

# Show of Hands - Group 7 - Project Brief

## Document Information

<b>Project name:</b>	Show of Hands
<b>Date:</b>	18/02/2021
<b>Author:</b>	Luke Tynan
<b>Owner</b>	Jason Quinlan
<b>Version:</b>	0.4

Text Key: New content since last SCRUM meeting

Updates based on most recent peer review feedback

## Definition

<b>Background:</b>	With COVID-19 still an issue in much of the world, in-person voting systems will experience lower levels of voter participation due to health concerns. With this in mind, we thought that a web-based election system would allow a user (from individuals to corporations) to hold an online election in order to maximize remote voter participation.
<b>Main Goal:</b>	A web-based election system offering high usability and customizability to election organizers, candidates and voters.
<b>Desired Outcomes:</b>	<p>As none of us have extensive experience in creating software as a team, we hope to improve our ability to work efficiently in a team. This includes our ability to communicate effectively and to organize ourselves in a way that allows us to complete tasks to a high standard and in line with the deadlines set by the Product Owner.</p> <p>Furthermore, we plan to use new frameworks to develop many aspects of the products. We hope to gain knowledge and experience in these frameworks that we can carry forward into our placements.</p>

# Show of Hands - Group 7 - Project Brief

---

**Constraints and** This project has no flexibility in terms of time or team members, as the number of team members and deadlines are fixed. Due to the current state of affairs, we will not have the opportunity to meet in person to work on the project.

Therefore, all our communication is restricted to online calls or messages.

As of now we don't plan on restricting our compatibility to one specific browser although this may be subject to change.

We have not, as of yet, encountered or discovered any major restrictions with these systems, however, we are not intending to use any other frameworks so any restrictions inherited from these systems will affect this project and will appear here as their effects are discovered during the development process.

---

**Framework:** We will be working with the React framework for the development of the frontend aspect of the web application, and Firebase for the respective backend.

React allows us to create a seamless, seemingly one page, app which is interactive and dynamic. Reactor router, React dom, and many other libraries associated with the framework also enables us to create a working prototype very easily, then convert to a fully fledged product later in the development process without requiring major architectural or structural changes

The profile system, user authentication, voting system, profile databases etc will all be managed with Firebase. Firebase provides some excellent utility to implement these features. This makes it very easy for us to integrate the frontend and backend together, interacting with Firebase from the React code.

---

**Project Approach:** We don't plan to use or build upon an existing project so our software will be developed in-house. We plan to use the Scrum framework to develop the product and conduct a daily stand up so that we are all on the same page and can ensure we're on track to meet deadlines and goals.

---

# Show of Hands - Group 7 - Project Brief

---

**Project Product Description:** A web application that serves as a platform for election organizers to hold an election. We will offer organizers a high level of customization in terms of type of voting system (ranked choice or popular vote mechanism) as well as the decision whether to have anonymous voting or not.

We will allow candidates in an election to have a profile where they can include a manifesto or credentials which would allow voters to make a more informed decision on what candidate to vote for.

We also want to offer election organizers the opportunity to choose who participates in their election, from an open or public election to an election where only a defined group of users can vote in.

---

## Outline Business Case

A web-based election system would automate an election system which is desirable for organizations as they can cut costs whilst also holding an election where the only barrier to vote will be an internet connection, resulting in higher voter participation. We are differentiating from competitors by having high levels of customization within the voting system and eligibility of voters and allowing candidates in an election to create a profile where they can include a manifesto.

## Key Stakeholders

Major Stakeholder	Notes
Jason	Product owner
Diarmuid, Finbarr, Con, Jack, Luke	Software Development team members

