Case Studies Task 3 Subtask 3 Documentation

# Use of version control

We use GitHub as our management tool to control the storage space where we accessed our project, its files, and all the versions of its file. The link to our repository is:

GitHub repo: <https://github.com/TonksM/CSSD_eVoting>

# Requirements

In creating our test suite, we wanted to make sure that we covered all areas of testing such as unit acceptance of our voting system.

* R001 - The system must make sure that a vote can be recorded accurately
  + The system will process a vote and deposit in a database with no change to the vote
    - Tested in T1.1

Corresponding to requirement (2) above:

Test case 2.1 coverers requirement R002

* R002 - The system must ensure that a vote cannot be linked to a specific voter
  + The system will associate no voter information with the vote, and the identity of the voter cannot be determined
    - This will be tested in T2.1

Corresponding to requirement (6) above:

* R006 - The system must ensure that an account is secure against multiple failed attempts to login
  + The system will log the number of login attempts against the account in question
    - This will be tested in T3.1 where the login attempts against an account will be observed
  + The system, once the amount of login attempts has exceeded the prescribed maximum, the account should be locked down and an email should be sent to the email address of said account.
    - This will be tested in T3.2 where the inability to log in with the correct details is impossible until after the email reset
  + The user upon the receiving the email can click a link to unlock their account
    - This will be tested in T3.3 where the correct password will allow access into the account after password reset

# Testing Plan For Requirements

|  |  |  |
| --- | --- | --- |
| Test Case | Method | Expect Result |
| T1.1 - Vote is recorded accurately | 1. Navigate to localhost:3000 2. On login page, enter admin details and login:    * Email: [admin@email.com](mailto:admin@email.com)    * Password: qwerty 3. Click view under the results heading 4. Make note of results displayed 5. Click back, and then click logout 6. Enter details and click login:    * Email: [participant1@email.com](mailto:particpant@email.com)    * Password: qwerty 7. Click the first checkbox on the page, and click submit 8. On a successful vote, click logout 9. On login page, enter admin details and login:    * Email: [admin@email.com](mailto:admin@email.com)    * Password: qwerty 10. Click view under the results heading | The first party will have one more vote |
| T2.1 - No voter data is recorded with the vote | 1. Navigate to localhost:3000 2. On the login page, enter the following details and click login:    * Email: [participant2@email.com](mailto:participant2@email.com)    * Password: qwerty 3. Click the second checkbox on the page, and click submit 4. On successful vote, click logout 5. Open a new command window, and type: the command ‘mongo’ or the location of your mongo installation appending ‘./bin/mongo.exe’ 6. Run the command ‘use CSSD\_eVoting’ 7. Run the command ‘db.votes.find()’ | The command ‘db.votes.find() should show all votes cast on the system with no information on the voter who cast the vote |
| T3.1 - Account lockout after 3 failed attempts | 1. Navigate to localhost:3000 2. On the login page, enter the following details and click login:    * Email: [participant3@email.com](mailto:participant3@email.com)    * Password: password 3. Repeat this twice more (three times in total), and then enter the correct details:    * Email: [participant3@email.com](mailto:participant3@email.com)    * Password: password1 | Each time the incorrect password is entered, the message “Email and password combination is incorrect”.  Upon entering the correct password, a message will be displayed “Too many failed login attempts have been made, and this account is locked” |
| T.3.2  The system, once the amount of login attempts has exceeded the prescribed maximum, the account should be locked down and an email should be sent to the email address of said account. | Part of Individual enhancement for Ben Hastings (b5017778), testing for this test case is within their own testing document. |  |
| T.3.3  The user upon the receiving the email can click a link to unlock their account | Part of Individual enhancement for Ben Hastings (b5017778), testing for this test case is within their own testing document. |  |

# UI Evaluation

In approaching our UI evaluation, we decided not to over complicate the process and wanted to test the difference in how long it takes to cast a vote alongside the usability of casting a vote on two different platforms. We also decided to test the UI of our administration page, therefore have added tasks to be completed for the admin UI.When testing our UI, we wanted to involve tasks that are most common when attempting to vote, therefore we will be able to see what platform provides the best usability for the user alongside what platforms does casting a vote take longer on.

When carrying out our UI evaluation we wanted to set a parameter of a start and finish within our session so we could time the duration, starting from when they have logged in until the last step of logging off. We wanted to make sure we looked at feedback and recommendations surrounding the time taken to cast a vote on different platforms as this was something that stood out to us when investigating areas of concern for electron voting in subtask 2.

