



# Project Status

The automated sign in system

# Table of Contents

|  |    |
|--|----|
| <b>1) Abstract</b>   | 3  |
| <b>2) Analysis</b>   | 4  |
| 2.1) Identification of Problem   | 4  |
| 2.2) Identification of Prospective Users                               | 5  |
| 2.3) Identification of Necessities and Acceptable Limitations          | 5  |
| 2.4) Data Sources and Destination                                      | 6  |
| 2.5) Data Volumes  | 7  |
| 2.6) Analysis Data Dictionary  | 7  |
| 2.7) Data Flow Diagram for Proposed Solution                           | 8  |
| 2.8) Numbered Objectives   | 10 |
| 2.9) Appraisal of Potential Solutions                                  | 10 |
| 2.10) Justification of Chosen Solution                                 | 11 |
| <b>3) Design</b>   | 13 |
| 3.1) System Design Overview  | 13 |
| 3.2) User Interface Design and Rationale                               | 14 |
| 3.3) Database Design and Data Requirements                             | 18 |
| 3.4) Validation and Verification Rules                                 | 19 |
| 3.5) Planned SQL Queries   | 21 |
| 3.6) System Security, Integrity and Robustness                         | 22 |
| 3.7) Algorithm Design and Rationale                                    | 23 |
| Hashing Algorithm  | 23 |
| Hash Verify Algorithm  | 25 |
| Check Location Algorithm   | 25 |
| Update a Status Algorithm  | 26 |
| Random String Algorithm  | 27 |
| Datatables Algorithm   | 27 |
| Access Level Check Algorithm   | 28 |
| Sign Out All Algorithm   | 28 |
| Show Password Algorithm  | 29 |
| Expand Form Algorithm  | 29 |
| Create Student Account (Object-Oriented Programming Version) Algorithm | 30 |
| Logout Algorithm   | 35 |
| Emergency Algorithm  | 35 |
| 3.8) Product Testing Strategy  | 36 |
| <b>4) Testing</b>  | 38 |
| 4.1) Test Plan   | 38 |
| 4.2) Test Results (linked to test plan)                                | 40 |
| 4.3) Post Testing Analysis and Fixes                                   | 58 |
| <b>5) Appraisal</b>  | 60 |
| 5.1) Objective Fulfilment  | 60 |
| 5.2) User Feedback   | 62 |
| 5.3) Proposed Future Revisions and Feedback Analysis                   | 63 |

|                                  |           |
|----------------------------------|-----------|
| 5.4) Conclusion .....            | 64        |
| <b>6) Appendix .....</b>         | <b>66</b> |
| 6.1) File Structure .....        | 66        |
| 6.2) Code with COMMENTS .....    | 67        |
| hash.php .....                   | 67        |
| hashverify.php .....             | 68        |
| connectDB.php .....              | 69        |
| footer.php .....                 | 69        |
| headernav.php .....              | 69        |
| styles.css .....                 | 74        |
| switch.css .....                 | 76        |
| scripts.js .....                 | 78        |
| 401.html .....                   | 78        |
| 404.html .....                   | 79        |
| 500.html .....                   | 80        |
| authenticate.php .....           | 82        |
| connect-mysql.php .....          | 84        |
| createStaffOOP.php .....         | 84        |
| createStudentOOP.php .....       | 89        |
| dashboard.php .....              | 96        |
| deletehistory.php .....          | 99        |
| deleteStaffOOP.php .....         | 101       |
| deleteStudentOOP.php .....       | 105       |
| emergency.php .....              | 110       |
| layout-sidenav-light.html .....  | 112       |
| login.php .....                  | 115       |
| logout.php .....                 | 119       |
| password.html .....              | 119       |
| staffAccounts.php .....          | 121       |
| statusHistory.php .....          | 128       |
| studentAccounts.php .....        | 130       |
| students.php .....               | 140       |
| updateStatus.php .....           | 143       |
| websigninout.php .....           | 145       |
| overnight.php .....              | 147       |
| signinout.php .....              | 148       |
| open screen .....                | 150       |
| initialise .....                 | 151       |
| signed_in .....                  | 152       |
| 6.3) Interface Screenshots ..... | 153       |
| 6.4) Project Resources .....     | 156       |
| 6.5) Database DDL/Schema .....   | 156       |

## 1) Abstract

Within this project, I intend to create a mobile application for tracking whether sixth form student are in school or not. The system will be automated and will sign the students in and out of their school. It will store the status of each student dependant on whether their chosen mobile device is on site. There will also be a database management web application for the school staff. This will have the functionality to manage the whole project and will be protected by a secure login system.

## 2) Analysis

### 2.1) Identification of Problem

Today, the use of paper registering systems is very outdated. However, this is currently Highcliffe School's only method of monitoring whether their Sixth Form students are on-site. It is also a potential safety concern, not being able to tell immediately which students are on-site and which are not. I have been asked to design a technological system to monitor the students.

This will result in a less time-consuming system to sign in and out. Understandably, it is a tiring and annoying task to have to sign in and out of Sixth Form on paper. I undertook some analysis to measure and represent the length of time it currently takes students to sign in and out of the Sixth Form campus between 08:30 and 09:30 on a Monday morning, the busiest time of the week. These are the results:

| Time Taken to Sign In (seconds) |     |     |     |     |     |     |     |     |     |
|---------------------------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 297                             | 296 | 261 | 296 | 220 | 255 | 281 | 307 | 292 | 151 |
| 261                             | 270 | 252 | 211 | 259 | 215 | 280 | 204 | 228 | 283 |
| 180                             | 283 | 187 | 214 | 170 | 325 | 217 | 237 | 300 | 232 |
| 295                             | 250 | 217 | 261 | 173 | 291 | 288 | 145 | 290 | 283 |
| 172                             | 322 | 278 | 192 | 245 | 241 | 255 | 266 | 187 | 193 |
| 261                             | 241 | 265 | 232 | 290 | 230 | 235 | 158 | 184 | 287 |
| 218                             | 171 | 223 | 201 | 345 | 214 | 230 | 294 | 249 | 216 |
| 266                             | 292 | 256 | 339 | 222 | 240 | 186 | 280 | 253 | 274 |

$$\bar{X} = \frac{\sum X}{N}$$

From this data we can calculate the mean to be **245.75** using the above formula

$$\sigma = \sqrt{\frac{\sum (x_i - \mu)^2}{N}}$$

$\sigma$  = population standard deviation

$N$  = the size of the population

$x_i$  = each value from the population

$\mu$  = the population mean

and calculate the standard deviation to be **45.3**.

This means our variable of time taken to sign in can be modelled with the normal distribution.

"A normal distribution is a probability distribution used to model phenomena that have a default behaviour and cumulative possible deviations from that behaviour." ("Normal distribution - Wikipedia") – used in A level Mathematics

$$X \sim N(\mu, \sigma^2)$$

mean =  $\mu$

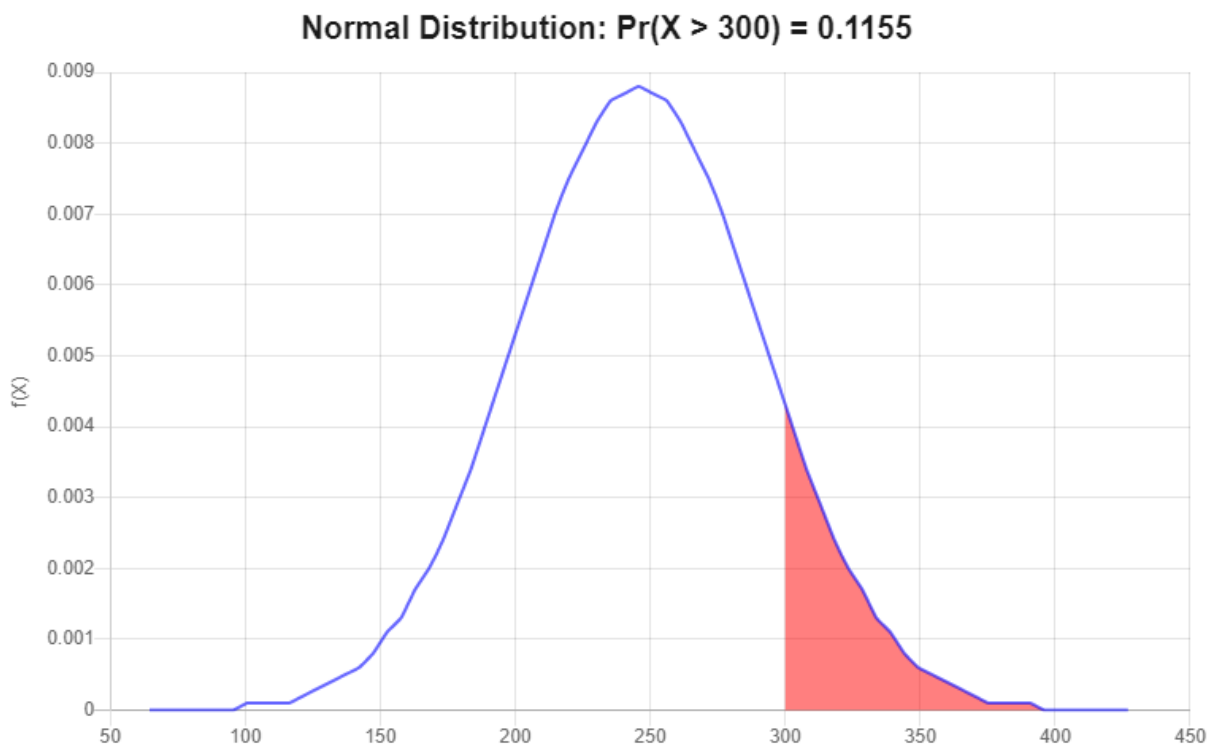
x has normal distribution

variance =  $\sigma^2$

Inputting our mean and standard deviation gives:

$$X \sim N(245.75, 45.3^2)$$

To find the probability that a student will have to wait more than 5 minutes, we find the area under our normal distribution's curve from the 300 second point onwards.



The area of the shaded region is 0.1155 meaning that there is a 11.55% chance, or roughly 1 in 10 students, that would take longer than 5 minutes to sign in. This is a high possibility for a student on a Monday morning. Creating my technical solution would negate this chance of missing class time.

## 2.2) Identification of Prospective Users

My system will primarily be used by Highcliffe Sixth Form. There will be two parts to the system: the student end, and the admin end. Students will not require much functionality as they only need to sign in and the app will do the rest. Students will solely be able to view their current status once initialising their app. However, the admin end is intended to be run by a group of IT professionals in conjunction with clerical assistants. This side of the system will have much more functionality, including viewing database tables of students and being able to manage or set up student accounts.

## 2.3) Identification of Necessities and Acceptable Limitations

So that I could identify the key elements required to be designed into my system, I asked both Highcliffe Sixth Form students and Sixth Form teachers for functions they would find crucial and discovered many others they would find helpful to improve their current method of signing in and out.

I first discussed their current system to uncover the most problematic parts. They said that their current method is using pen and paper, a very outdated procedure in their eyes. The staff's main issues with this are its safety issues; students cannot be quickly identified until the database is updated manually by a member of staff who checks the sign-in book every hour. Between the hour, if a fire were to occur, it could be unclear which students are still in the school and which aren't. They also added that not only is it a health risk, but it is also a very tiresome process not only for the CA updating the online database but also for the sixth form students. The students said that, due to only having one place to be able to sign in and out, it could sometimes take up to 5 minutes just to sign in if there was a queue. This is made more likely now they are no longer allowed to share pens due to Covid-19.

Obviously, the most crucial part of my system is the monitoring function: a student's phone being able to update whether they are on-site or off-site. This must be able to use the network to update a local database stored on the school's servers with an individual student's current status. It should be automatic, so that the students should not have to update their status manually and so that a staff member should not have to update the database manually. Also, the students' applications should remain logged into their student account each day.

After proposing my idea, staff responded saying that it would be particularly important for the database system to be easily accessible by authorised members of staff but secure from everyone else. An IT admin suggested it could be useful to involve two-factor authentication. They said that the information must be available readily enough that it can be used effectively in fire situations. Therefore, being able to order the database by tutor group would be necessary. The IT technicians added that maintenance functions would be very useful such as managing user accounts. Also, they said that, should the system fail for a particular student, they would see it essential to be able to update the student's status manually. They proposed that, should it be possible, a method of tracking student positions within school could be useful too.

They did however say that the system was not required to be functional between 16:30-07:30 each day. They also said that it was not necessary for a status to update instantly, checking every 15 minutes would be perfectly reasonable and would still maintain an acceptable safety aspect.

## 2.4) Data Sources and Destination

The majority of data in my system will be stored on the school's server and will be accessed regularly by members of staff on separate machines. Other than this data, a small amount will be required to be stored on each student's mobile device. And then, obviously, the status of a student will be frequently sent from the student device to the server.

| What is it?  | Source of Data   | Destination of Data                     |
|--|--|---|
| User Login Information<br><i>The login that a user will use when initialising their application.</i> | Used by user when they first set up their sign-in application. | Separate User Info table.               |
| Student Name/Admin Number  | From the school's records.                                     | As records in the main status database. |
| MAC address of Devices<br><i>The unique reference of each student's device.</i>                      | Stored at the point of initialisation.                         | As records in the main database.        |
| Students' Statuses<br><i>Whether each student is signed in or out.</i>                               | From the students' phones.                                     | As records in the main status database. |

## 2.5) Data Volumes

The data flow through this project will be recurrent every 15 minutes. Every student device will visit the signinout.php webpage on the school server once within this period. Since there are 300 students in the Highcliffe Sixth Form this would be 300 times that the server's URL is accessed. However, this should be no strain to the infrastructure of Highcliffe School as they have a large bandwidth a strong powerful central server.

Highcliffe School staff said they were happy for the system to sign every sixth former out at the end of each day and become non-operational until 07:30 on the next school day. This will not require any data transfer over the network as this can be an internal procedure on the server. The 300 records of student statuses will all be set to OUT which the server is more than capable of doing at midnight, since it will be under less stress anyway at this time.