---

# Show of Hands - Group 7 - Project Brief

## Project Objectives

	Target	Tolerance
<b>Scope</b>	<ul style="list-style-type: none"><li>● User profile system</li><li>● Profile authentication system</li><li>● Web Application (UI) deployed on Firebase</li><li>● Candidate Profile System</li><li>● Voting System allowing users vote on user organised and customised votes</li><li>● Profile / Vote search system (search bar for finding user profiles and votes)</li></ul>	<p>These systems listed are part of the base functionality of the web app. Because of this, unless we encounter major roadblocks during the development process, there should be as little deviation as possible and the goal is to implement all of these systems thoroughly.</p>
<b>Quality</b>	<ul style="list-style-type: none"><li>● Email and identity verification on sign up</li><li>● Recaptcha on sign up and login to prevent bot accounts</li><li>● Web App responsive design to facilitate users on mobile and other devices</li><li>● Vote organisers can generate an invite link to send which invites the users contact to participate in the vote</li><li>● A role can be specified on the invite to allow the user to participate as only this role</li><li>● Search feature includes filter options and categorised results to make it easier to find more accurate results</li></ul>	<p>Each quality feature is subject to change based on progress throughout the development process. New features may be added and existing ones can be removed, altered or implemented only partially depending on needs later in the project lifecycle.</p> <p>These features are additions to the base systems listed in the project scope above.</p>
<b>Time</b>	7 weeks	No flexibility, we must use the full 7 weeks and no more.
<b>Cost</b>	N/A	N/A
<b>Risks</b>	<p>Using new frameworks or languages forces us to include learning time into our schedule which will hinder the quality of the project.</p> <p>Scheduling is strict and if a team member cannot complete their assigned duties, the project will suffer.</p>	<p>Factoring in time to learn new skills allows us to be realistic in our vision of the result. We can also develop features in order of priority so that important features can be implemented and functional,</p>

# Show of Hands - Group 7 - Project Brief

---

increasing the flexibility of the schedule.

The members of the team are flexible and can adapt to new workloads along the way allowing development to continue in the event of an absent team member.

---

**Benefits** Learning new skills will benefit each member of the team.

The user can use the platform to reduce the workload involved in organizing or participating in remote elections and votes.

---

## MOSCOW

### Must have:

- User profile system - allow users to sign up and log in via traditional email/password authentication
- Simple voting system - allow users to organise votes, participate as a candidate and vote in other user created votes
- Candidate profile system - allow users who are participating in a vote to create specific candidate profiles for this vote, enabling them to convince voters to vote for them

### Should have:

- Vote organisation options and optional restrictions
- Third party login options such as Google, Twitter and Github (maybe more)
- Allow users to vote in unrestricted votes without logging in (Anonymous Voting)

### Could have:

- Ranking / featured / top rated systems for candidates/profiles/votes
- Search filters / categories
- Ability for a vote organiser

### Will not have:

# Show of Hands - Group 7 - Project Brief

## Project Team

Role	Appointee	Duties
Team Lead	Luke	Motivating and inspiring team members.  Organizing the group's Daily Scrum meetings.  Directing team members being sensitive to the team's needs.
Front end (React)	Luke	Implementing each page and view of the app using the React framework as designed with Figma.  Implement and maintain functionality of the front end app
Front End	Finbarr	Add CSS to the React to conform with Jack's design.  Style each implemented page and view using CSS and javascript.  Responsible for icons, colours, animations and responsive design of the app
Back End	Diarmuid, Con	Implement and maintain: <ul style="list-style-type: none"><li>● Integration with the Firebase system</li><li>● Functionality of the profile and authentication system (log in, sign up)</li><li>● The voting system by storing, caching and logging relevant user interactions with the voting system (i.e. Vote creations, candidate profiles, vote results)</li></ul>

## Weekly Progress

### Week 2:

- We pushed our first basic structure of the initial layout as well as working on the CSS for it.
- We have designed the pages we have decided on, for our web app, and have started to implement the design in react with the appropriate CSS.
- Progress has been made with the interaction between the web app and the Firebase storage. We can now create collections of each type of user using our Web app.

# Show of Hands - Group 7 - Project Brief

- We decided on the majority of the Views, which the Project will incorporate. Of course, more can be added or removed if need be.
- Finalized the Scrum Rota for the coming weeks.
- Created a Use Case Diagram to help us picture the overall functioning of the App.

## Week 3:

- Currently all main pages have been designed and subsequently created and styled in React with CSS.
- Responsive CSS implemented for all pages
- User login through various providers, anonymous session functionality, user registration and email verification for newly registered users implemented.
- Password reset functionality implemented.
- Added reCaptcha to login.
- Merge of front-end pages with back end functionality.
- Successful merge to main git branch and first beta version
- Updated README for Firebase deployment.