Another method of approaching our UI evaluation was how we was going to test the usability of our UI, therefore we wanted to the user to provide feedback in why a certain platform provided a better experience, alongside the integrity of our system by asking the user to try cast two votes.

We have designed our evaluation SUS questionnaire on asking the user about consistency between the different platforms as well as the consistency of the UI for each individual device. We want to have also addressed if they felt the UI held back the speed in which they completed casting a vote. From this we plan to make re design recommendation on improving the layout of our UI to improve usability and speed of casting a vote.

The reason in setting our UI evaluation out like this is to gain as much understanding on how the user is feeling during the process of casting a vote for each device, and how they felt about the overall timing of casting a vote. This allows us to make re design suggestions on if the consistency if UI is slowing the process for the user or if the platform gives a user a different experience, resulting in a more usable and overall smoother experience of our UI.

We also believe testing our UI on two different platforms gives us the ability to have a wider range of feedback based on how the UI performs and what it looks like across the vote process. With structured questions throughout each UI evaluation session we can capture how the user feels between the two platform devices during the session to get a better user experience understanding.

## UI Plan

### Scope

In considering the electronic voting system, we decided to cover the basic implementation of casting a vote. This consisted of logging in as a user, as we must assume the voter has already registered in which they have the correct user login details. Once logged in the user will then have a choice to log out or to continue their process of casting a vote. Once the vote has been submitted we will ensure to register the vote and not to store any information regarding who the voter voted for. Once a vote has been submitted the system then request the voter to confirm their vote. we cover the process to ensure confirmation of a vote, this is then followed by logging out and the user has completed the process of casting a vote.We also want to cover the UI from an admin point of view as although there is not a design difference between the admin page on a laptop to a mobile, however due to the monitor difference we are expecting a different experience.

1.Test logging in to the E Voting system.

2.Casting a vote on a ballot form and then confirming the vote.

3.Submitting the vote and checking the vote had been registered.

4.Test all of these for both Laptop and Mobile device platforms.

5. To be able view the voting results as a administrator.

### Purpose

While working on the electronic voting system and UI evaluation plan we started to Identify the concerns we initially had from our first design, we wanted to test if the users can perform these tasks on interfaces on different platforms. We wanted to see if any if of the users prefers one type of interface to the other. We will measure this on how usable the user found the UI to be as well as the time taken when casting a vote on two different devices, in which our goals for this evaluation is to find out if the user liked or disliked the interface. In each instance, we will have several general and several specific concerns to focus on that we will ask the user such as why they disliked it or liked it. Our concerns drive the scenarios we choose for the evaluation.

### Schedule & Location

We will be using the Sheffield Hallam university campus as our location.This will be conducted during the times from 9:00am – 5:00pm between 1st March – 15th 2019

Sessions

Once a user has sat down and is ready for the session to begin , the facilitator will begin to explain the project and reasons behind conducting the following tasks. Once the user is happy to continue with the session, they will be asked to read and sign a consent form, this will be explained to them that all details are anonymous. To begin the testing the User is allocate a laptop to test the systems UI’ first. The user is allocated time to perform all of the tasks for the first device. Feedback will be taken down after all the tasks for the first device is completed. Users will be asked to make comments on how they felt UI and how using the device impacted.The process will be repeated for the mobile device. Once all tasks and feedback is competed for both devices, The user is then provided with administrator details to complete the admin tasks.

1. Sessions will take 30 minutes to 1 hour.

2. Systems data has been preloaded on the system to simulate a real election

3. Sessions will begin with explanation of the project.

4. Users read information sheet and sign consent form.

5. Users are assigned a laptop to test the UI first.

6. Users perform all tasks for this device.

7. Feedback is taken for the first device.

8. Users perform all tasks for mobile interface.

9. Feedback is taken for the second device.

10. Participant then logs in as an admin.

11. User checks if they can view the voting results and then logs out.

12. Feedback is taken from the administrator task.

13. Users completes SUS form.

### Equipment

The type of equipment we will be using is a laptop and mobile device so we can display our system across 2 of the most used electrical devices. We are planning on screen casting the process with a laptop and screen recording for the mobile device.

1.Laptop.

2.Mobile device.

3.Screencasting software.

4.Stopwatch.

### Participants

We have five participants that we recruited around Sheffield Hallam university. The participants chosen was based around the same teaching rooms and cantor facility. We have included this documentation in consent forms.

### 

### 

### Scenarios

Scenario 1: Casting two votes as a single user on a Laptop.