The data which will be stored in the database per student will be: first name, surname, year group, tutor group, candidate number, admin number, cardID, and status.

| Field Name      | Length   | Size (bits)              |
|-----------------|----------|--------------------------|
| Firstname       | 6 (avg.) | 6x16(Unicode characters) |
| Surname         | 6 (avg.) | 6x16(Unicode)            |
| YearGroup       | 2        | 4                        |
| TutorGroup      | 3        | 8                        |
| CandidateNumber | 4        | 14                       |
| AdminNumber     | 6        | 20                       |
| CardID          | 11       | 11x16(Unicode)           |
| Status          | 1        | 1                        |

Therefore, the total bits stored per student is 415

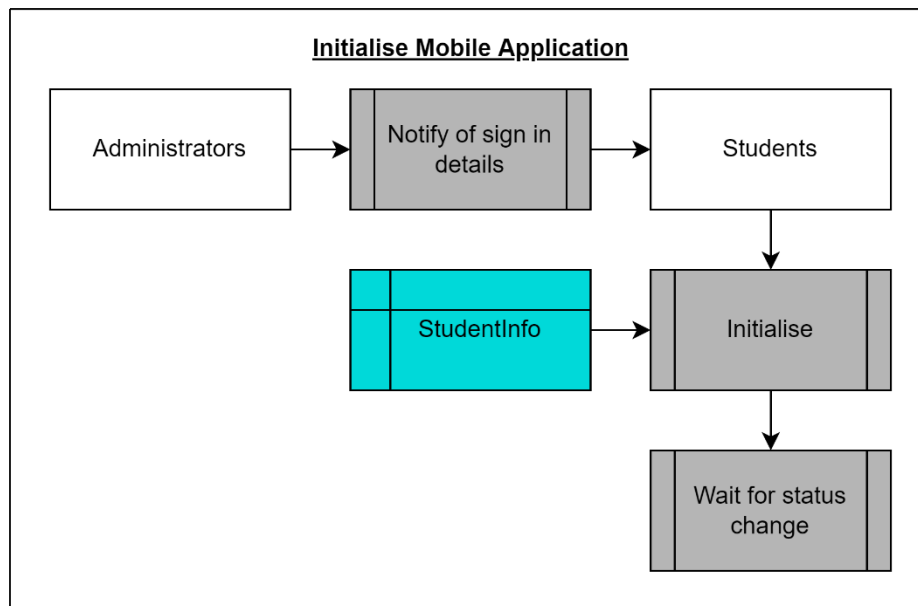
And so, the total data stored would be 415x300: **124500bits, 185563bytes, 15kB**

## 2.6) Analysis Data Dictionary

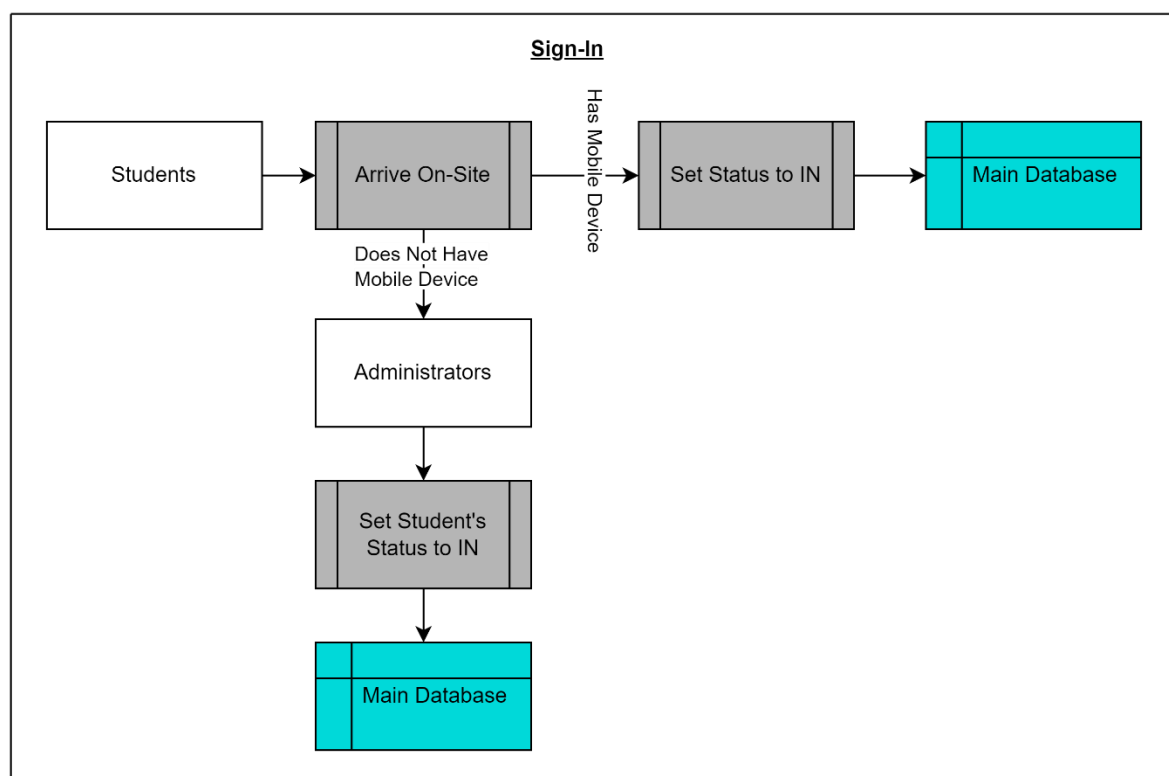
| Data            | What it is  | Data Type             | Example                    |
|-----------------|---|-----------------------|----------------------------|
| MAC             | Each student's device's unique reference code.                                  | Abstract (length: 12) | 2c549188c9e3, 67f7a5b8c697 |
| Status          | Whether a student is onsite or offsite.   | Boolean Binary        | 1,0                        |
| CandidateNumber | The students' candidate number (unique reference number assigned to each pupil) | INTEGER (length:4)    | 4681, 6872, 1564           |
| Firstname       | The students' first name  | STRING                | Charlie, Jack, Bob, Phil   |
| Surname         | The students' surname   | STRING                | Silver, Smith, Jones       |
| YearGroup       | The students' year group  | Integer (length: 2)   | 12,13                      |
| TutorGroup      | The students' tutor group   | STRING                | 12.6, 13.12, 12.3          |
| AdminNumber     | The unique admin number assigned to each student                                | INTEGER (length: 6)   | 513634, 302859, 284028     |
| CardID          | The random code used to authenticate the mobile app                             | STRING (length: 11)   | 19s9148s920                |



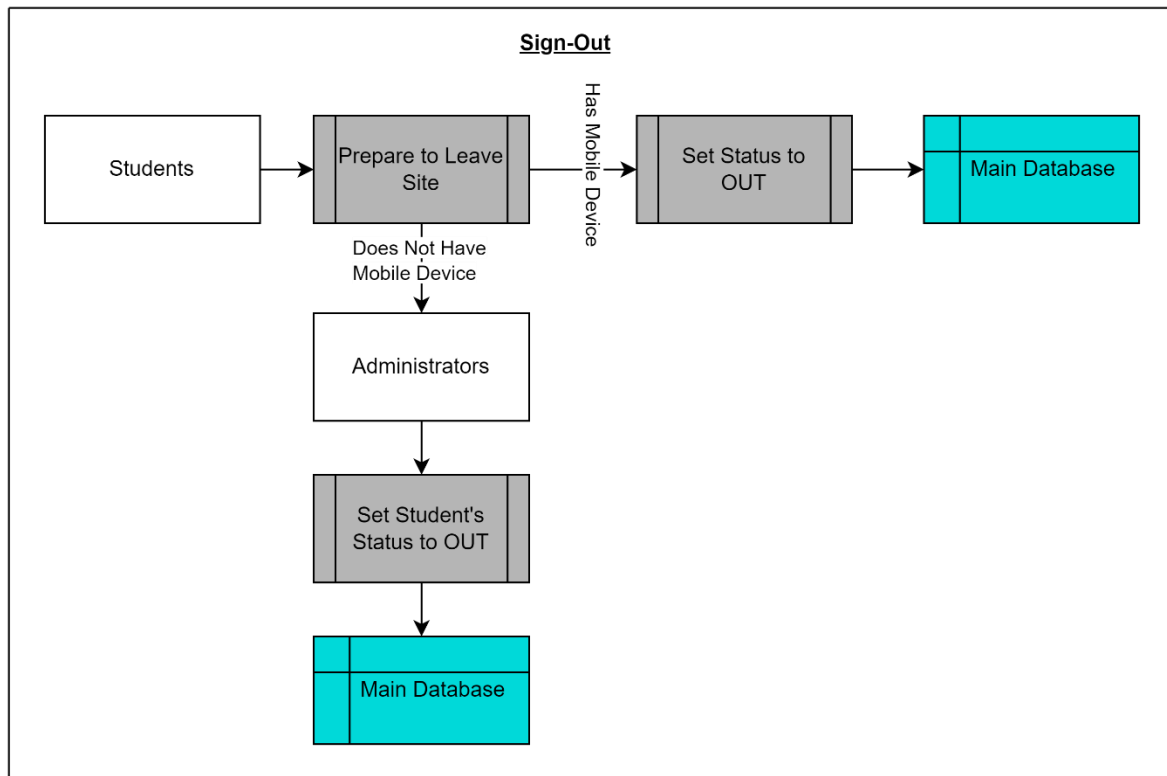
## 2.7) Data Flow Diagram for Proposed Solution



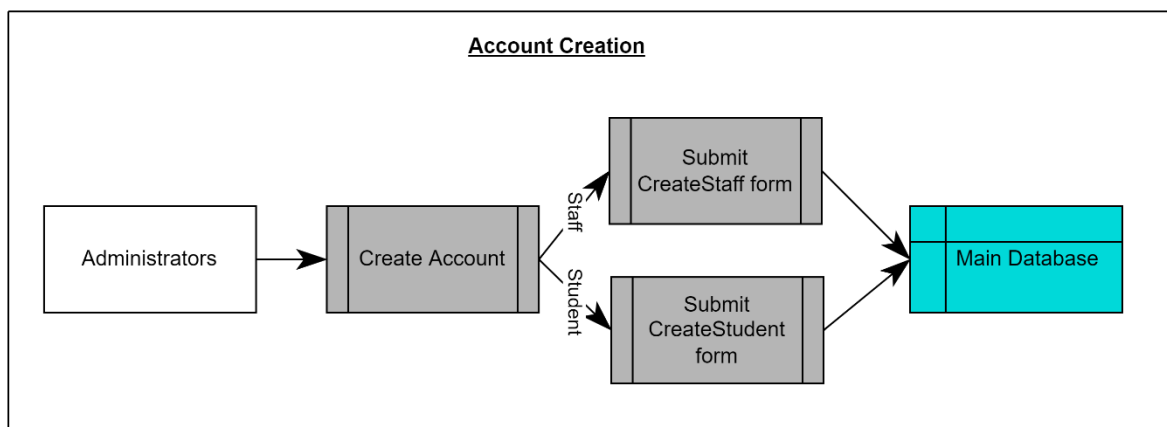
The administrator informs the student of their details (CardID, CandidateNumber) and then the student sets up their application with a verification check against the details held on the school server. After this has been complete, the app is fully ready and working.



When a student arrives on site, either their mobile application will update their status automatically or the small proportional who won't use the app can request their status to be updated by a member of staff.



When a student leaves site, either their mobile application will update their status automatically or the small proportional who won't use the app can request their status to be updated by a member of staff.



An admin can create either a new account for the students or create a new account for staff via the web application.

## 2.8) Numbered Objectives

| Client-Side Application |  |
|-------------------------|--|
| 1                       | Can be initialised easily and efficiently  |
| 2                       | Can only be setup for a specific student, students cannot falsely sign in another student  |
| 3                       | Once initialised, has no user functionality  |
| 4                       | Runs a location service with little load to the client device  |
| 5                       | Uses only small amounts of data and uses the fields: <ul style="list-style-type: none"> <li>❖ CandidateNumber</li> <li>❖ Status</li> </ul> |
| 6                       | Uses encryption to protect the fields when signing in and out  |
| 7                       | Updates location every 15 minutes and updates status if necessary  |

| Server-Side Application |   |
|-------------------------|---|
| 8                       | Maintains statuses of all students                  |
| 9                       | Resets temporary statuses at the end of each day    |
| 10                      | Keeps a record of each status change                |
| 11                      | Deletes old status changes at the end of each month |

| Admin-Side Application |   |
|------------------------|---|
| 12                     | Has a login system  |
| 13                     | Utilises Two-Factor Authentication  |
| 14                     | Has the ability to update an individual student status  |
| 15                     | Can display different databases depending on access level   |
| 16                     | Can sort the database by different fields: <ul style="list-style-type: none"> <li>❖ Tutor Group</li> <li>❖ Year Group</li> <li>❖ Surname</li> <li>❖ Status</li> </ul>   |
| 17                     | Has functions to manage user accounts   |
| 18                     | Has an emergency function <ul style="list-style-type: none"> <li>❖ Collates all students who are supposed to be on site</li> <li>❖ Compiles them into a CSV file</li> <li>❖ Downloads the file to the client</li> </ul> |

## 2.9) Appraisal of Potential Solutions

A solution to the problem could be hiring a company or programmer to create a solution. However, this would be an expensive method as the functionality required for this project is not generally offered in any commercial products currently. Therefore, we would have to approach a private company to create a bespoke product. This would maintain a high security standard and high reputability also. This would also potentially mean the project could be completed faster as a whole team could work on it. It might not be a valid option though as the Highcliffe Sixth Form's budget is not very large understandably as a school. Also, it might be easier to collaborate with an internal team as this provides simpler communication and is more likely that the project will be suited to the intended use.

Therefore, an alternative option is that I programme myself. I could create a server, mobile, and web application. This would mean that I could test the programme using Highcliffe School's actual network meaning guaranteed results. I would need to create a mobile application for IOS and

Android as these are the main two operating systems that the sixth form students' mobile devices use. These devices would have to be able to use location services and discern whether the students are onsite or not. Then they would have to communicate their status to the school's server using the network. The server will keep records which could then be viewed via the web app by school staff.

Or instead, I could create a system whereby the students update their status manually via a website however this could lead to several problems. Many students may forget to update their status, hence why using an automatic system is perhaps better. Also, this could open the option for students to update their status from anywhere; they wouldn't have to be on site to update their status and make it appear as if they are. They could also potentially update the status of another person which again raises several problems.

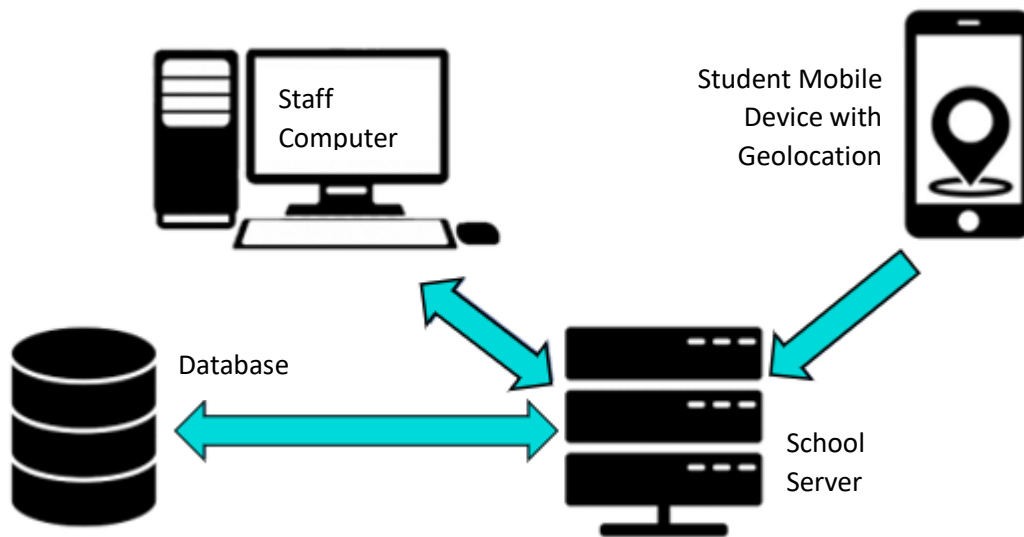
## 2.10) Justification of Chosen Solution

I have decided that the second option, programming the system myself, is the better choice. This is due to a range of factors including that the school does not have the budget to hire an outside team. I will use programming languages such as PHP, JavaScript, HTML, SQL, and potentially Python to create the applications necessary to make the programme run. I will use software environments such as Microsoft Visual Studio, Adobe Dreamweaver, and Notepad. I will also use phpMyAdmin and XAMPP to aid me. I will create a mobile application which checks, using location services, whether the device is within Highcliffe School and updates the status of the student who owns the device accordingly. I will also create a web application so that site administrators can view and update the statuses of students as well as manage accounts.



