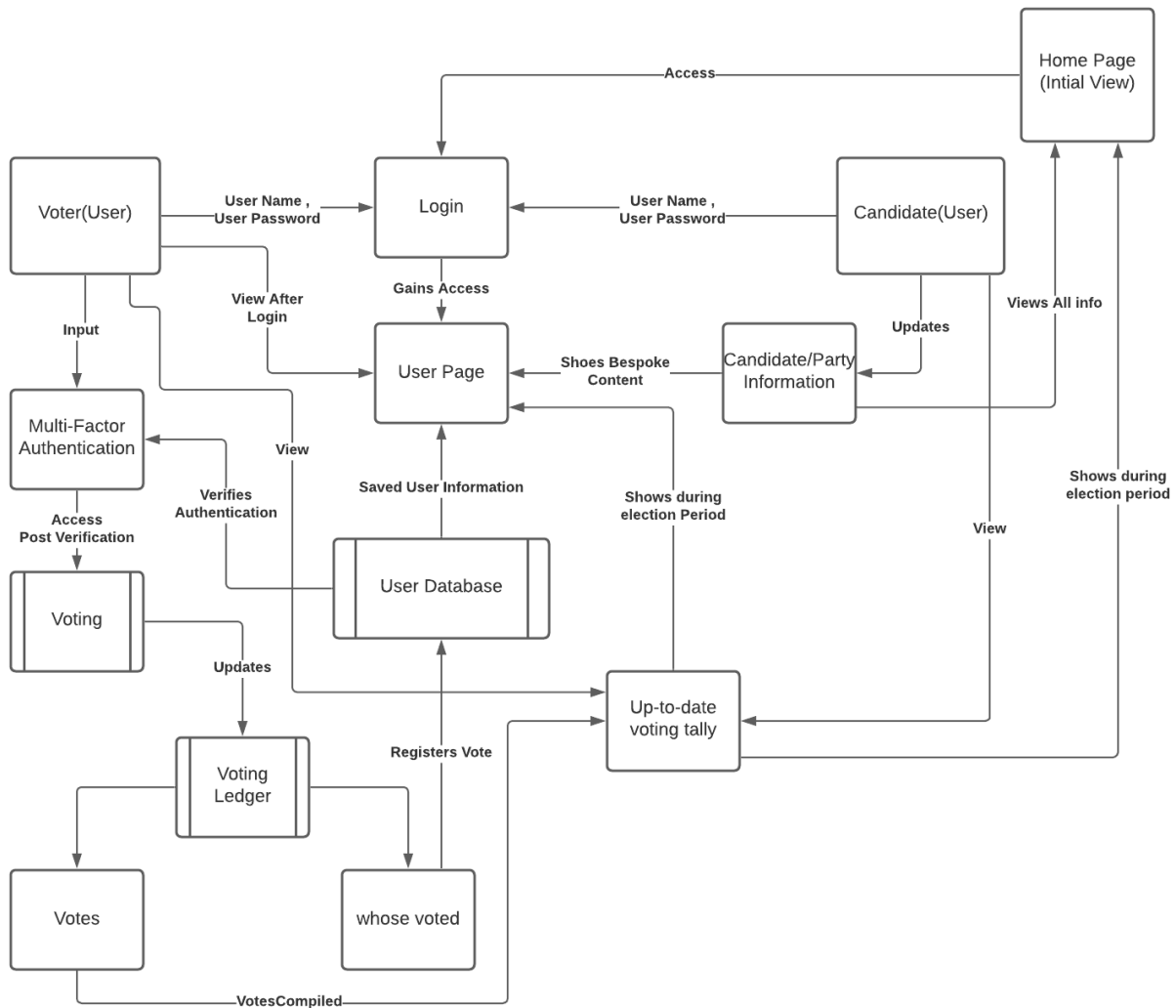


## UML Diagram - Users



## 3.4 PROCESS MODELLING

### 3.4.1 DATA FLOW DIAGRAM



## 4 NON-FUNCTIONAL REQUIREMENTS

- During open voting windows, system downtime may not exceed 5 minutes.
- Entire voting process must take voters less than 5 minutes total.
- Web application must be redundant in the event one AWS data center goes offline.
- Web application must meet federal privacy laws.
- QLDB instances must meet Electoral Integrity and Security standards.

## 5 LOGICAL DATABASE REQUIREMENTS

The Smart Voting application will utilize three different types of databases. They include PostgreSQL, AWS DynamoDB and AWS QLDB. The system will also be using S3 buckets.

### Amazon Web Services - RDS PostgreSQL

Amazon RDS is a traditional relational database service and PostgreSQL will be utilized as the database engine. PostgreSQL is an open source database engine that operates similar to MySQL or Microsoft SQL. PostgreSQL will mainly be used for the voter's personal data, accounts for election and party officials and logs of any errors in the application.

### Amazon Web Services - DynamoDB

DynamoDB is a NoSQL database that operates similarly to MongoDB. It supports key-value and document data models. DynamoDB will mainly be used to store party campaign and candidate information.

### Amazon Web Services - Quantum Ledger Database (QLDB)

Quantum Ledger Database is a ledger database with immutable and cryptographically verifiable transaction logs. QLDB will only store two things; which votes have been cast, and which voters have cast their votes. To comply with election laws, all entries in QLDB will be stored in separate database tables, and it is not possible to cross reference any entries.

### Amazon Web Services - S3 Buckets

Amazon S3 is a blob storage system for various types of files. S3 will be used for storing any media files, PDF documents or other large files that are not suited to be stored in a database.

### Data Retention & Integrity

Both data retention and integrity are vital for elections. With this in mind, all three databases and all S3 buckets will have strict access control measures. Only election officials with the correct permissions will be able to add and edit voter information. Only campaign officials will be able to modify and change information directly related to their respective campaigns. With regards to the actual votes cast and who has voted, federal law states election ballots are anonymous. As a result, the QLDB tables are not able to be linked together. Another regulation states once a ballot has been cast, it can not be changed. QLDB entries are not able to be modified or deleted, even by a system administrator.

## 6 OTHER REQUIREMENTS

There are no other requirements needed for the Smart Voting System.

## 7 APPROVAL

The signatures below indicate their approval of the contents of this document:

Project Role	Name	Signature	Date
Team Member	Matthew Campbell	<a href="mailto:matthew.campbell@georgebrown.ca">matthew.campbell@georgebrown.ca</a>	2021-11-10
Team Member	Stephen Davis	<a href="mailto:stephen.davis@georgebrown.ca">stephen.davis@georgebrown.ca</a>	2021-11-10
Team Member	Satabdi Sangma	<a href="mailto:satabdi.sangma@georgebrown.ca">satabdi.sangma@georgebrown.ca</a>	2021-11-10
Team Member	Michael Sirna	<a href="mailto:michael.sirna@georgebrown.ca">michael.sirna@georgebrown.ca</a>	2021-11-10