Task 1: User logs in with the following email and password provided.

Task 2: User selects a party of their choice.

Task 3: User logs out and logs back in.

Task 4: User attempts to cast another vote.

Task 5: User logs out.

Scenario 2.Casting two votes as a single user on a Mobile device.

Task 1: User logs in with the following email and password provided

Task 2: User selects a party of their choice.

Task 3: User logs out and logs back in

Task 4: User attempts to cast another vote.

Task 5: User logs out.

Scenario 3. Logging in as administrator and check the voting results on laptop platform.

Task 1: login with the email and password provided.

Task 2: Click the view button under results.

Task 3: Make note if results are being displayed.

Task 4: Click the Back button.

Task 3: Click Logout.

### Subjective metrics

1. SUS questionnaire (SUS forms attached.)

### Objective metrics

1.Was The user able to login?

2. The ability to cast a vote.

3. Time taken on casting a vote.

4. Are votes recorded accurately?

4. Can Vote results be viewed via the admin page.

### Roles

1.Single person acted as facilitator and took some notes.

2.Screen recording was also conducted for documentation alongside feedback.

# Ethical Considerations

When considering our UI, we had ethical reasons that affected the outcome. We felt as a team we had positives and negatives from and ethical standpoint. When basing the design aspect of our UI we researched into the electoral law of the UK and started to base our design pattern upon this aspect.

This started with the colour scheme of our E Voting system, we had discovered that the use of any other colour than black or white could lead to allegations of trying to sway elections through the basis of what colour the ballot is. We addressed this by making a formal black and white layout, after further research we found several constituencies to have parties with a black or white colour theme in the middle east, however due to our system implementing an admin system in which gives a poll officer the power to make changes to ballot forms and details this can be addressed according to the circumstances at the time.

During our time designing the UI for our E Voting system we also found out how much we must limit information surrounding the party on the ballot form. This was due to ethical issues of trying to promote a party or constituency more than others based on providing more information. There we accommodated for this on controlling what type of information and how much information is displayed next to the party on the ballot form. To address the ethical issue at its root we decided to limit all information regarding a party to only its name and constituency. This therefore allows us not to worry about what information is being passed to the voter on the ballot form.

One major ethical consideration we had to take into place was how we are storing data of what voter voted for what party. Due to this being a main policy within voting systems, we wanted to make sure that one, we couldn’t view the data and two that if data from our system was ever stolen the hackers would not be able to view the information.

This allowed us to reflect on how we wanted to connect out backend and frontend systems together, we knew that if our front end displayed directly what our backend received we knew if we didn’t store anything in our databases then there was no risk of showing provide information.

# Evaluation Report

When carrying out our evaluation we wanted to make sure the user knew what they was taking part in as well as confirming what they was going to be doing, evan before we explained the whole project.

In carrying out the evaluation we split it into 4 sections so we could analyze and understand the feedback in a structured way. For example for each evaluation the participant went through a setup stage, this was so we could be assured each participant was testing out UI in the same environment. The second section was having a step by step guide that allows the user to follow each task with as little direction from the facilitator as possible.During the testing section we also have each participant under take the SUS questionnaire.

We then applied a feedback section in which this is used to address direct comments made by the participant. These comments are noted by the facilitator during the short interview after all of the task have been completed.

Lastly to consolidate our feedback we base our recommendations for re designing suggestions what the user felt needed to change as well as based of the results, such as the time taken on both platforms and from what they felt needed to be improved on the UI.

The following results are from each participant during our UI evaluation.