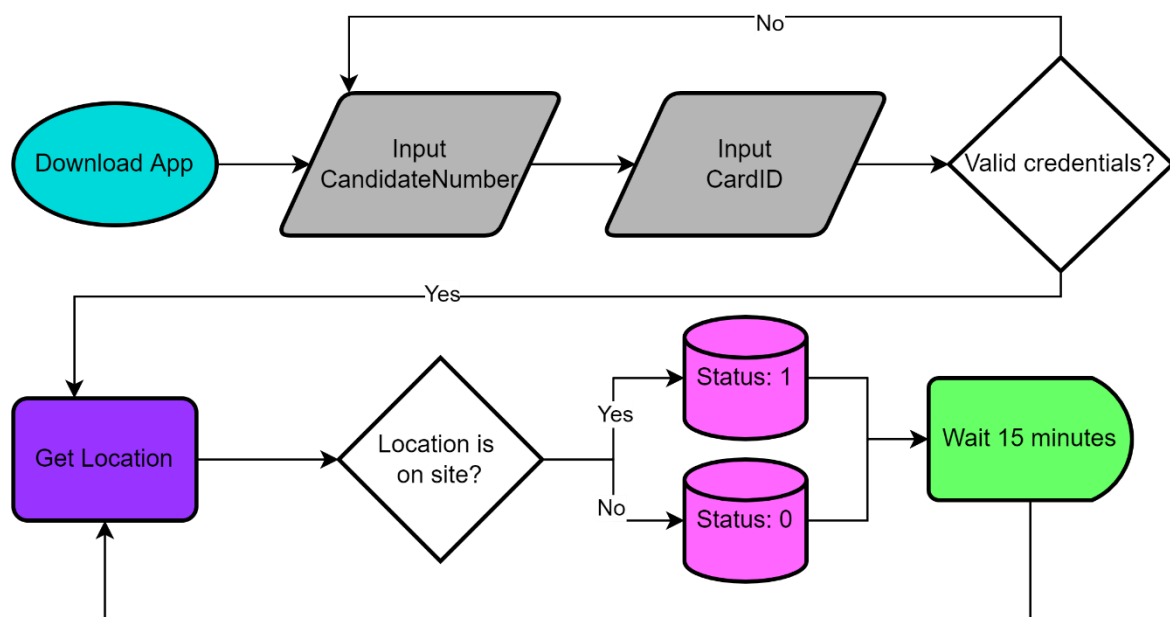
### 3) Design

#### 3.1) System Design Overview

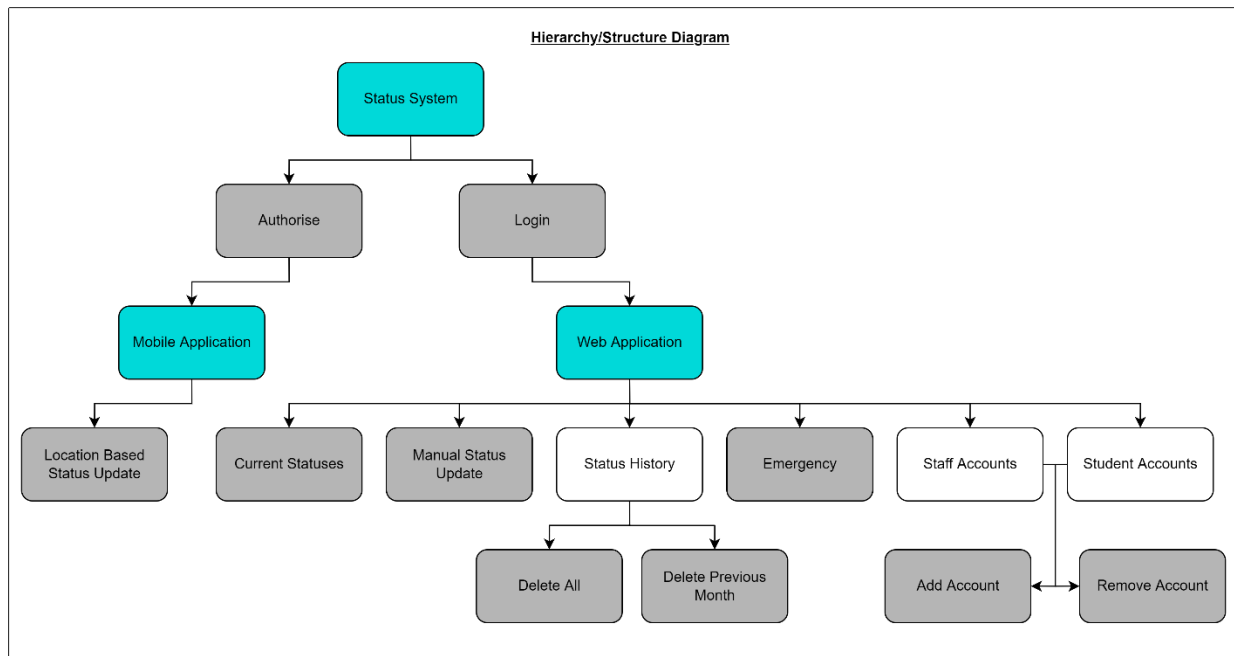


This diagram visualises a simplified model of my system. The server is the central point, with access to update and read the database. The web application can always request from the server and, when signed in as an admin, can also send to the server. The mobile application can always send updates to the server.

Below is a flowchart demonstrating the design of the mobile application.



Below is a Structure Diagram for my system.

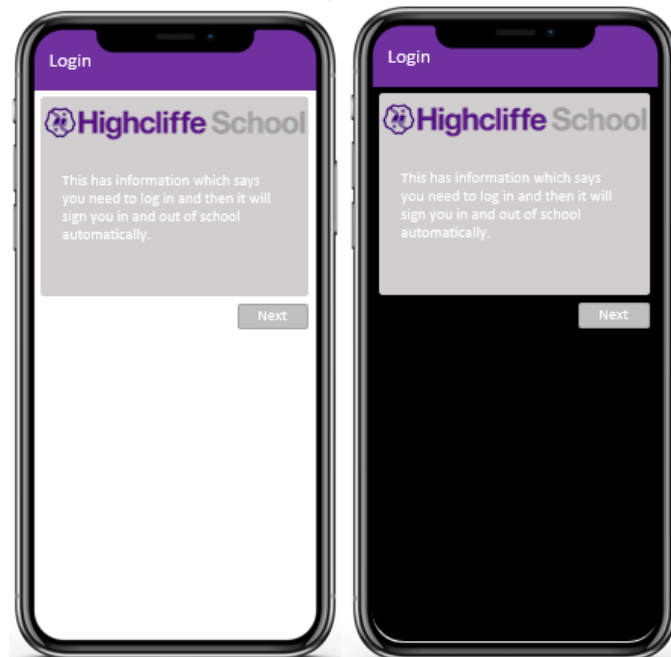


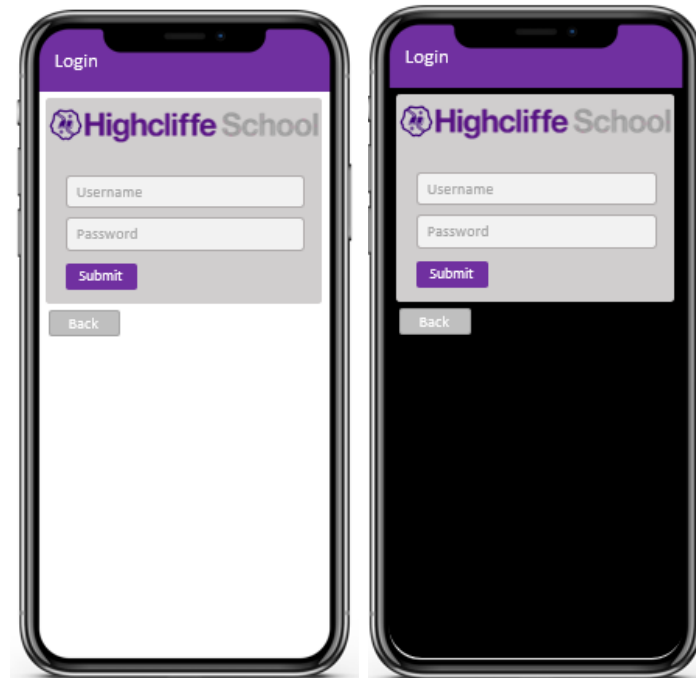
### 3.2) User Interface Design and Rationale

My user interface will be in two sections: The mobile application for the students, and the web application for the staff. The most obvious choice for the colour scheme of my applications would be the Highcliffe School colour purple (#522a5b). This colour pairs very nicely with different shades of grey.

#### Mobile Application

When creating the mobile application, the main objective was to build something self-explanatory and easy to use. All the function of the mobile app is passive and does not require any user input other than the original two fields. For this reason, a more aesthetic user interface could be created.









### Web Application

The web application has a lot more user functionality than the mobile application and therefore there arose a need for a higher weighting on the ergonomic aspect of the application. For this reason, I would choose to have a convenient toggleable sidebar on the left to easily allow the user to select each webpage. The use of the jQuery plugin Datatables will be especially useful, providing a clear table format, which is easily sortable and adaptable. I also plan to use the popular icon set FontAwesome to provide memorable symbols throughout my website.

Highcliffe School

Login

Username

Password

☐ Show Password

Submit

Highcliffe School

Search

Subtitle

Page 1

Page 2

Page 3

Page 4

Subtitle

Page 1

Page 2

Page 3

Page 3

Subtitle

Page 1

Page 2

Page 3

Page 4

Page 5

Dashboard

This is the dashboard

Shortcut 1

Shortcut 2

Shortcut 3

Shortcut 4

Signed in: 182

Signed out: 127

Username

Copyright n that

Terms and conditions

Highcliffe School

Search

Subtitle

Page 1

Page 2

Page 3

Page 4

Subtitle

Page 1

Page 2

Page 3

Page 3

Subtitle

Page 1

Page 2

Page 3

Page 4

Page 5

Student Statuses

This is the statuses

Signed In

| Candidate Number | Surname | Firstname | Tutor Group | Year Group | Time Changed | Status |
|------------------|---------|-----------|-------------|------------|--------------|--------|
|                  |         |           |             |            |              |        |
|                  |         |           |             |            |              |        |
|                  |         |           |             |            |              |        |
|                  |         |           |             |            |              |        |
|                  |         |           |             |            |              |        |

Username

Copyright n that

Terms and conditions

### 3.3) Database Design and Data Requirements

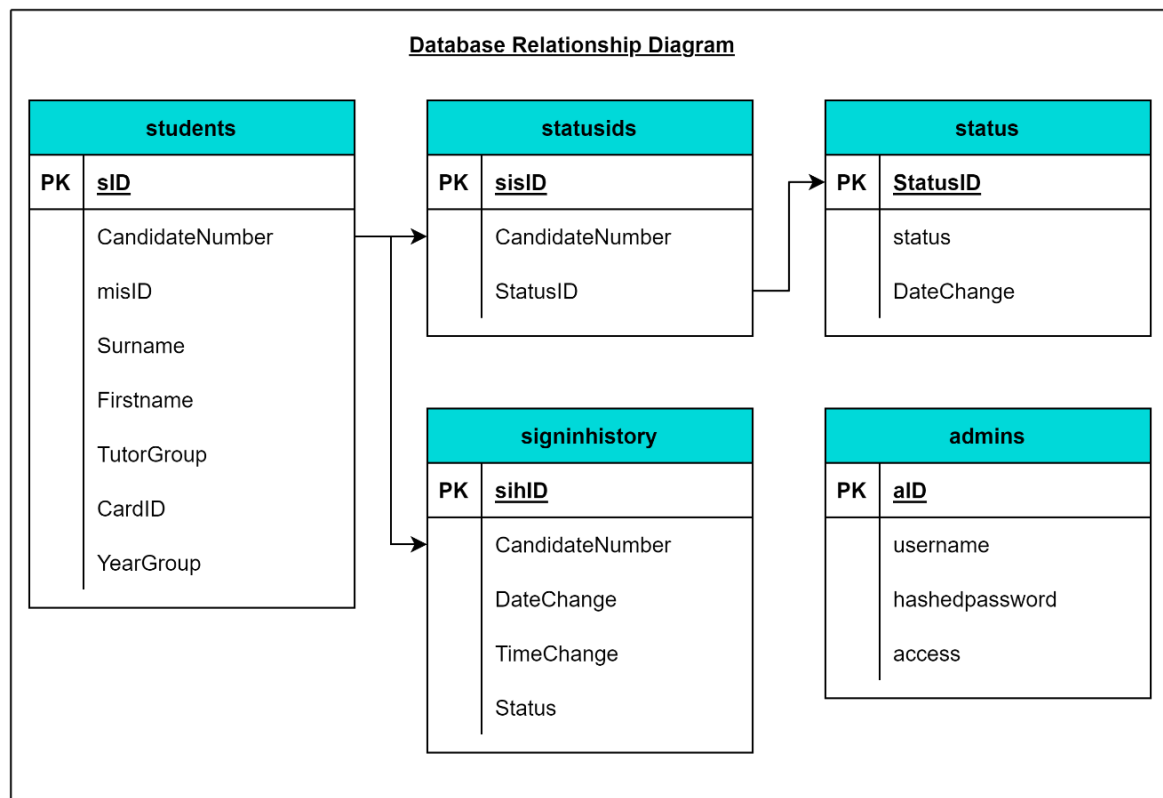
| <b>admins</b>  |             |         |   |
|----------------|-------------|---------|---|
| Name           |             | Type    | Description   |
| alID           | Primary Key | Int(11) | An autoincrementing key   |
| username       | Unique      | Text    | The username of the staff account   |
| hashedpassword |             | Text    | The value created by the hashing algorithm, used so passwords aren't stored in blank text |
| access         |             | Text    | Either an "A" for admin or an "S" for staff.  |

| <b>signinhistory</b> |                              |           |   |
|----------------------|------------------------------|-----------|---|
| Name                 |                              | Type      | Description   |
| sihID                | Primary Key                  | Int(11)   | An autoincrementing key   |
| CandidateNumber      |                              | Int(4)    | The candidate number of the student signing in or out                       |
| DateChange           | Default: Current_timestamp() | Date      | The date of the status change   |
| TimeChange           | Default: Current_timestamp() | Time      | The time of the status change   |
| Status               |                              | Binary(1) | Either a 1 for signed in or a 0 for signed out                              |
| Type                 |                              | Text      | Whether the status change was due to mobile app, manual, or staff on webapp |

| <b>status</b> |                                |           |  |
|---------------|--------------------------------|-----------|--|
| Name          |                                | Type      | Description                                    |
| StatusID      | Primary Key                    | Int(11)   | An autoincrementing key                        |
| status        |                                | Binary(1) | Either a 1 for signed in or a 0 for signed out |
| DateChange    | On update: current_timestamp() | Timestamp | The datetime of the status change              |

| <b>statusids</b> |             |         |  |
|------------------|-------------|---------|--|
| Name             |             | Type    | Description  |
| sisID            | Primary Key | Int(11) | An autoincrementing key                              |
| CandidateNumber  | Unique      | Int(4)  | The candidate number of the student                  |
| StatusID         | Unique      | Int(11) | The statusID which is used to join to `status` table |

| students        |             |            |   |
|-----------------|-------------|------------|---|
| Name            |             | Type       | Description                               |
| sID             | Primary Key | Int(11)    | An autoincrementing key                   |
| CandidateNumber | Unique      | Int(4)     | The candidate number of the student       |
| misID           | Unique      | varchar(6) | The admin number assigned to each student |
| Surname         |             | Text       | Student surname                           |
| Firstname       |             | Text       | Student first name                        |
| TutorGroup      |             | Text       | Student tutor group                       |
| CardID          | Unique      | Text       | An 11 character randomly generated value  |
| YearGroup       |             | Int(2)     | Student year group                        |



### 3.4) Validation and Verification Rules

With such a data heavy system, there is a need for prevention of unintended data. My system will feature several PHP forms and therefore I can access built in functions used to format them, taking full advantage of both HTML bootstrap functions and PHP functions. Utilising bootstrap forms ensures that I can check data before the form is submitted. My forms will be submitted using the POST protocol, which ensures that data is encrypted between each web location visited when using HTTPS. I would ensure that the secure version of HTTP with SSL Security Certificate was used should my system go into production. This way the packets are protected. It will also open the ability to access the PHP \$\_POST variables ensuring that data from forms is easily checkable and manipulatable even after the form has been submitted.