## Week 4:

- Achieved a detailed description of the app in terms of pages, components, features and users
- Beginning of cleaning up the git branches, deleted firebaseWork and initialLayout branches, intending to start more descriptive, specific and organised branches, using long term and short term branches
- Added beginning of a MOSCOW section, this will be updated throughout the development process as new features are suggested and implemented.

## App Structure:

### Key:

>Page	> : redirect link / button Page : user is redirected to this page
-------	--

## Common Page Components:

### Header:

- Not logged in:
  - >Log In
  - >Sign Up
- Logged in:

### Footer:

- Quick link menu
  - All useful pages (Use same case logic as header)
  -

# Show of Hands - Group 7 - Project Brief

- >Profile
- >Voting
- Both cases:
  - >About Us
  - >Search
- Copyright statement

## Pages:

### Log In:

- Sign in form
- Third party login options
- >Sign Up
- >Forgot Password

### Sign Up:

- “Create Account” >Sign Up
- >Log in
- Sign up form
- On input error, give warning

### Landing Page:

- Main content:
  - Site / product introduction
  - Useful links
  - Feature information etc.

### Forgot Password:

- Email text input field
- Send request button >RequestSent
- >Log in

### RequestSent:

- Back to log in button
- Email sent notification
- Resend Link
- Email Sent AGAIN notification

### Change Password:

- New password input form
- New password guidelines
- Change Password button

### Search:

- Search Bar
- Unique vote / profile code system
- Results of search text votes/profiles
- Organise a vote button >Organiser panel
- Extra features:
  - Featured / top rated votes/profiles
  - Rating system for votes
  - Categories

### Profile

- Profile Avatar
- Info:
  - Occupation
  - Location
  - Name
- Socials links
- Tabs:
  - Bio
  - Elections/Votes/Polls
    - On going



# Show of Hands - Group 7 - Project Brief

- Filters
  - Closed/Finished
  - History
  - Show role
- Restriction/Public/Private
- Profile Settings
  - Verification

## About Us:

- Team lineup
- Text headings
- Socials Links
- How to get in touch

## FAQ:

- To be decided

## User functions / interactions:

### To organise/edit a vote:

- Vote information input form
- Options / Filters / Settings (time limit, max voters/votes)
- Generate invite link
- Invite button to invite other profiles (allow specific roles)

### Candidate Profile Set up:

- Media upload
- Text input fields (motto, methods/reasoning)
  - Could have full editor
  - Short feature bio

## Voting Page Views (Based on user state / role):

### Voter - Before vote:

- Options
- Vote button

### Voter - view Candidate profile:

- View text / images as set up by the candidate

### Viewing a vote as the organiser:

- Edit button
- Close vote button
- Kick candidate / voter
- Revoke vote

### Viewing a vote as a Candidate:

- Highlight their votes/show different stats

### Voter - After vote:

- View stats / candidates / no edit or interaction buttons

# Show of Hands - Group 7 - Project Brief

## Functions / Features available to user types:

### Anonymous User:

- Register
  - Email verification
- Log in
  - Forgot Info
- View Unrestricted / Public votes / profiles
- No access to vote organisation

### Logged In (Unverified):

- Same restrictions as anonymous user
- Access profile tab
  - Verify - Have option to resend verification email

### Logged In (Verified):

- Full access to vote organisation / voting / candidacy
- Access to profile tab

## Use Cases:

This section will show the exact flow path a user takes to use each function of the app. Sometimes the user must have a specific state before being able to use a feature, for example, to vote in a restricted vote, a user must be logged in and have a verified email address. Required states are notated by surrounding the state in brackets and appear beside the functions title, e.g. (LoggedIn, Verified)

### Key:

### Pages:

**LP:** landing page, **SU:** Sign up page, **LI:** Log in, **PP:** profile page, **VP:** Voting page, **SP:** Search page