## Participant 1

Name: Anonymous

|  |  |
| --- | --- |
| **Setup** | 1. Participant 1 was told the sessions will take 30 minutes to 1 hour and received an explanation of the project. 2. User proceeded to read the information sheet and sign consent form. 3. User was randomly assigned a laptop or mobile device to carry out the test. 4. Screen casting for the laptop device or mobile device is started alongside the stop watch for when they begin to login. |
| **Testing** | 1. User asked to perform the first scenario and task 1 by entering the login details provided.    * (Email = Particpant1@email.com Password = Password1) 2. User is then presented with a ballot form. In which they proceed to complete by ticking a checkbox for the party they choose to vote for. 3. Participant 1 proceeds to finish casting his vote and then logs off. 4. User then proceeds to login again and cast another vote. 5. User is unable to vote twice and how now finished the evaluation by logging off. 6. User then proceed to perform all of the tasks again but for the second platform (Mobile Device). 7. User is then provided with Admin credentials    * (Email : [Admin@gmail.com](mailto:Admin@gmail.com) Password: Admin ) 8. User entered login details into the admin audit page. 9. User locates Vote total page. 10. User checks total votes. 11. The user then logs out. 12. Users complete SUS form. 13. Users are interviewed about their experience with the 2 platforms |
| **Feedback** | 1. User asked a question on step 2 of testing regarding where he could click if he didn’t want to vote even after logging on. 2. User made a positive comment on step 4 of testing about the scalable view on the mobile device, however some test on pages could of been bigger. 3. SUS questionnaire results provides positive feedback in the usability and speed of the casting a vote process 4. User preferred casting a vote on the mobile device than the laptop, they felt it was quicker and although some of the text was small, they felt the general usability of it was more suited to a mobile platform. |
| **Recommendations** | 1. Enlarge the back button on every page so the user knows they are able to go back at anypoint even if they haven't voted. 2. User was happy about scable view for the mobile device however suggest making areas larger as people with bad eyesight may suffer. 3. Making the mobile web version into an actual app instead of just website. |

### 

## 

## 

## Participant 2

Name: Anonymous

|  |  |
| --- | --- |
| **Setup** | 1. Participant 2 was told the sessions will take 30 minutes to 1 hour and received an explanation of the project. 2. User proceeded to read the information sheet and sign consent form. 3. User was randomly assigned a laptop or mobile device to carry out the test. 4. Screen casting for the laptop device or mobile device is started alongside the stop watch for when they begin to login |
| **Testing** | 1. User asked to perform the first scenario and task 1 by entering the login details provided.    * (Email = Particpant2@email.com Password = Password2) 2. User is then presented with a ballot form. In which they proceed to complete by ticking a checkbox for the party they choose to vote for. 3. Participant 2 proceeds to finish casting his vote and then logs off. 4. User then proceeds to login again and cast another vote. 5. User is unable to vote twice and how now finished the evaluation by logging off. 6. User then proceed to perform all of the tasks again but for the second platform (Mobile Device). 7. User is then provided with Admin credentials    * (Email : [Admin@gmail.com](mailto:Admin@gmail.com) Password: qwerty ) 8. User entered login details into the admin audit page. 9. User locates Vote total page. 10. User checks total votes. 11. The user then logs out. 12. Users complete SUS form. 13. Users are interviewed about their experience with the 2 platforms |
| **Feedback** | 1. User felt like overall layout on the laptop device looked simple and formal, compared to the mobile device being to empty. 2. Participant 2 made a comment in regards to the equal amount of time taken to cast a vote on both platforms, however based on the time taken results and his comments, it was longer to perform the admin tasks on the mobile phone due to the styling. 3. SUS questionnaire suggested user felt comfortable using the system. 4. Facilitator was asked for directions where to click to view the results of casted votes. 5. Participant 2 recommended not allowing them to log in again once voted, although he was not able to vote again. |
| **Recommendations** | 1. Looking into a redesign suggestion from point 1 of our feedback from participant 2 is the re-styling of the css for the mobile device would benefit the usability of our system on a mobile device. 2. Look into making the general size of every button and text larger to stop any issues of not being able to see anything clearly. 3. Recommend to not allow the use to be able to log back on again due to not being allowed to vote, however as we have it there to let the user know they have voted, we will not look to redesign this aspect of our system |

## 

## 

## 

## 

## Participant 3

Name: Anonymous

|  |  |
| --- | --- |
| **Setup** | 1. Participant 3 was told the sessions will take 30 minutes to 1 hour and received an explanation of the project. 2. User proceeded to read the information sheet and sign consent form. 3. User was randomly assigned a laptop or mobile device to carry out the test. 4. Screen casting for the laptop device or mobile device is started alongside the stop watch for when they begin to login |
| **Testing** | 1. User asked to perform the first scenario and task 1 by entering the login details provided.    * (Email = Particpant3@email.com Password = Password1) 2. User is then presented with a ballot form. In which they proceed to complete by ticking a checkbox for the party they choose to vote for. 3. Participant 1 proceeds to finish casting his vote and then logs off. 4. User then proceeds to login again and cast another vote. 5. User is unable to vote twice and how now finished the first task by logging off. 6. User then proceed to perform all of the tasks again but for the second platform (Mobile Device). 7. User is then provided with Admin credentials    * (Email : [Admin@gmail.com](mailto:Admin@gmail.com) Password: Admin ) 8. User entered login details and is displayed with the admin audit page. 9. User locates Vote total page. 10. User checks total votes. 11. The user then logs out. 12. Users complete SUS form. 13. Users are interviewed about their experience with the 2 platforms. |
| **Feedback** | 1. Participant made positive comments in regards to the speed of casting his vote on both the laptop and mobile device, stating they was both quick. 2. User made a comment in regards to how small some of the text was on the mobile device when casting a vote on the ballot form. 3. User felt the layout of the admin page was unique and looked good on the laptop platform, however felt some of the text on the admin audit page was to small on the mobile device. 4. SUS questionnaire suggests User felt the consistency between the mobile and laptop layout however saw inconsistency in styling between the both such as size of the text and button. |
| **Recommendations** | 1. Redesigning the text and button styling of the mobile device. 2. increase the size of styling for the admin audit page on mobile device. 3. Look into keeping the same styling size of text and buttons across both the mobile and laptop platform. |