Data needing to be validated can be split into three main categories: Malicious, Incorrect, and Improperly Formatted.

#### Malicious Data

To prevent malicious data being submitted my database, I will use a cleansing algorithm on each occasion when an SQL insert is required. The intention is to strip special characters into their ASCII representation. This way no code can be injected into my system. My algorithm in PHP would look something like this:

```
function test_input($data){  
    $data = trim($data);  
    $data = stripslashes($data);  
    $data = htmlspecialchars($data);  
    return $data;  
}
```

#### Incorrect Data

To prevent incorrect data being submitted to my database, I will ensure that only an administrator can submit forms which require an “SQL insert” rather than an “SQL replace”. This way if incorrect data is accidentally submitted, the system administrators are likely to realise and will be competent to rectify the mistake in the database. It could be possible to accidentally use the delete account form for example, therefore I would create a JavaScript function to double check that the user wishes for the form to be submitted. It would look something like this:

```
onclick="return confirm('Are you sure you want to delete this?');"
```

#### Improperly Formatted Data

To prevent improperly formatted data being submitted to my database, I will use bootstrap “input type”. This way, when entering a student’s tutor group for example, I can use the dropdown option so that only certain choices are available (the tutor groups). To guarantee all data is collected before the form is submitted, the “required” tag will prove very useful. Also using regular expression to check if an input contains a special character and checking that enough characters are present will be important when creating passwords. My conditions in PHP would look something like this:

```
if(preg_match('/[\`^£$%&*()]{@#~?><>,|_+~-' }', $password){  
    Contains special character  
} else{  
    Doesn't contain special character  
}  
  
if (strlen($password)>=8){  
    Password is Long enough  
} else{  
    Password isn't Long enough  
}
```

### 3.5) Planned SQL Queries

| SQL Syntax  |  |             |   |
|-------------|--|-------------|---|
| SQL Command | Effect   | SQL Command | Effect  |
| SELECT      | Selects specific records from the table that meet certain criteria | OR          | For a match, one or more of the criteria is required to be fulfilled    |
| FROM        | Identifies which table the records will come from                  | DELETE      | Used to delete data from a database table                               |
| WHERE       | Adds the criteria that returned records need to meet               | INSERT INTO | Used to add data to a database table                                    |
| COUNT       | Used to count the number rows returned by a SELECT statement       | JOIN        | Allows selection data from multiple tables at once                      |
| AND         | For a match, both criteria are required to be fulfilled            | UPDATE      | Allows records to be updated – used in conjunction with the SET command |

To link to my database, I will use the MySQLi extension in a php function:

```
function connectDB(){
    $servername = "localhost";
    $username = "username";
    $password = "password";
    $dbname = "database";

    $conn = new mysqli($servername, $username, $password, $dbname);
    if($conn->connect_error){
        die("Connection failed: " . $conn->connect_error);
    }
}
```

The majority of my web application will be displaying tables using Datatables (a plug-in for the jQuery JavaScript library). Using php to fill the table, I would use:

```
$sql = "SELECT * FROM `table`";
$result = mysqli_query($conn,$sql);
while($row = mysqli_fetch_array($result)){
    echo "<tr>";
    echo "<td>" . $row['rowname1'] . "</td>";
    echo "<td>" . $row['rowname1'] . "</td>";
    echo "</tr>";
}
```

Also, I will have to create new accounts, both student and staff, using SQL inserts. As creating a student account involves inserting into several tables, I would use an SQL transaction to ensure that all inserts are possible before commitment. Using php OOP, I would use:

```
$conn->autocommit(false);
$sql = "INSERT INTO `students` (`CandidateNumber`, `misID`, `Surname`, ...)
VALUES ('$this->candidateNumber', '$this->adminNum', '$this->surname' ...)";
$conn->query($sql);
```

```
$sql2 = "INSERT INTO `statusids` (`CandidateNumber`, `StatusID`)
VALUES ('$this->candidateNumber', '$this->statusID');
$conn->query($sql2);
$sql3 = "INSERT INTO `status` (`StatusID`, `Status`)
VALUES ('$this->statusID', '0');
$conn->query($sql3);
$conn->commit();
```

Another major part of my system will be the updating of student statuses. I will have to locate the StatusID affiliated with the student's candidate number and use that to update the table `status`. Using php, I would use:

```
$sql = "SELECT * FROM `statusids` WHERE CandidateNumber=$dbCandidateNumber;";
$result = $conn->query($sql);
$row = mysqli_fetch_array($result);

$dbStatusID = $row['StatusID'];
$sql = "REPLACE `status` (`StatusID`, `Status`)
VALUES ('$dbStatusID', '$dbStatus')";
$conn->query($sql);
```

And each time a status is updated into table `status`, a new record needs to be created into `signinhistory`. I would use:

```
$sql = "INSERT INTO `signinhistory` (`CandidateNumber`, `Status`, `Type`)
VALUES ('$dbCandidateNumber', '$dbStatus', 'type');";
$conn->query($sql);
```

A smaller part that would also be required would be to count the total number of students signed in and out. This could be done very simply in php with something like:

```
$sql = "SELECT status, COUNT(*) as count FROM status
WHERE status = 1
GROUP BY status;";
$result = mysqli_query($conn, $sql);
```

### 3.6) System Security, Integrity and Robustness

All data in my system will be stored in a secure SQL database. This database cannot be remotely accessed without authorisation meaning it would be much harder for an unwanted user to gain entry. The server is located in a safe location on-site at Highcliffe School meaning that no one would be able to gain access manually as it is locked away behind multiple doors. I could also disable remote access to the server seeing as the only times a validated user should require access would be when they are onsite. Limiting access to only devices with the same IP address as the server will prevent unwanted users from gaining entry. Using this server has many robustness and security benefits too. The server is securely backed up regularly both onsite and offsite meaning no data would be destroyed in the case of an emergency. The server also has backup power so, in the case of a power cut, the system would still be fully operational.

The choice to use PHP for the bulk of the system was in a conscience effort to seamlessly integrate with SQL. PHP (Hypertext Preprocessor) is a server-side general-purpose scripting language that is especially suited for web development and can be embedded into HTML. For this reason, through the use of libraries/extensions such as MySQLi, it is the perfect choice for my system. Using PHP ensures that, since all processing is rendered on the server before the webpage is displayed on the client, my website will function correctly on all browsers.

Security of data is crucial in my system, particularly when submitting forms. For this reason, I have chosen to ensure that the \$\_POST method is used rather than the \$\_GET method. Although the \$\_GET method is more simplistic and is therefore suited to the average user, the users for my web application will be experienced professionals. This therefore means that I can reap the benefits of the \$\_POST method like the fact that all data submitted in a form via the \$\_POST method is fully encrypted, when using HTTPS, and invisible to all users and anyone who might potentially be packet sniffing. Also, there is no limit to the number of variables which may be sent at one time via this method whereas with the \$\_GET method there is a limit. This is useful for my submissions where large amounts of data will be collected.

It is apparent that a hashing algorithm would be very beneficial for my system. Due to a collection of confidential data being used throughout, if a user were to access the application unapproved it could be catastrophic. For this reason, it would be advantageous to use hashing for the passwords for staff accounts. Hashing a password is good because it is quick and easy to store. Instead of storing the user's password as plain text, which is open for anyone to read, it is stored as a hash which is impossible for a human to understand. Password hashing is used to verify the integrity of a password, submitted during my login, against the stored hash in my database.

### 3.7) Algorithm Design and Rationale

#### Hashing Algorithm

Almost all hashing algorithms begin by converting each character into its ASCII or Unicode value. However, this becomes problematic as what is known as a "collision" can occur regularly. This is where two different inputs produce the same hash value. For example, the string "this" and "hits" would have the same ASCII values. This can be overcome by including some notion of each character's location into the algorithm. Therefore, by multiplying each character's value by some constant raised to the power of its position will eliminate these potential collisions.

$$\begin{array}{cccc}
 t & h & i & s \\
 \downarrow & \downarrow & \downarrow & \downarrow \\
 116 & 104 & 105 & 115 \\
 \times & \times & \times & \times \\
 g^0 & g^1 & g^2 & g^3 \\
 \hline
 116 + g(104 + g(105 + g(115 + g)))
 \end{array}$$

Where g is some integer constant; I shall use 31.

My hashing algorithm in simple instruction form:

- 1 Take an input
- 2 Find the sum of each character's ASCII value multiplied by  $(31^{\text{character's position}})$
- 3 Takes each digit of this integer and replaces it with the factorial of itself
- 4 Splits the new integer into groups of 4 digits
- 5 Converts these 4-digit groups into a hexadecimal value
- 6 Compiling these values into a string gives the hashed value

My hashing algorithm in PHP:

```

1 function passwordHash($input){
2     $g = 31;
3     $hashedPassword = 0;
4     for($i=0; $i<strlen($input); $i++){
5         $hashedPassword = $g * ($hashedPassword + mb_ord($input[$i]));

```



```
6     }
7
8     $hashedPassword = array_map('intval', str_split($hashedPassword,1));
9
10    $arrayLocation = 0;
11    foreach($hashedPassword as $arrayNum){
12        for ($x=$arrayNum-1; $x>=1; $x--){
13            $arrayNum = $arrayNum * $x;
14        }
15        $hashedPassword[$arrayLocation] = $arrayNum;
16        $arrayLocation = $arrayLocation + 1;
17    }
18
19    $hashedPassword = implode($hashedPassword);
20    $hashedPassword = array_map('intval', str_split($hashedPassword,4));
21
22    $length = count($hashedPassword);
23    $firsthalf = array_slice($hashedPassword, 0, $length / 2);
24    $secondhalf = array_slice($hashedPassword, $length / 2);
25
26    $hashedPassword = [];
27
28    for($i=0; $i=count($secondhalf); $i++){
29        try{
30            array_push($hashedPassword, array_pop($secondhalf));
31        } catch (Exception $e){
32            echo 'Caught exception: ', $e->getMessage(), "\n";
33        }
34        try{
35            array_push($hashedPassword, array_pop($firsthalf));
36        } catch (Exception $e){
37            echo 'Caught exception: ', $e->getMessage(), "\n";
38        }
39    }
40
41    $hashedPassword = array_filter($hashedPassword);
42
43    $arrayLocation = 0;
44    foreach($hashedPassword as $arrayNum){
45        $hashedPassword[$arrayLocation] = dechex($arrayNum);
46        $arrayLocation = $arrayLocation + 1;
47    }
48
49    $hashedPassword = implode($hashedPassword);
50    return $hashedPassword;
51 }
```

My hashing algorithm comments:

5 #use of RECURSION as explained in the diagram above  
8 #converts integer from before into an array of each digit  
13 #RECURSION to implement factorial to each element of array  
15 #replaces the element with factorial of it  
19 #implodes the array into a string  
20 #splits string into array of 4 digits  
23 #splits into two evenly lengthed lists  
29 #use of EXCEPTION HANDLING  
30 #use of QUEUE implementations and functions  
31 #riffle sort the two "halves" together  
41 #removes any null values caused by list "halves" being different lengths  
45 #replaces each number in the array with its hexadecimal value  
49 #implodes the array into a string

Hash Verify Algorithm

Of course, if I have created a hashing algorithm then a hash verify algorithm must be paired with it for use during authorisation processes.

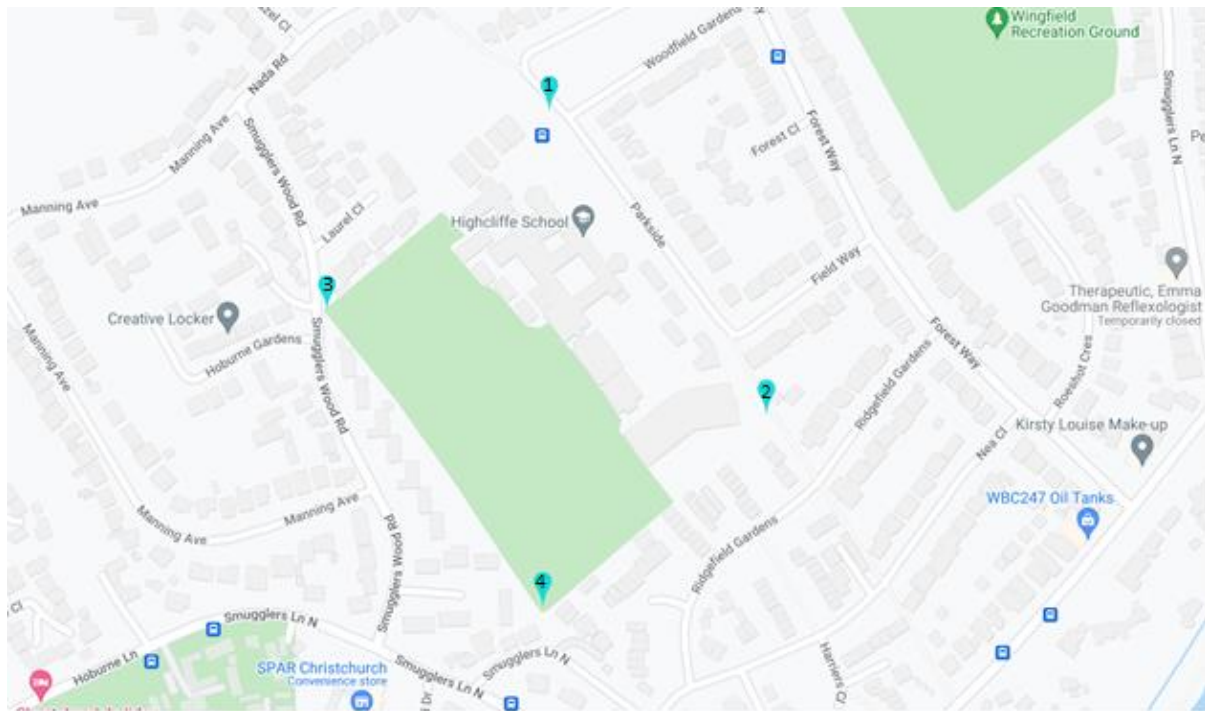
My Hash Verify Algorithm in PHP:

```

1 include("hash.php");
2
3 function verifyHash($input, $hashInput){
4     if(passwordHash($input) == $hashInput){
5         return True;
6     }else{
7         return False;
8     }
9 }
```

Check Location Algorithm

The location of the mobile device needs to be checked to verify if the user is within the grounds of Highcliffe School. This updates a status variable for when the URL is visited by the application. My method of location is longitude/latitude so I will need to find the values at the perimeter of Highcliffe School.