### User States:

**Registered:** User must have an account, **LoggedIn:** User must be signed in to their account,

**Verified:** User must have verified their email address

### SignUp:

LP > SU

### SignIn: (Registered)

LP > LI

### Find a Vote:

# Show of Hands - Group 7 - Project Brief

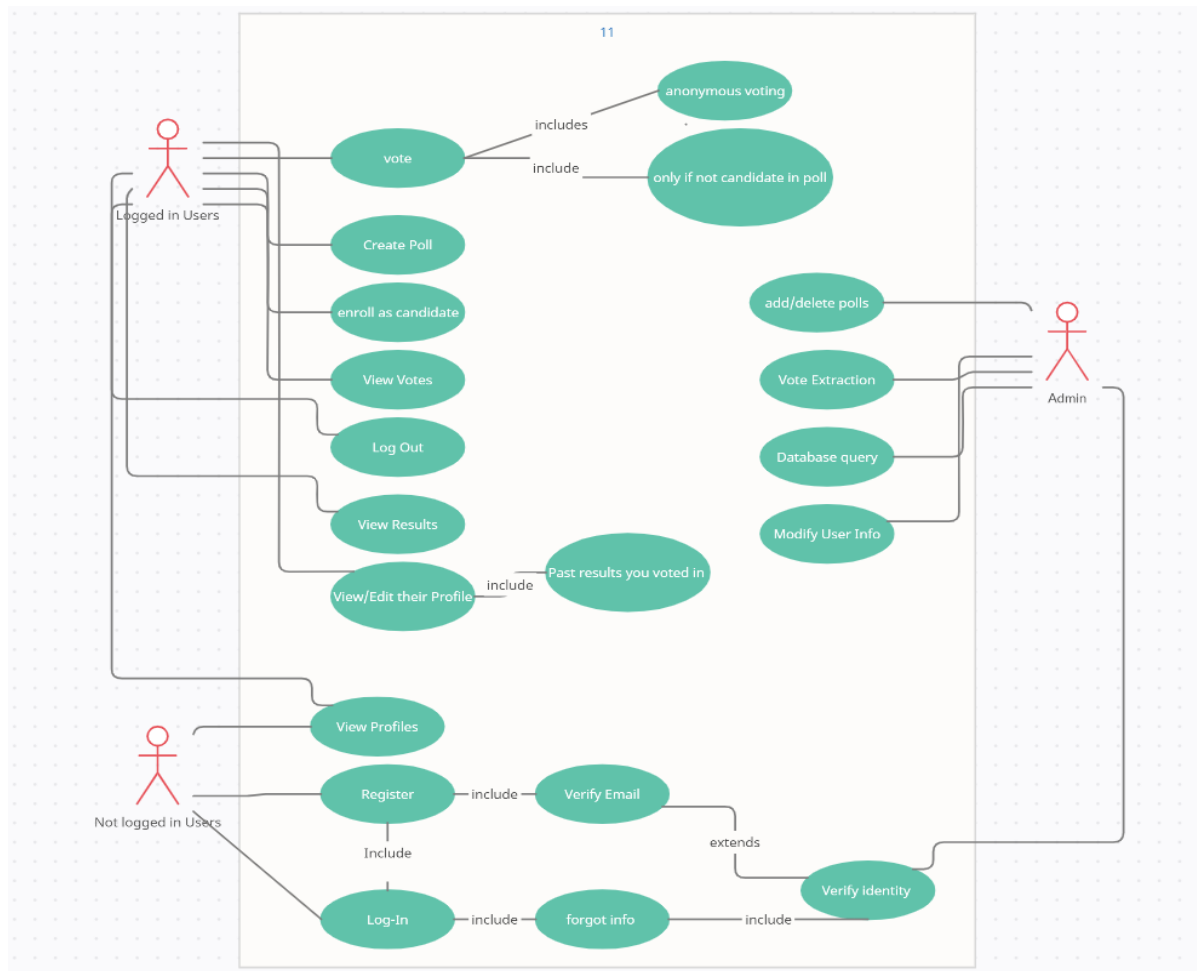
LP > VP

**Create a Vote:** (Verified)

LP > VP > "Create a Vote"

**View User's Elections (Current/past etc.)**

LP > SU > Verify > LI > PP > elections tab (elections lists)



----- PRINCE2® Templates -----

A guide for this template, it's latest version, and all other templates are available at

[mplaza.pm/templates](http://mplaza.pm/templates)

Also, you may be interested in using our PRINCE2 eLearning Courses available at [mplaza.pm](http://mplaza.pm)

# Show of Hands - Group 7 - Project Brief

Copyright 2018, Management Plaza

You can use this document for free in your projects and for your personal purposes. Redistributing this document or using it for training requires permission from Management Plaza.

This document is based on AXELOS PRINCE2® material. Reproduced under licence from AXELOS. All rights reserved.