## Participant 4:

Name: Anonymous

|  |  |
| --- | --- |
| Setup | 1. Participant 4 was told the sessions will take 30 minutes to 1 hour and received an explanation of the project. 2. User proceeded to read the information sheet and sign consent form. 3. User was randomly assigned a laptop or mobile device to carry out the test. 4. Screen casting for the laptop device or mobile device is started alongside the stop watch for when they begin to login |
| Testing | 1. User asked to perform the first scenario and task 1 by entering the login details provided.    * (Email = Particpant4@email.com Password = Password1) 2. User is then presented with a ballot form. In which they proceed to complete by ticking a checkbox for the party they choose to vote for. 3. Participant 4 proceeds to finish casting his vote and then logs off. 4. User then proceeds to login again and cast another vote. 5. User is unable to vote twice and how now finished the evaluation by logging off. 6. User then proceed to perform all of the tasks again but for the second platform (Mobile Device). 7. User is then provided with Admin credentials    * (Email : [Admin@gmail.com](mailto:Admin@gmail.com) Password: Admin ) 8. User entered login details into the admin audit page. 9. User locates Vote total page. 10. User makes checks total votes. 11. The user then logs out. 12. Users complete SUS form. 13. Users are interviewed about their experience with the 2 platforms. |
| Feedback | 1. User made a comment in regards to UI design being formal and tidy. 2. User prefered laptop device of mobile device due to size of text. 3. Participant made a comment that he did not feel the process was any quicker on each of the devices. 4. User made a comment in regards to the size of the check boxes was to small when casting a vote. 5. The user asked if he could update all address as once when in the admin audit. 6. The user was satisfied on how they was able to edit, add and remove objects with ease |
| Recommendations | 1. Keep the tidy and formal view of our UI while increasing the overall size of both text and buttons. 2. Redesign suggestion to have a bookmark listing on top of the voting page to allow the user what stage they are at when casting a vote. 3. The ability to update all chosen objects at once instead of having to update one object at a time. 4. Increase the overall size of check boxes when casting a vote. |

## Participant 5:

Name: Anonymous

|  |  |
| --- | --- |
| Setup | 1. Participant 5 was told the sessions will take 30 minutes to 1 hour and received an explanation of the project. 2. User proceeded to read the information sheet and sign consent form. 3. User was randomly assigned a laptop or mobile device to carry out the test. 4. Screen casting for the laptop device or mobile device is started alongside the stop watch for when they begin to login |
| Testing | 1. User asked to perform the first scenario and task 1 by entering the login details provided.    * (Email = Particpant5@email.com Password = Password5) 2. User is then presented with a ballot form. In which they proceed to complete by ticking a checkbox for the party they choose to vote for. 3. Participant 5 proceeds to finish casting his vote and then logs off. 4. User then proceeds to login again and cast another vote. 5. User is unable to vote twice and how now finished the evaluation by logging off. 6. User then proceed to perform all of the tasks again but for the second platform (Mobile Device). 7. User is then provided with Admin credentials    * (Email : [Admin@gmail.com](mailto:Admin@gmail.com) Password: Admin ) 8. User entered login details into the admin audit page. 9. User locates Vote total page. 10. User makes a note of total votes. 11. The user then logs out. 12. Users complete SUS form. 13. Users are interviewed about their experience with the 2 platforms. |
| Feedback | 1. User felt the logging in process took to log. 2. Participant felt the time taken was quicker on the mobile device than the laptop. 3. The user as an admin I wasn’t too sure if I was displaying the votes or it was just showing me results in general. 4. User felt like although a poll officer who is experienced in doing this would be fine, however as I don’t I feel there wasn’t much guidance 5. SUS from provided positive feedback in the consistency of our Ui and the usability of our form. |
| Recommendations | 1. Increase the overall text and button size for every component within the UI for the mobile device. 2. Have some kind of confirmation button once the party has been clicked on when casting a vote. 3. Maybe add colour to different pages that are not to do with casting a vote such as the login page and logout page |