**Location of Points**

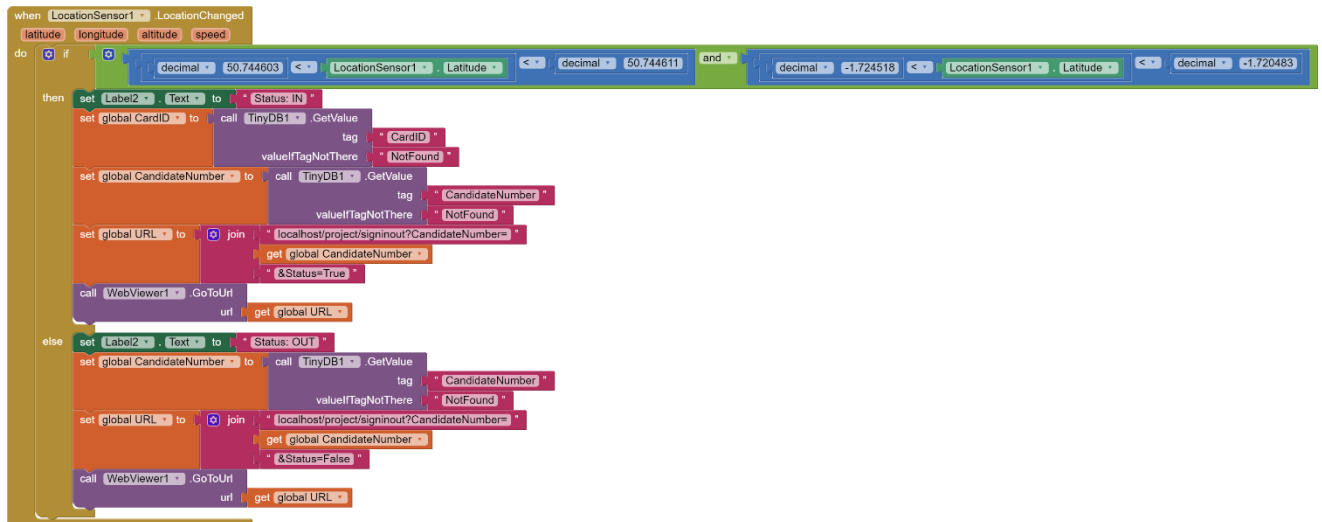
| Point Number | Longitude | Latitude  |
|--------------|-----------|-----------|
| 1            | 50.747491 | -1.722409 |
| 2            | 50.745635 | -1.720306 |
| 3            | 50.746272 | -1.72456  |
| 4            | 50.744462 | -1.722472 |

From this, we can see that the constraints for being classed as onsite would be:

```

1 IF ((50.744462 < get_longitude() < 50.747491) AND (-1.72456 < get_latitude() < -1.720306))
2     THEN status = IN
3 ELSE
4     status = OUT
5 visitURL(status)
```

## My Check Location Algorithm:



TinyDB is a document-oriented database written in pure Python with no external dependencies which is used to store the credential of the student who is currently signed into their application meaning that the application can run in the background and the students will not have to login to their app each time.

Update a Status Algorithm

The main function of my system is to update statuses. Therefore, my algorithm will need to use variables taken in from the URL with the \$\_GET function to be used in the creation of new records in my database.

## My Update a Status Algorithm in PHP:

```

1  $servername = "localhost";
2  $username = "username";
3  $password = "password";
4  $dbname = "database";
5
6  $conn = new mysqli($servername, $username, $password, $dbname);
7  if($conn->connect_error){
8  die("Connection failed: " . $conn->connect_error);
9  }
10
11 $dbCandidateNumber = $_GET["CandidateNumber"];
12 $status = $_GET["Status"];
13
14 if ($status == "True"){
15     $dbStatus = 1;
16 } elseif ($status == "False"){
17     $dbStatus = 0;
18 } else {
19     die("Error: Invalid Status");
20 }
21
22 $sql = "SELECT * FROM `statusids` WHERE CandidateNumber=$dbCandidateNumber;";
23 $result = $conn->query($sql);
24 $row = mysqli_fetch_array($result);
25 if ($row){
26     $dbStatusID = $row['StatusID'];
27
28     $sql = "REPLACE status (StatusID, Status)

```

```

29     VALUES ('$dbStatusID', '$dbStatus');"
30
31     if ($conn->query($sql)==True){
32         echo "New currentstatus record created successfully" . "<br>";
33     } else{
34         echo "Error: " . $sql . "<br>" . $conn->error . "<br>";
35     }
36
37     $sql = "INSERT INTO `signinhistory` (`CandidateNumber`, `Status`, `Type`)
38         VALUES ($dbCandidateNumber, $dbStatus, 'ONLINE');"
39
40     if ($conn->query($sql)==True){
41         echo "New signinhistory record created successfully";
42     } else{
43         echo "Error: " . $sql . "<br>" . $conn->error;
44     }
45 } else{
46     echo "Error: This student account does not exist";
47 }

```

My Update a Status algorithm comments:

```

1-9 #connects to database
11-12 #assigns variables from URL
22 #finds account with candidate number from URL
25 #if account exists
31 #updates current status
37 #adds to signinhistory

```

### Random String Algorithm

Some data fields, like cardID, require a random string.

My Random String Algorithm in PHP:

```

1 function generateRandomString($length) {
2     $characters = '0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ';
3     $charactersLength = strlen($characters);
4     $randomString = '';
5     for ($i = 0; $i < $length; $i++) {
6         $randomString .= $characters[rand(0, $charactersLength - 1)];
7     }
8     return $randomString;
9 }

```

### Datatables Algorithm

I will need to display tables filled from a database frequently throughout my web application. Using datatables will make the implantation very simple and aesthetic.

My Datatables Algorithm in JavaScript:

```

1 <link href="//cdn.datatables.net/1.11.3/css/jquery.dataTables.min.css" rel="stylesheet" />
2 <script src="//cdn.datatables.net/1.11.3/js/jquery.dataTables.min.js"></script>
3
4 $(document).ready(function () {
5     $('#myTable').DataTable({
6         "paging": false,
7         "order": [[4, 'asc'], [5, 'asc'], [1, 'asc']]
8     });
9 });

```

Access Level Check Algorithm

Some pages are not authorised to be viewed by all staff. The user must have an admin account to view the accounts pages therefore I will need an algorithm to redirect the user if unauthorised.

My Access Level Check Algorithm in PHP:

```

1
1 session_start();
2 if (empty($_SESSION['username'])){
3     $username = "Unknown Account";
4     redirect("login.php");
5 } else{
6     $access = $_SESSION['access'];
7     if ($access == "A"){
8         $staffType = "Admin";
9     }elseif($access == "S"){
10        $staffType = "Staff";
11        redirect("401.html");
12    }else{
13        $staffType = "Unknown Staff Type";
14        redirect("401.html");
15    }
16 }
```

My hashing algorithm comments:

2 #if the user has signed in using login.php then \$\_SESSION['username'] will be set to their account username

6 #assigns whether the use is an admin-user or staff-user

11,14 #if the user is not an admin they are redirected to the "Unauthorised" error page

Sign Out All Algorithm

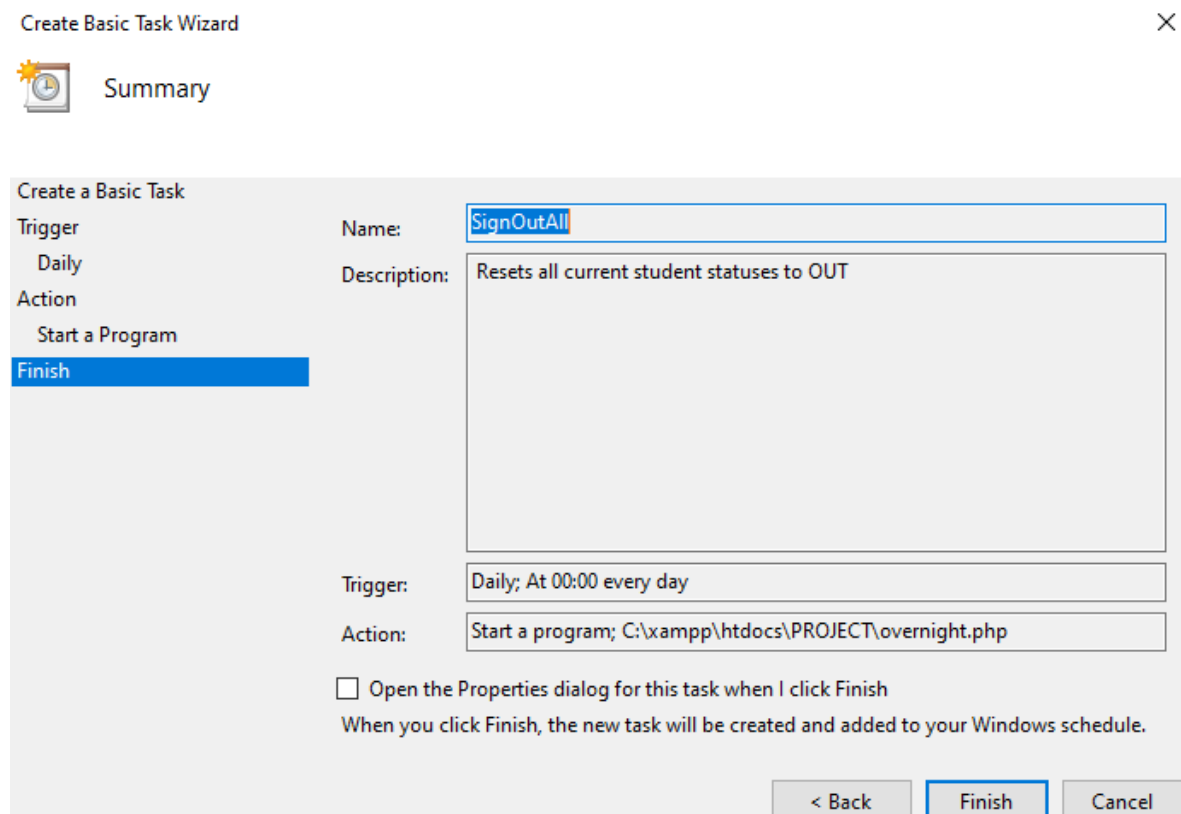
Overnight, all student statuses will need to be reset to OUT for the next coming day.

My Sign Out All Algorithm in PHP:

```

1 $servername ="localhost";
2 $username = "username";
3 $password = "password";
4 $dbname = "database";
5
6 $conn = new mysqli($servername, $username, $password, $dbname);
7 if($conn->connect_error){
8     die("Connection failed: " . $conn->connect_error);
9 }
10 echo "Connected successfully" . "<br>";
11
12 $sql = "UPDATE `status`
13     SET
14     `status` = 0";
15
16 if ($conn->query($sql)==True){
17     echo "New record created successfully";
18 } else{
19     echo "Error: " . $sql . "<br>" . $conn->error;
20 }
```

This would be run by the server every day at midnight using the scheduled task methods. This would be very simple on a windows-based server as shown below.



### Show Password Algorithm

When logging in, the user may require for the password field to be viewable in plain text rather than dots.

My Show Password Algorithm in JavaScript:

```

1 function showPassword() {
2     var x = document.getElementById("inputPassword");
3     if (x.type === "password") {
4         x.type = "text";
5     } else {
6         x.type = "password";
7     }
8 }

```

### Expand Form Algorithm

I will need forms to have a toggle for their visibility. I will implement this using jQuery website animation.

My Expand Form Algorithm in JavaScript:

```

1 function createOpen(){
2     updateClose();
3     deleteClose();
4     $('#newText').hide();
5     $('#createButton').animate({width: '0%'});
6     $('#createStudentForm').slideDown("slow");
7 }
8
9

```

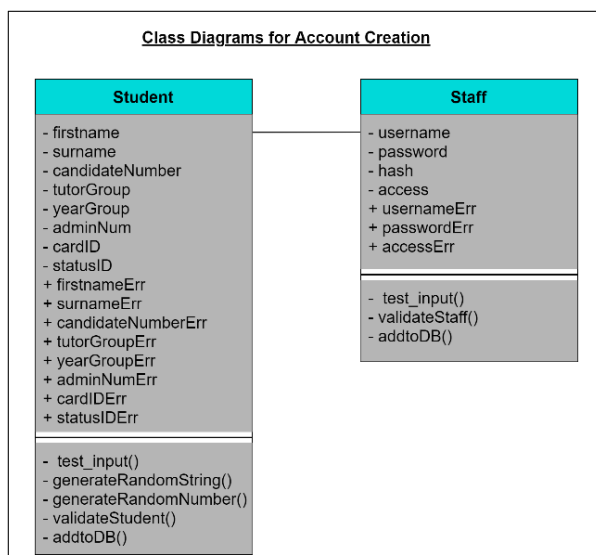
```

10 function createClose(){
11     $("#createStudentForm").slideUp("fast");
12     $("#createButton").animate({width: '100%'});
13     $('#newText').show();
14 };
15
16 function toggleCardID(){
17     $("#inputCardID").slideToggle("fast");
18 };
19
20 function updateOpen(){
21     createClose();
22     deleteClose();
23     $('#updateText').hide();
24     $('#updateButton').animate({width: '0%'});
25     $("#updateStudentForm").slideDown("slow");
26 };
27
28 function updateClose(){
29     $("#updateStudentForm").slideUp("fast");
30     $("#updateButton").animate({width: '100%'});
31     $('#updateText').show();
32 };
33
34 function deleteOpen(){
35     createClose();
36     updateClose();
37     $('#deleteText').hide();
38     $('#deleteButton').animate({width: '0%'});
39     $("#deleteStudentForm").slideDown("slow");
40 };
41
42 function deleteClose(){
43     $("#deleteStudentForm").slideUp("fast");
44     $("#deleteButton").animate({width: '100%'});
45     $('#deleteText').show();
46 };

```

### Create Student Account (Object-Oriented Programming Version) Algorithm

The creation of student accounts can be done very efficiently with OOP, as the student can be modelled as an object with the attributes which are the data stored in my database. The staff account creation is a similar algorithm to this with fewer input fields required. Below is a class diagram for these OOP classes.