## 

## 

## 

## 

## 

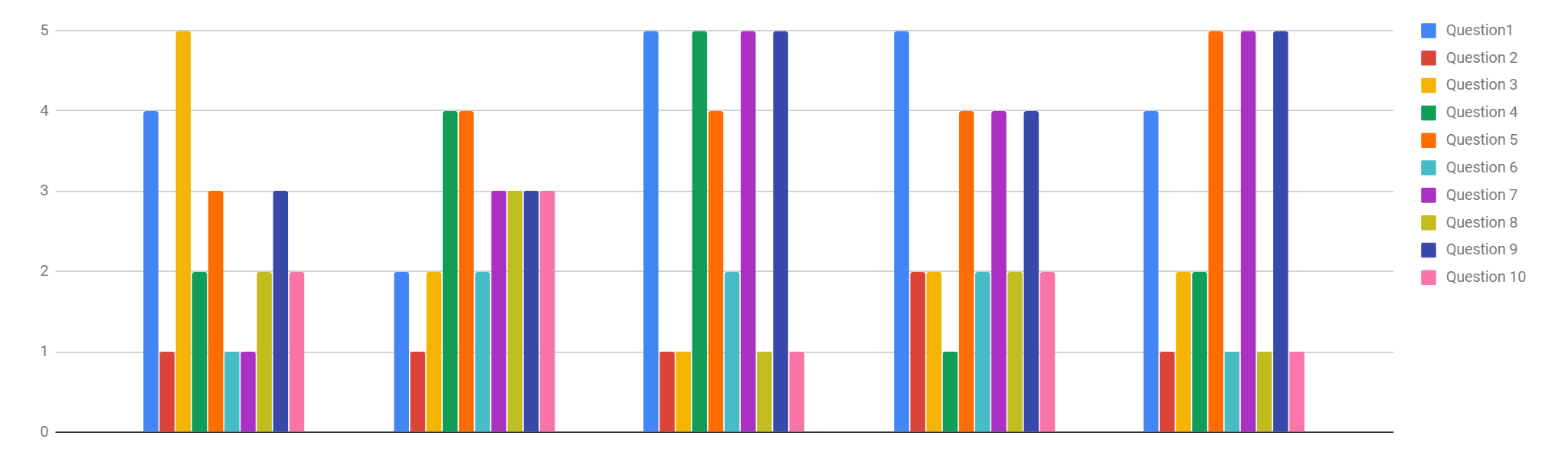
## 

## 

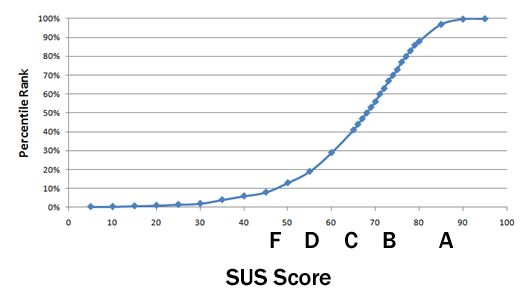
## SUS Results

With our SUS forms filled in, we generated SUS scores for each participant (fig.1) and then got an average score across all of them.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Participant** | **Q 1** | **Q2** | **Q 3** | **Q 4** | **Q 5** | **Q 6** | **Q 7** | **Q 8** | **Q 9** | **Q 10** | **SUS Score** |
| 1 | 4 | 1 | 5 | 2 | 3 | 1 | 1 | 2 | 3 | 2 | **70.0** |
| 2 | 2 | 1 | 2 | 4 | 4 | 2 | 3 | 3 | 3 | 3 | **52.5** |
| 3 | 5 | 1 | 1 | 5 | 4 | 2 | 5 | 1 | 5 | 1 | **75.0** |
| 4 | 5 | 2 | 2 | 1 | 4 | 2 | 4 | 2 | 4 | 2 | **75.0** |
| 5 | 4 | 1 | 2 | 2 | 5 | 1 | 5 | 1 | 5 | 1 | **87.5** |

*Fig 1 Tabled Results* *Fig 2 Graphed Results*

The average SUS score produced from our results was 72. This gives us a SUS rating of around a high C to a low B (as shown in figure 3 below ), which leaves us with room for improvements when it comes to the user experience.



*Figure 3 Our Average On SUS Graph*

### Timings of Participants

This table displays the results of the time taken for each participants to complete the first scenarios using both the mobile and laptop tasks.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Participant 1 | Participant 2 | Participant 3 | Participant 4 | Participant 5 |
| Laptop time | 2 minute 40 seconds | 3 minutes 22 seconds | 2 minute 33 seconds | 3 minutes 5 seconds | 3 minutes 12 seconds |
| Mobile time | 3 minutes 55 seconds | 4 minutes 30 seconds | 4 minutes 1 seconds | 3 minutes 55 seconds | 4 minutes 6 seconds |
| Difference in Time Taken | 1 minute 15 seconds | 1 minute 8 seconds | 1 minute 68 seconds | 50 seconds | 1 minute 34 seconds |

*Figure 4 Tabled Timings of Participants*

# Re-design suggestions

During our evaluation we investigated into the results that we have gathered from our findings inchich we have fount well-reasoned redesign recommendations.

One key suggestion during our evaluation was the consistency throughout our system, as a couple of users found the UI didn’t have clear narrative of where to go in areas of logging in and after a vote had been casted, feedback had showed that they felt a user may not know what to do during their time casting a vote unless you had done it before.

As a team we have recommend we take the feedback on board and would make changes to the UI of displaying a clear format of book markers in essence, for example once logged in, they would be on page 1/4 in which logging off would be 4/4, This suggestion can help users understand where they are within the voting system but also keep consistency of letting the user know what page they are on.

After noticing feedback regarding the colour of our UI being too bland as it only consists of black and white, although we have taken the recommendation into consideration we decided not to chance breaking any kind of electoral law. However, as we want to make the usability of our system the best we can, so we decided to look into how we could implement an option of changing colour schemes on for your voting process, although many users was not concerned about this , we also saw benefits for different types of colour blindness that we have tried to accommodate for with the black and white coloring.

Looking into our time taken results for the UI evaluation we between the amount of time to cast a vote for the laptop device and the mobile device we have found that although the time taken across all 5 participants was roughly the same we saw a increase in time taken for the mobile device. Based on our feedback and recommended suggestions we understand there needs to be a re design around the styling of text and button size for the mobile UI. Evan though among the feedback each user did not feel there was much of a difference in time taken when casting a vote, once the results had been compared we could see a valuable difference.

# Group Critical Reflection

## Detailed account of major Design Changes

Frameworks and language change

During development we determined that a NodeJS based web server would better suit our requirements than a more traditional object-oriented framework such as ASP.NET. Due to this we made the following changes:

* NodeJS allows for rapid development and deployment
* Express allows for easier code readability and a clear structure to our routers and views, as well as allowing us to use a templating framework such as EJS to help provide dynamic data
* MongoDB offered advantages in scalability and allowed us to be more flexible in the way that we both stored and retrieved data as NoSQL databases’ data structure is loosely coupled allowing for us to store data within JSON documents
* Mongoose, a database access middleware layer, that allows us to define document structure with the use of schemas to compensate for MongoDB’s inherit lack of relational data.

This language change to Node also made a lot of the methods and functions in our static diagram redundant. With the use of the Mongoose schemas do not require getters or setters unlike more traditional OO implementations as we could access the attribute we would require directly from the JSON object. These schemas also replace our classes as they hold all the data and some of the functionality of our classes from or original diagram.

### Changes in Design From Class Diagram to Implementation

#### Voter

The voter’s address was going to be stored as a string, however upon reflection of this, it should be of type Address and as such the function get Address should be of type Address

#### Candidate

No constructor was implemented in the design and no setters where present so had to add a constructor to class to set its values when an object of class Candidate is initialised.

No id variable in candidate, class GroupOfCandidates has function to get said ID.

#### Election

Constructor should have a group of Constituencies so a admin can create the group and add pass that group to the new election.

#### Ballot

The ballot is not stored in the server’s database as stated in our original design, however, is created at runtime, using the voter’s ID to query the database to find the voter’s address and then retrieving the constituency that address’ postcode belongs to using the *\_validPostcodes* field to determine which candidates the voter can vote for. This is all done within the ballot & proxy routers.

This has been altered due to the realisation that the ballot class would hold no unique data to itself and would be indistinguishable from the VoterUI when it came to implementation being composed of multiple aspects of our design.This realisation did not strike us during the design period due to us thinking of the design too physically, i.e. the ballot is an actual object when voting the traditional way

### Changes to User interaction

Ballot UI has changed from what was designed in the wireframes. This was due to the ballot not meeting our requirement of not meeting local electoral law and we had to change the design to closely match the physical ballot.