## My Create Student Account Algorithm in PHP:

```
1  class student {
2      private $firstname;
3      private $surname;
4      private $candidateNumber;
5      private $tutorGroup;
6      private $yearGroup;
7      private $adminNum;
8      private $cardID;
9      private $statusID;
10     private $firstnameErr;
11     private $surnameErr;
12     private $candidateNumberErr;
13     private $tutorGroupErr;
14     private $yearGroupErr;
15     private $adminNumErr;
16     private $cardIDErr;
17     private $statusIDErr;
18
19     function set_firstname($firstname){
20         $this->firstname = $firstname;
21     }
22
23     function set_surname($surname){
24         $this->surname = $surname;
25     }
26
27     function set_candidateNumber($candidateNumber){
28         $this->candidateNumber = $candidateNumber;
29     }
30
31     function set_tutorgroup($tutorgroup){
32         $this->tutorgroup = $tutorgroup;
33     }
34
35     function set_yeargroup($yeargroup){
36         $this->yeargroup = $yeargroup;
37     }
38
39     function set_adminNum($adminNum){
40         $this->adminNum = $adminNum;
41     }
42
43     function set_cardID($cardID){
44         $this->cardID = $cardID;
45     }
46
47     function get_firstname(){
48         return $this->firstname;
49     }
50
51     function get_surname(){
52         return $this->surname;
53     }
54
55     function get_candidateNumber(){
56         return $this->candidateNumber;
57     }
58
59     function get_tutorgroup(){
60         return $this->tutorgroup;
61     }
62
63     function get_yeargroup(){
64         return $this->yeargroup;
```



```
65     }
66
67     function get_adminNum(){
68         return $this->adminNum;
69     }
70
71     function get_cardID(){
72         return $this->cardID;
73     }
74
75     function get_firstnameErr(){
76         return $this->firstnameErr;
77     }
78
79     function get_surnameErr(){
80         return $this->surnameErr;
81     }
82
83     function get_candidateNumberErr(){
84         return $this->candidateNumberErr;
85     }
86
87     function get_tutorgroupErr(){
88         return $this->tutorGroupErr;
89     }
90
91     function get_yeargroupErr(){
92         return $this->yearGroupErr;
93     }
94
95     function get_adminNumErr(){
96         return $this->adminNumErr;
97     }
98
99     function get_cardIDErr(){
100         return $this->cardIDErr;
101     }
102
103     function generateRandomString($length) {
104         $characters = '0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ';
105         $charactersLength = strlen($characters);
106         $randomString = '';
107         for ($i = 0; $i < $length; $i++) {
108             $randomString .= $characters[rand(0, $charactersLength - 1)];
109         }
110         return $randomString;
111     }
112
113     function generateRandomNumber($length) {
114         $characters = '0123456789';
115         $charactersLength = strlen($characters);
116         $randomNumber = '';
117         for ($i = 0; $i < $length; $i++) {
118             $randomNumber .= $characters[rand(0, $charactersLength - 1)];
119         }
120         return $randomNumber;
121     }
122
123     function validateStudent(){
124         $sendtoDB = True;
125
126         if ($_SERVER["REQUEST_METHOD"] == "POST") {
127
128             if (empty($_POST["fname"])) {
129                 $this->firstnameErr = "First Name Required <br>";
130                 $sendtoDB = False;
```

```
131     } elseif (ctype_alpha($_POST["fname"])) {
132         $this->firstname = test_input($_POST["fname"]);
133     } else {
134         $this->firstnameErr = "Only letters allowed in first name <br>";
135         $sendtoDB = False;
136     }
137
138     if (empty($_POST["sname"])) {
139         $this->surnameErr = "Surname Required <br>";
140         $sendtoDB = False;
141     } elseif (ctype_alpha($_POST["sname"])) {
142         $this->surname = test_input($_POST["sname"]);
143     } else {
144         $this->surnameErr = "Only letters allowed in surname <br>";
145         $sendtoDB = False;
146     }
147
148     if (empty($_POST["canNum"])) {
149         $this->candidateNumberErr = "Candidate Number Required <br>";
150         $sendtoDB = False;
151     } elseif (strlen($_POST["canNum"]) == 4) {
152         $this->candidateNumber = test_input($_POST["canNum"]);
153     } else {
154         $this->candidateNumberErr = "Candidate number must be 4 numbers <br>";
155         $sendtoDB = False;
156     }
157
158     if (empty($_POST["tgroup"])) {
159         $this->tutorGroupErr = "Tutor Group Required <br>";
160         $sendtoDB = False;
161     } else {
162         $this->tutorGroup = test_input($_POST["tgroup"]);
163     }
164
165     if (empty($_POST["ygroup"])) {
166         $this->yearGroupErr = "Year Group Required <br>";
167         $sendtoDB = False;
168     } else {
169         $this->yearGroup = test_input($_POST["ygroup"]);
170     }
171
172     if (empty($_POST["aNum"])) {
173         $this->adminNumErr = "Admin Number Required <br>";
174         $sendtoDB = False;
175     } elseif (strlen($_POST["aNum"]) == 6) {
176         $this->adminNum = test_input($_POST["aNum"]);
177     } else {
178         $this->adminNumErr = "Admin number must be 6 numbers <br>";
179         $sendtoDB = False;
180     }
181
182     if (empty($_POST["cardID"])) {
183         $this->cardID = $this->generateRandomString(11);
184     } elseif (strlen($_POST["cardID"]) == 11) {
185         $this->cardID = $_POST["cardID"];
186     } else {
187         $this->cardIDErr = "CardID must be 11 characters <br>";
188         $sendtoDB = False;
189     }
190 }
191 return $sendtoDB;
192 }
193
194 function addtoDB(){
195     $IPATH = $_SERVER["DOCUMENT_ROOT"]. "/PROJECT/Website/assets/IPATH/";
196     include($IPATH."connectDB.php");
```

```

197
198     $this->statusID = $this->generateRandomNumber(8);
199
200     $conn->autocommit(false);
201
202     $sql = "INSERT INTO `students` (`CandidateNumber`, `misID`, `Surname`, `Firstname`,
203 `TutorGroup`, `CardID`, `YearGroup`)
204         VALUES ('$this->candidateNumber', '$this->adminNum', '$this->surname',
205 '$this->firstname', '$this->tutorGroup', '$this->cardID', '$this->yearGroup')";
206
207     $conn->query($sql);
208
209     $sql2 = "INSERT INTO `statusids` (`CandidateNumber`, `StatusID`)
210         VALUES ('$this->candidateNumber', '$this->statusID')";
211
212     $conn->query($sql2);
213
214     $sql3 = "INSERT INTO `status` (`StatusID`, `Status`)
215         VALUES ('$this->statusID', 0)";
216
217     $conn->query($sql3);
218
219     if ($conn->commit() == TRUE) {
220         echo "New record created successfully";
221         $conn->close();
222         redirect("studentAccounts.php");
223     } else {
224         echo "Error: " . $sql . "<br>" . $conn->error;
225     }
226 }
227 }
228
229 function redirect($url) {
230     ob_start();
231     header('Location: '.$url);
232     ob_end_flush();
233     die();
234 }
235
236 function test_input($data) {
237     $data = trim($data);
238     $data = stripslashes($data);
239     $data = htmlspecialchars($data);
240     return $data;
241 }
242
243 $student = new student();
244 $sendtoDB = $student->validateStudent();
245 if ($sendtoDB == True) {
246     $student->addtoDB();
247 } else {
248     echo "Cannot create new student account due to these errors: <br>";
249     echo $student->get_firstnameErr();
250     echo $student->get_surnameErr();
251     echo $student->get_candidateNumberErr();
252     echo $student->get_tutorGroupErr();
253     echo $student->get_yearGroupErr();
254     echo $student->get_adminNumErr();
255     echo $student->get_cardIDErr();
256 }

```

My Create Student Account comments:

1 # instantiating student class

2-17 #initialising the attributes of the student class

103 #generates a random string used for cardID  
 113 #generates a random number used for statusID  
 123-192 #checks that each input is valid (no special characters in names, correct lengths etc)  
 195-196 #connects to database  
 200 #use of SQL TRANSACTIONS  
 228 #function to redirect to a new URL  
 235 #function to strip data to its raw string value, ensuring no code can be injected  
 244-255 #if all inputs are valid then create new account and if not display why they are invalid inputs

### Logout Algorithm

Users should have the ability to logout of my web application. Since my login uses php sessions, the easiest way to achieve this would be to destroy the session.

My Logout Algorithm in PHP:

```
1 Session_start();
2 Session_destroy();
3 header('Location: ' . "login.php");
```

### Emergency Algorithm

The emergency function should quickly take a record of all the students signed in currently and add them all to a single CSV file which is force downloaded to the client. Use of the SQL ordering systems will help to make the file generated be more effective. Using a 2D array to store each student's credentials to be added to the file will be beneficial.

My Emergency Algorithm in PHP:

```
1 $servername = "localhost";
2 $username = "username";
3 $password = "password";
4 $dbname = "database";
5
6 $conn = new mysqli($servername, $username, $password, $dbname);
7 if($conn->connect_error){
8     die("Connection failed: " . $conn->connect_error);
9 }
10
11 $studentArray = array();
12
13 $sql = "SELECT students.Firstname, students.Surname, students.TutorGroup,
14 students.YearGroup FROM `status`
15     LEFT JOIN `statusids` ON status.StatusID = statusids.StatusID
16     LEFT JOIN `students` ON statusids.CandidateNumber = students.CandidateNumber
17     WHERE status.status = 1
18     ORDER BY students.TutorGroup ASC, students.Surname;";
19
20 $result = $conn->query($sql);
21
22 $j = 0;
23
24 while($row = mysqli_fetch_array($result)) {
25     $studentArray[$j]['firstname'] = $row['Firstname'];
26     $studentArray[$j]['surname'] = $row['Surname'];
27     $studentArray[$j]['tutorgroup'] = $row['TutorGroup'];
28     $studentArray[$j]['yeargroup'] = $row['YearGroup'];
29     $j+=1;
30 }
31
32 $filename = date("D-M-Y H.i") . ".csv";
33 $myfile = fopen($filename, "w") or die("Unable to open file!");
```

```
34
35 $txt = "This is all students with the current status of IN \n\n";
36 fwrite($myfile,$txt);
37
38 for ($i = 0 ; $i < count($studentArray); $i++) {
39     $txt = $studentArray[$i]['tutorgroup'] . ", " . $studentArray[$i]['surname'] . ", " .
40     $studentArray[$i]['firstname'] . ", " . $studentArray[$i]['yeargroup'] . "\n";
41     fwrite($myfile, $txt);
42 }
43
44 fclose($myfile);
45
46 header('Content-Description: File Transfer');
47 header('Content-Disposition: attachment; filename='.basename($filename));
48 header('Expires: 0');
49 header('Cache-Control: must-revalidate');
50 header('Pragma: public');
51 header('Content-Length: ' . filesize($filename));
52 header("Content-Type: text/plain");
53 readfile($filename);
```

### 3.8) Product Testing Strategy

For the testing of my system, I could choose Blackbox testing. This is essentially a software testing method in which the internal structure, design and implementation of the item being tested is not known to the tester. This would require me to source an outside tester who is not aware of the function of my system. I would allow the user full access to my system, but not the rear end code, and see if they raise any issues. These could then be noted and fixed, ready for the same process to occur once more. This could be a useful method as it would be a functional test, meaning that a bug would be most likely to be found in the more commonly used elements of the system. This would have a low time-consumption as an external tester could continue to test as I, the engineer, could simultaneously fix any bugs found.

However, a more plausible testing process would be an iterative Whitebox testing. This is a software testing method in which the internal structure, design and implementation of the item being assessed is known to the tester. This option seems like the more obvious option as it provides a more rigorous test of the system. Although a Whitebox test would be more time-consuming than Blackbox, it would be more beneficial. This is due to the procedural method, by which all errors should be uncovered. These can then be fixed, leaving a fully integral system. This type of testing would be possible as I will know my system inside out and have a high level of programming knowledge. It will also allow me to experiment with erroneous, boundary and typical data which would not be possible if using an external tester. For these reasons, I have chosen to complete a Whitebox test on my system.



## 4) Testing

### 4.1) Test Plan

My test objectives will be split into two groups: the mobile application and the web application.

#### Mobile Application

My mobile app will not be able to be tested as fully working as it is not hosted and neither is the server. Both are locally hosted meaning that I currently have no way for a mobile device to communicate with the server. Therefore, my testing will be theoretically based, ensuring that when the server becomes an online host, the system will operate correctly. The main function of my mobile app is to update the status of a student which is done via a web address visited in the background of the app. Therefore, when testing, I can test that the app selects the correct URL and then test that URL on a pc with the local PHP server running.

#### Web Application

My web application can be tested using a local PHP server. This ensures that the system will operate when hosted online. I am using XAMPP to run an Apache server.

#### Test Objectives

Mobile Application:

- 1) Ensure the mobile authorisation function works correctly
- 2) Ensure the location-based status works correctly
- 3) Ensure the URL visited by the mobile app functions correctly

Web Application:

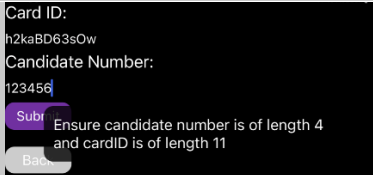
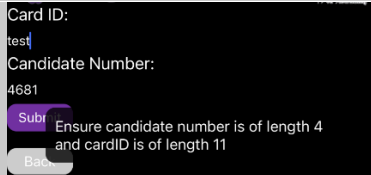
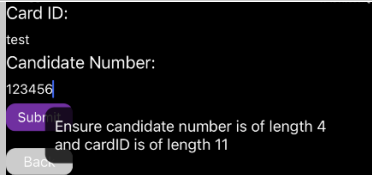
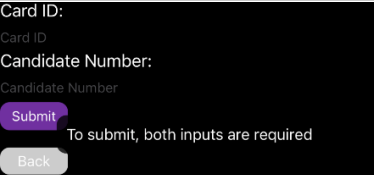
- 4) Ensure the web login function works correctly
- 5) Ensure access levels operate correctly
- 6) Ensure the "Student Statuses" page functions correctly
- 7) Ensure the "Status History" page functions correctly
- 8) Ensure the "Update Student Status" page functions correctly
- 9) Ensure the "Student Accounts" page functions correctly
- 10) Ensure the "Staff Accounts" page functions correctly
- 11) Ensure the logout function operates correctly
- 12) Ensure the "Sign Out All" function operates correctly
- 13) Ensure the "Emergency" function operates correctly

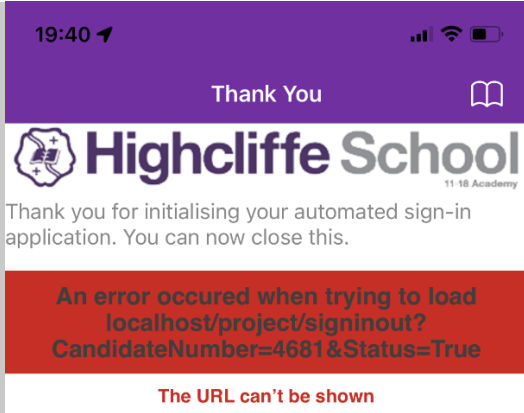
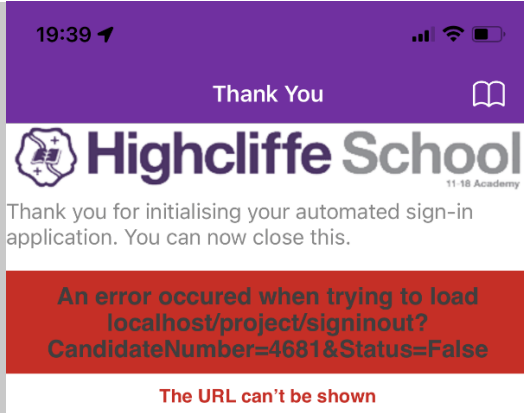
An example of my testing table is seen below. The data in the table is the explanation of each field.

| Test Number         |   | Test Passed? |
|---------------------|---|--------------|
| Objective           | The reference for the objective this test is completing                                 |              |
| Purpose of Test     | The function I am testing   |              |
| Description of Test | What I wish to test within that function  |              |
| Test Data           | The input to my system for this test. For example, erroneous, boundary and typical data |              |
| Expected Result     | What I expect to happen for each input  |              |
| Actual Result       | What actually happens   |              |
| Changes required?   | Whether I need to make any changes  |              |
| Changes Made        | The reference for any change I have made in "4.3) Post Testing Analysis and Fixes"      |              |

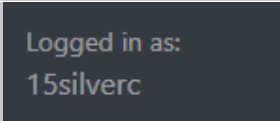
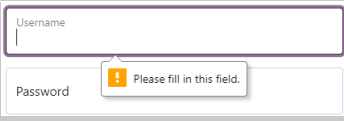
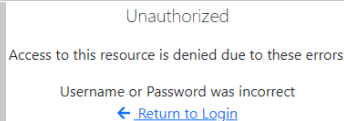
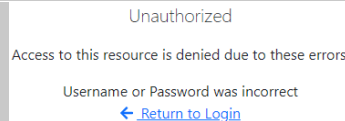
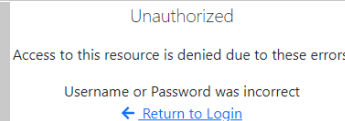


## 4.2) Test Results (linked to test plan)

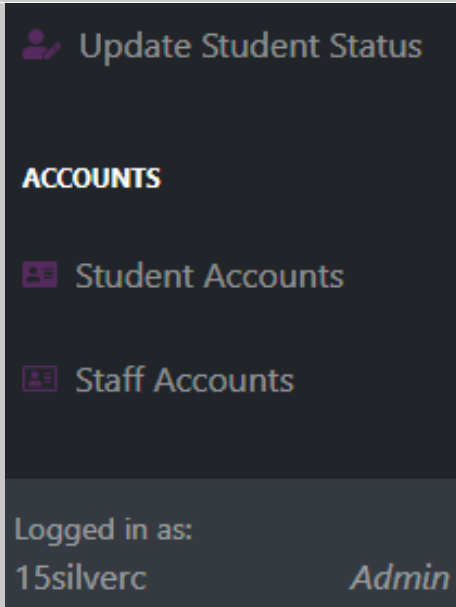
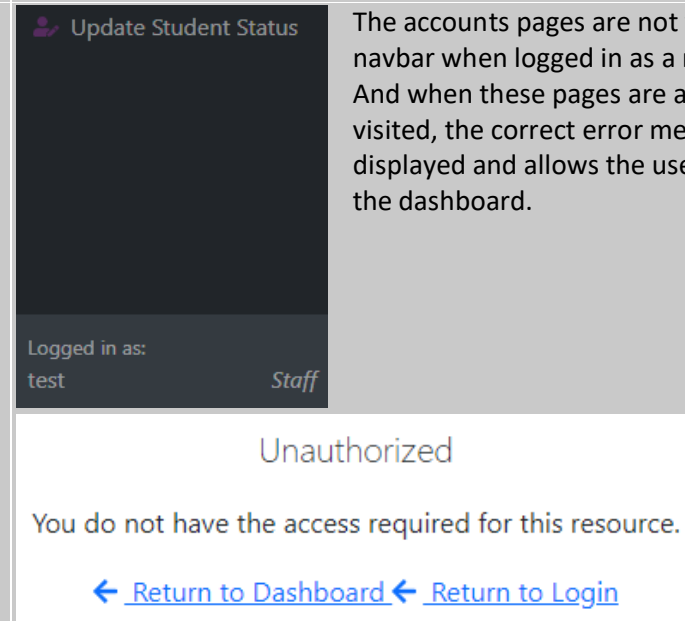
| Test 1              |  |  |   |  | Test Passed?   |
|---------------------|--|--|---|--|--|
| Objective           | 1) Ensure the mobile authorisation function works correctly                  |  |   |  |  |
| Purpose of Test     | Test the function of the mobile application                                  |  |   |  |  |
| Description of Test | Test that the input of cardID and candidate number must be of a valid format |  |   |  |  |
| Test Data           | CardID: h2kaBD63sOw<br>Candidate Number: 4681                                | CardID: h2kaBD63sOw<br>Candidate Number: 123456                                    | CardID: test<br>Candidate Number: 4681  | CardID: test<br>Candidate Number: 123456   | Empty CardID and CandidateNumber fields  |
| Expected Result     | Allows continuation  | Error message  | Error message   | Error message  | Error message  |
| Actual Result       | Redirects to next page   |  |  |  |  |
| Changes required?   | No   |  |   |  |  |
| Changes Made        |  |  |   |  |  |

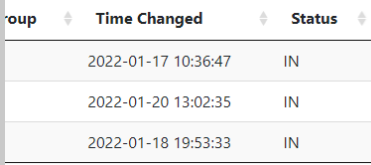
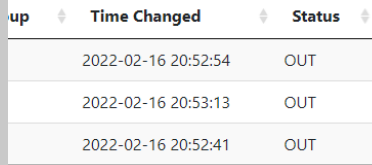
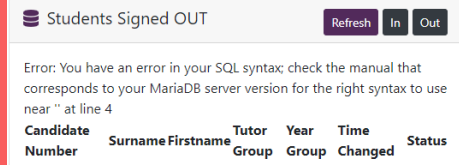
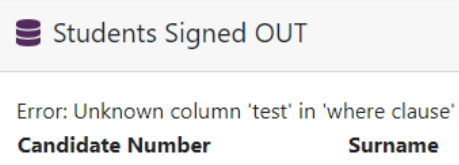
| Test 2a             |  |   | Test Passed? |
|---------------------|--|---|--------------|
| Objective           | 2) Ensure the location-based status works correctly  |   |              |
| Purpose of Test     | Test the function of the mobile application  |   |              |
| Description of Test | Test that when my location is within the constraints of Highcliffe School, the correct web address is visited.<br>The webpage will be unable to be loaded however as currently, in the debugging stage, the system is not hosted online yet. |   |              |
| Test Data           | Candidate Number: 4681<br>Location: Highcliffe School  | Candidate Number: 4681<br>Location: Away from Highcliffe School |              |
| Expected Result     | URL visited:<br>/signinout.php?CandidateNumber=4681&Status=True  | URL visited:<br>signinout.php?CandidateNumber=4681&Status=False |              |
| Actual Result       | <div>   </div>   |   |              |
| Changes required?   | No   |   |              |
| Changes Made        |  |   |              |

| Test 2b             |   |  |   |  | Test Passed? |
|---------------------|---|--|---|--|--------------|
| Objective           | 3) Ensure the URL visited by the mobile app functions correctly   |  |   |  |              |
| Purpose of Test     | Test the function of the signinout.php page   |  |   |  |              |
| Description of Test | Test that the singinout.php page updates status dependant on the candidate number and status in the URL   |  |   |  |              |
| Test Data           | URL:<br>/signinout.php?CandidateNumber=4611&Status=True<br>This Candidate Number exists as a student account.                                     | URL:<br>/signinout.php?CandidateNumber=4611&Status=False<br>This Candidate Number exists as a student account.                                     | URL:<br>/signinout.php?CandidateNumber=1234&Status=False<br>This Candidate Number doesn't exist as a student account. | URL:<br>/signinout.php?CandidateNumber=4611&Status=Test<br>Status variable is invalid. |              |
| Expected Result     | The student with the candidate number 4611 has their status updated to IN   | The student with the candidate number 4611 has their status updated to OUT   | An error message is displayed alerting the user the candidate number is invalid.                                      | An error message is displayed alerting the user the status is invalid.                 |              |
| Actual Result       | /signinout.php?CandidateNumber=4611&Status=True<br>New currentstatus record created successfully<br>New signinhistory record created successfully | /signinout.php?CandidateNumber=4611&Status=False<br>New currentstatus record created successfully<br>New signinhistory record created successfully | /signinout.php?CandidateNumber=1234&Status=False<br>Error: This student account does not exist                        | /signinout.php?CandidateNumber=4611&Status=Test<br>Error: Invalid Status               |              |
| Changes required?   | No  |  |   |  |              |
| Changes Made        |   |  |   |  |              |

| Test 3              |   |  |   |   | Test Passed?  |
|---------------------|---|--|---|---|---|
| Objective           | 4) Ensure the web login function works correctly  |  |   |   |   |
| Purpose of Test     | Test the login function to my web application   |  |   |   |   |
| Description of Test | Test that access is only granted when both the username and password input are correct                      |  |   |   |   |
| Test Data           | Username: 15silverc<br>Password: password!<br>Both the username and password are in the database and match. | Empty username and password.   | Username: 15silverc<br>Password: asdfkh<br>The username is in the database however the password does not match. | Username: asdfkh<br>Password: password!<br>The password is in the database however the username does not match. | Username: asdfkh<br>Password: asdfkh<br>Neither the username nor the password is in the database. |
| Expected Result     | Allows access   | Error message appears  | Does not allow access   | Does not allow access   | Does not allow access   |
| Actual Result       |                            |  |                              |                              |                |
| Changes required?   | No  |  |   |   |   |
| Changes Made        |   |  |   |   |   |

| Test 4              |  | Test Passed?              |
|---------------------|--|---------------------------|
| Objective           | 4) Ensure the web login function works correctly   |                           |
| Purpose of Test     | Test users must be signed in   |                           |
| Description of Test | Test that users are redirected to login page automatically when not signed in; I will be trying to visit dashboard.php |                           |
| Test Data           | Not signed in using login.php  | Signed in using login.php |
| Expected Result     | Redirects to login.php   | Allows access             |
| Actual Result       | Redirects to login.php   | Allows access             |
| Changes required?   | No   |                           |
| Changes Made        |  |                           |

| Test 5              |  |  | Test Passed? |
|---------------------|--|--|--------------|
| Objective           | 5) Ensure access levels operate correctly  |  |              |
| Purpose of Test     | Test the admin/regular staff access levels   |  |              |
| Description of Test | Test that certain webpages are only viewable and accessible to admin staff and not to regular staff  |  |              |
| Test Data           | Signed in as: admin  | Signed in as: staff  |              |
| Expected Result     | Accounts pages are fully viewable.   | Accounts pages do not appear in the navbar and an “Unauthorised” error message when trying to manually visit these pages via URL   |              |
| Actual Result       | <div></div> <p>Both the accounts pages are viewable and accessible when logged in as an admin.</p> | <div></div> <p>The accounts pages are not viewable in the navbar when logged in as a regular staff. And when these pages are attempted to be visited, the correct error message is displayed and allows the user to return to the dashboard.</p> |              |
| Changes required?   | No   |  |              |
| Changes Made        |  |  |              |

| Test 6              |  |                                     |  | Test Passed?                             |
|---------------------|--|-------------------------------------|--|--|
| Objective           | 6) Ensure the "Student Statuses" page functions correctly  |                                     |  |  |
| Purpose of Test     | Test display of Student Statuses dependant on URL  |                                     |  |  |
| Description of Test | Test that the students status table is different based on the show variable in the URL   |                                     |  |  |
| Test Data           | URL:<br>/students.php?show=1   | URL:<br>/students.php?show=0        | URL:<br>/students.php                    | URL:<br>/students.php?show=test          |
| Expected Result     | Table shows all students signed IN   | Table shows all students signed OUT | Displays error message in place of table | Displays error message in place of table |
| Actual Result       |     |                                     |  |  |
| Changes required?   | Yes. Need to improve the error handling of the webpage   |                                     |  |  |
| Changes Made        | Yes  |                                     |  |  |

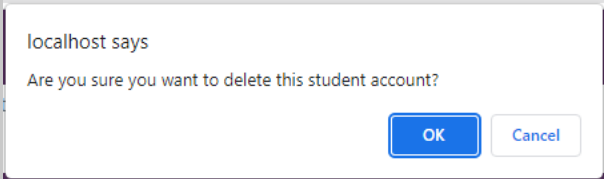
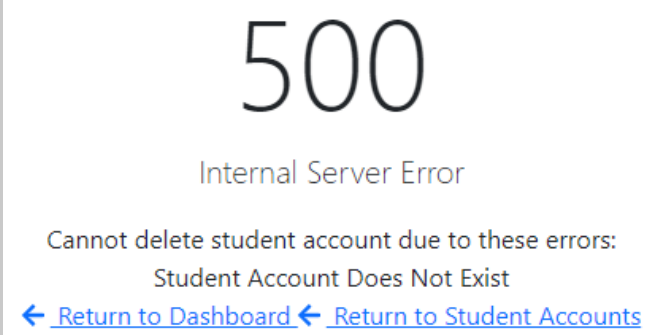
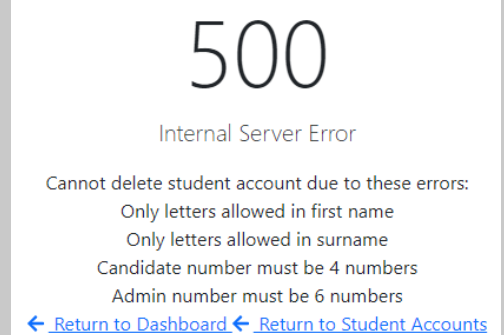
| Test 7              |   | Test Passed? |
|---------------------|---|--------------|
| Objective           | 6) Ensure the “Student Statuses” page functions correctly                           |              |
| Purpose of Test     | Test refresh of table   |              |
| Description of Test | Test that the students status table is updated when the “refresh” button is clicked |              |
| Test Data           | Click the refresh button  |              |
| Expected Result     | Page refreshes table  |              |
| Actual Result       | Page refreshes table  |              |
| Changes required?   | No  |              |
| Changes Made        |   |              |

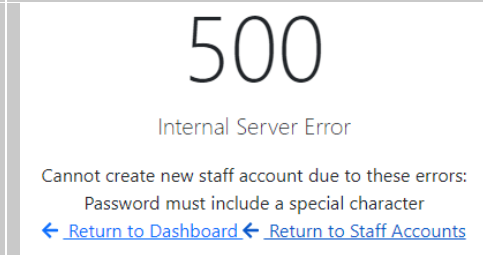
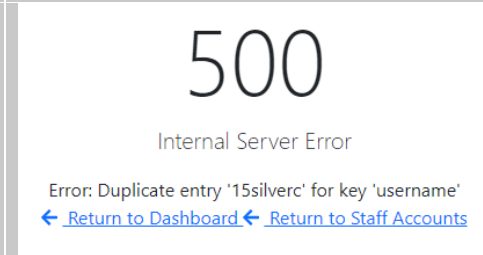