As a group we feel that the major redesign changes to our system was necessary, we felt the main benefit to the framework and programming language was that as as a collective we shared more knowledge in all areas of node.js and express with the use of mongodb more than ASP.net. We also felt we could all independently produce a more usable enhancement in having used an ejs engine for our front end compared to all but one group member lacking in the c# skill set.

Once we attempted the first framework and language we soon felt it was a larger task lerner certain aspects of the framework than it was to use a framework and programming language we all had experience in.

### Changes in Design From Storyboards & Wireframes to Implementation

When voting in an election, we designed it so the user is confronted with a confirmation page before making their vote, asking them if they want to continue with their vote, or if they want to go back and change it. For the implementation of vote we did not include this, instead implemented a two step process where you click who you want to vote for and then click submit. We thought this would make the flow of making a vote feel more streamlined.

When going through each stage of our voting system we planned to have an help button which would display a help description box , indicating what help you need for the page. We did not get these added in due to seeing the functionality behind it as a enhancement to the system.This is also the case for the soundbar, magnifying glass and brightness enhancements to our system.When looking back at why we did not this implemented, it is due to not planning what enhancements we should've chosen based on our original design specification.

Proxy voter has changed slightly from what was designed in the wireframes and storyboards. The change was to remove a page where the user can select between two buttons, to vote for themselves or to be taken to a page to select a proxy to vote for. We changed this to just show the associated voters they can vote on behalf of on that page, as well as their own account. This streamlines the process and accounts for scalability allowing for more than one associated voter per proxy account.

When an account is locked in our storyboards and wireframes, the user would be able to click the help text and see that an error message has been given if their account is locked. This has been removed so the error message is displayed on the screen instead of through a button click to make the process more streamlined and more evident to the user.

Looking into the admin Ui wireframe, we decided not to use the designed layout for all of the pages.We have kept the main admin audit page the same and is consistent with our original designs.However when editing a form we have decided to display all f the information on one page instead of tabs. We did this so that the admin would be able to see everything at once instead of having to manually go through each tab to only change one or more forms.

### Changes from Requirements to Implementation

Two requirements were not fulfilled in our implementation. These where:

1. The system should provide the ability to adjust system audio and visual elements such as screen brightness and sound level. (Usability requirement)
2. The response time of the system without any interaction after 4 minutes, has to automatically log out of the current account and re-display the login screen. (Performance requirement)

As discussed earlier in “Changes in Design From Storyboards & Wireframes to Implementation” we removed the users ability to listen to audio elements and change the brightness as we deemed it was more appropriate to focus on other areas for our individual contributions.

The requirement for logging out an user if they have stayed logged in for too long without any interaction was removed as we found from our testing that users were taking a similar amount of time to vote, some taking longer than 4 minutes. So this requirement would have made it an issue if users had to keep logging back in when they have decided on who to vote for.

### Changes from Sequence Diagrams to Implementation

Due to the change in planned framework and programming language, changes had to be made to certain function and method calls and because of this our software does not exactly match some of the dynamic diagrams mentioned in the design of the system, but do follow their basic flow of events.

#### Creating An Election

In the diagram for the “Creating An Election” there are a lot of calls to the election calls to set different attributes in it. These function calls in our built system are made redundant with Mongoose as the attribute them selves can be set through the schemas. The call for function ”getConsituencyById” is also redundant and is handled by the mongoose database call.

As discussed earlier with the ballot class, we removed it as we realised it was not needed, so the call to create a new ballot within this diagram has also been made redundant.

Despite these changes, the general flow of the sequence diagram is kept in the final build of the software, mainly being the set and get function being replaced with mongoose schema calls.

#### Voter Login

Login flow kept very similar to the original design, however some areas have changed. The increment login details function call has been moved from before the users password is checked to when a user’s password is not equal to the one in the stored in the database. We have done this as a user does not need their login attempts incrementing when they have got a valid login.

Calls to classes “”LoginUi” and “VoterUi” have been changed to routes which correspond to the login page and the ballot page of the web application.

# Appendix

SUS Questionnaire form………………………………………………………………………………….

Consent form………………………………………………………………………………………………

Fig1- SUS table results…………………………………………………………………………………..

Fig2- SUS Participant score graph………………………………………………………………………

Fig3- SUS score…………………………………………………………………………………………..

Fig4- Time casting vote for each device………………………………………………………………..