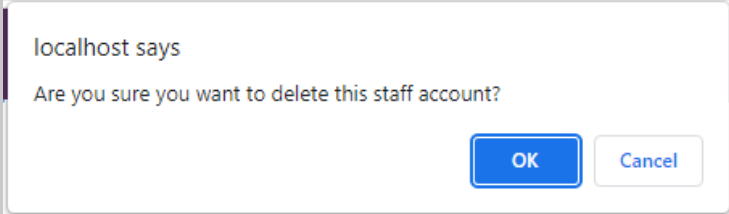
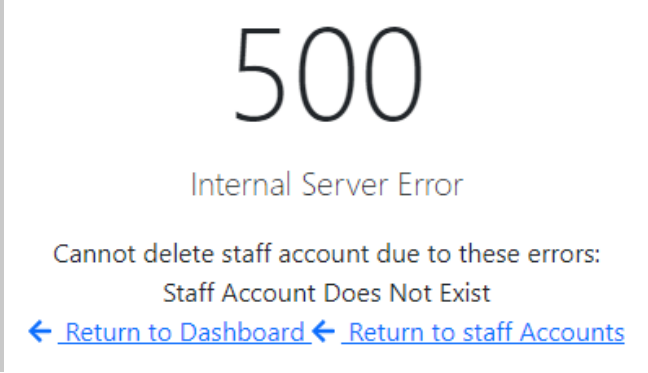
| Test 8              |   |   |                               | Test Passed? |
|---------------------|---|---|-------------------------------|--------------|
| Objective           | 7) Ensure the “Status History” page functions correctly   |   |                               |              |
| Purpose of Test     | Test deletion of student status history   |   |                               |              |
| Description of Test | Testing that the student status history can have the previous month deleted or all deleted  |   |                               |              |
| Test Data           | Pressing the “Delete Previous Month” button   | Pressing the “Delete All History” button                            | Pressing the “Refresh” button |              |
| Expected Result     | “Are you sure?” button appears. On select yes, all data from longer than a month ago is deleted.  | “Are you sure?” button appears. On select yes, all data is deleted. | Page refreshes table          |              |
| Actual Result       | <div><p>localhost says</p><p>Are you sure you want to delete status history from longer than a month ago?</p><div><div>OK</div><div>Cancel</div></div></div> <p>And data from longer than a month ago is deleted.</p> | No “Are you sure?” button appears however the data is deleted.      | Page refreshes table          |              |
| Changes required?   | Yes. Need to add the “Are You Sure?” check to the delete all history button   |   |                               |              |
| Changes Made        | Yes   |   |                               |              |

| Test 9              |   |   | Test Passed? |          |        |            |          |   |                     |
|---------------------|---|---|--------------|----------|--------|------------|----------|---|---------------------|
| Objective           | 8) Ensure the “Update Student Status” page functions correctly  |   |              |          |        |            |          |   |                     |
| Purpose of Test     | Test the Update A Student Status form   |   |              |          |        |            |          |   |                     |
| Description of Test | Test that the form submission only allows valid candidate numbers   |   |              |          |        |            |          |   |                     |
| Test Data           | Candidate Number: 4681<br>Status: IN<br>Candidate Number exists in database   | Candidate Number: 1234<br>Status: IN<br>Candidate Number does not exist in database |              |          |        |            |          |   |                     |
| Expected Result     | The `status` table is updated and a new record created in the `signinhistory` table   | The 500-error page appears alerting the user that the candidate number is not valid |              |          |        |            |          |   |                     |
| Actual Result       | <div><table><thead><tr><th>StatusID</th><th>status</th><th>DateChange</th></tr></thead><tbody><tr><td>23434346</td><td>1</td><td>2022-02-16 21:40:35</td></tr></tbody></table><p>The `status` table is updated however there is no new record created in `signinhistory`.</p></div> <div><h1>500</h1><p>Internal Server Error</p><p>Error: The student account, Candidate Number 1234, does not exist</p><p><a href="#">← Return to Dashboard</a> <a href="#">← Return to Status Update</a></p></div> |   |              | StatusID | status | DateChange | 23434346 | 1 | 2022-02-16 21:40:35 |
| StatusID            | status  | DateChange  |              |          |        |            |          |   |                     |
| 23434346            | 1   | 2022-02-16 21:40:35   |              |          |        |            |          |   |                     |
| Changes required?   | Yes. Need to ensure that the status history table also is updated when the form is submitted  |   |              |          |        |            |          |   |                     |
| Changes Made        | Yes   |   |              |          |        |            |          |   |                     |

| Test 10             |  |   | Test Passed? |
|---------------------|--|---|--------------|
| Objective           | 9) Ensure the “Student Accounts” page functions correctly  |   |              |
| Purpose of Test     | Test the New Student Account form  |   |              |
| Description of Test | Test that the form submission only allows properly formatted inputs  |   |              |
| Test Data           | First Name: Test<br>Surname: Test<br>Candidate Number: 1234<br>Admin Number: 123456<br>CardID: <i>random</i><br>All valid inputs | First Name: !<><br>Surname: !<><br>Candidate Number: 123456<br>Admin Number: 1234<br>CardID: test<br>All invalid inputs   |              |
| Expected Result     | New student account created  | The 500-error page appears alerting the user that each of the fields are invalid  |              |
| Actual Result       | New account created  | <div>500</div> <div>Internal Server Error</div> <div>Cannot create new student account due to these errors:<br/>Only letters allowed in first name<br/>Only letters allowed in surname<br/>Candidate number must be 4 numbers<br/>Admin number must be 6 numbers<br/>CardID must be 11 characters</div> |              |
| Changes required?   | No   |   |              |
| Changes Made        |  |   |              |

| Test 11             |  |   |   | Test Passed? |
|---------------------|--|---|---|--------------|
| Objective           | 9) Ensure the "Student Accounts" page functions correctly  |   |   |              |
| Purpose of Test     | Test the Delete Student Account form   |   |   |              |
| Description of Test | Test that the form submission only allows properly formatted inputs and does not break when an invalid account is input                |   |   |              |
| Test Data           | First Name: Test<br>Surname: Test<br>Candidate Number: 1234<br>Admin Number: 123456<br>A valid student account                         | First Name: Tester<br>Surname: Tester<br>Candidate Number: 4321<br>Admin Number: 654321<br>Valid inputs but not a valid student account | First Name: !<><br>Surname: !<><br>Candidate Number: 123456<br>Admin Number: 1234<br>Valid inputs but not a valid student account |              |
| Expected Result     | Account is deleted from the database, after an "Are you sure?" check, from `students` table along with `status` and `statusID` records | The 500-error page appears alerting the user the account does not exist   | The 500-error page appears alerting the user that each of the fields are invalid  |              |
| Actual Result       |  <p>Account is deleted</p>                           |   |   |              |
| Changes required?   | No   |   |   |              |
| Changes Made        |  |   |   |              |

| Test 12             |   |  |  | Test Passed? |
|---------------------|---|--|--|--------------|
| Objective           | 10) Ensure the "Staff Accounts" page functions correctly            |  |  |              |
| Purpose of Test     | Test the New Staff Account form                                     |  |  |              |
| Description of Test | Test that the form submission only allows properly formatted inputs |  |  |              |
| Test Data           | Username: tester<br>Password: Password\$<br>Access Level: Admin     | Username: tester<br>Password: password (no special character)<br>Access Level: Admin | Username: 15silverc (duplicate)<br>Password: Password\$<br>Access Level: Admin       |              |
| Expected Result     | New staff account created, with a hashed password stored            | The 500-error page appears alerting the user that each of the fields are invalid     | The 500-error page appears alerting the user that each of the fields are invalid     |              |
| Actual Result       | New account created<br>tester 28fb7fb91f72992e2c22606b816a8991890 A |  |  |              |
| Changes required?   | No  |  |  |              |
| Changes Made        |   |  |  |              |

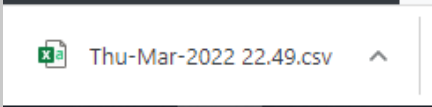

| Test 13             |   |  | Test Passed? |
|---------------------|---|--|--------------|
| Objective           | 10) Ensure the "Staff Accounts" page functions correctly  |  |              |
| Purpose of Test     | Test the Delete Staff Account form  |  |              |
| Description of Test | Test that the form submission only allows properly formatted inputs and does not break when an invalid account is input |  |              |
| Test Data           | Username: tester<br>A valid staff account   | Username: username<br>Not a valid staff account                                      |              |
| Expected Result     | Account is deleted from the database, after an "Are you sure?" check  | The 500-error page appears alerting the user the account does not exist              |              |
| Actual Result       |  <p>Account is deleted</p>            |  |              |
| Changes required?   | No  |  |              |
| Changes Made        |   |  |              |

| Test 14             |  | Test Passed? |
|---------------------|--|--------------|
| Objective           | 11) Ensure the logout function operates correctly  |              |
| Purpose of Test     | Test the logout function   |              |
| Description of Test | Test that the current session is destroyed, and the user must be authorised again                          |              |
| Test Data           | Clicking Logout button<br>Attempting to view dashboard.php without logging back in                         |              |
| Expected Result     | Redirected to login page on button click<br>Redirected to login page when attempting to view dashboard.php |              |
| Actual Result       | Redirected to login page on button click<br>Redirected to login page when attempting to view dashboard.php |              |
| Changes required?   | No   |              |
| Changes Made        |  |              |

| Test 15             |  |                     |  |  |  |  |  |  |  | Test Passed? |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |
|---------------------|--|---------------------|--|--|--|--|--|--|--|--------------|----------|--------|------------|----------|---|---------------------|----------|---|---------------------|----------|---|---------------------|----------|---|---------------------|----------|---|---------------------|----------|---|---------------------|----------|---|---------------------|----------|--------|------------|----------|---|---------------------|----------|---|---------------------|----------|---|---------------------|----------|---|---------------------|----------|---|---------------------|----------|---|---------------------|----------|---|---------------------|
| Objective           | 12) Ensure the “Sign Out All” function operates correctly  |                     |  |  |  |  |  |  |  |              |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |
| Purpose of Test     | Test overnight.php   |                     |  |  |  |  |  |  |  |              |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |
| Description of Test | Test that all current statuses are reset to OUT  |                     |  |  |  |  |  |  |  |              |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |
| Test Data           | Running page   |                     |  |  |  |  |  |  |  |              |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |
| Expected Result     | All student statuses are set to OUT  |                     |  |  |  |  |  |  |  |              |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |
| Actual Result       | <div><div>Before:</div><table><tr><th>StatusID</th><th>status</th><th>DateChange</th></tr><tr><td>23434346</td><td>1</td><td>2022-02-16 21:40:35</td></tr><tr><td>23475684</td><td>1</td><td>2022-02-16 20:52:04</td></tr><tr><td>35402406</td><td>0</td><td>2022-02-16 21:15:15</td></tr><tr><td>39692020</td><td>1</td><td>2022-02-18 11:57:51</td></tr><tr><td>43990890</td><td>0</td><td>2022-02-19 12:45:42</td></tr><tr><td>49374797</td><td>1</td><td>2022-02-16 21:14:47</td></tr><tr><td>88888888</td><td>0</td><td>2022-02-17 12:56:41</td></tr></table></div> <div><div>After:</div><table><tr><th>StatusID</th><th>status</th><th>DateChange</th></tr><tr><td>23434346</td><td>0</td><td>2022-03-17 13:04:59</td></tr><tr><td>23475684</td><td>0</td><td>2022-03-17 13:04:59</td></tr><tr><td>35402406</td><td>0</td><td>2022-02-16 21:15:15</td></tr><tr><td>39692020</td><td>0</td><td>2022-03-17 13:04:59</td></tr><tr><td>43990890</td><td>0</td><td>2022-02-19 12:45:42</td></tr><tr><td>49374797</td><td>0</td><td>2022-03-17 13:04:59</td></tr><tr><td>88888888</td><td>0</td><td>2022-02-17 12:56:41</td></tr></table></div> |                     |  |  |  |  |  |  |  |              | StatusID | status | DateChange | 23434346 | 1 | 2022-02-16 21:40:35 | 23475684 | 1 | 2022-02-16 20:52:04 | 35402406 | 0 | 2022-02-16 21:15:15 | 39692020 | 1 | 2022-02-18 11:57:51 | 43990890 | 0 | 2022-02-19 12:45:42 | 49374797 | 1 | 2022-02-16 21:14:47 | 88888888 | 0 | 2022-02-17 12:56:41 | StatusID | status | DateChange | 23434346 | 0 | 2022-03-17 13:04:59 | 23475684 | 0 | 2022-03-17 13:04:59 | 35402406 | 0 | 2022-02-16 21:15:15 | 39692020 | 0 | 2022-03-17 13:04:59 | 43990890 | 0 | 2022-02-19 12:45:42 | 49374797 | 0 | 2022-03-17 13:04:59 | 88888888 | 0 | 2022-02-17 12:56:41 |
| StatusID            | status   | DateChange          |  |  |  |  |  |  |  |              |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |
| 23434346            | 1  | 2022-02-16 21:40:35 |  |  |  |  |  |  |  |              |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |
| 23475684            | 1  | 2022-02-16 20:52:04 |  |  |  |  |  |  |  |              |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |
| 35402406            | 0  | 2022-02-16 21:15:15 |  |  |  |  |  |  |  |              |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |
| 39692020            | 1  | 2022-02-18 11:57:51 |  |  |  |  |  |  |  |              |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |
| 43990890            | 0  | 2022-02-19 12:45:42 |  |  |  |  |  |  |  |              |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |
| 49374797            | 1  | 2022-02-16 21:14:47 |  |  |  |  |  |  |  |              |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |
| 88888888            | 0  | 2022-02-17 12:56:41 |  |  |  |  |  |  |  |              |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |
| StatusID            | status   | DateChange          |  |  |  |  |  |  |  |              |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |
| 23434346            | 0  | 2022-03-17 13:04:59 |  |  |  |  |  |  |  |              |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |
| 23475684            | 0  | 2022-03-17 13:04:59 |  |  |  |  |  |  |  |              |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |
| 35402406            | 0  | 2022-02-16 21:15:15 |  |  |  |  |  |  |  |              |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |
| 39692020            | 0  | 2022-03-17 13:04:59 |  |  |  |  |  |  |  |              |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |
| 43990890            | 0  | 2022-02-19 12:45:42 |  |  |  |  |  |  |  |              |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |
| 49374797            | 0  | 2022-03-17 13:04:59 |  |  |  |  |  |  |  |              |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |
| 88888888            | 0  | 2022-02-17 12:56:41 |  |  |  |  |  |  |  |              |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |
| Changes required?   | No   |                     |  |  |  |  |  |  |  |              |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |
| Changes Made        |  |                     |  |  |  |  |  |  |  |              |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |        |            |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |          |   |                     |



| Test 16             |   |                   |        |                     |  |        |          |        |                     | Test Passed? |
|---------------------|---|-------------------|--------|---------------------|--|--------|----------|--------|---------------------|--------------|
| Objective           | 12) Ensure the “Sign Out All” function operates correctly |                   |        |                     |  |        |          |        |                     |              |
| Purpose of Test     | Test overnight.php  |                   |        |                     |  |        |          |        |                     |              |
| Description of Test | Test that the scheduled task operates at the correct time |                   |        |                     |  |        |          |        |                     |              |
| Test Data           | Time: 00:00   |                   |        |                     |  |        |          |        |                     |              |
| Expected Result     | Programme runs, statuses set to out                       |                   |        |                     |  |        |          |        |                     |              |
| Actual Result       | Before:   | StatusID          | status | DateChange          |  | After: | StatusID | status | DateChange          |              |
|                     |   | 23434346          | 1      | 2022-02-16 21:40:35 |  |        | 23434346 | 0      | 2022-03-17 13:04:59 |              |
|                     |   | 23475684          | 1      | 2022-02-16 20:52:04 |  |        | 23475684 | 0      | 2022-03-17 13:04:59 |              |
|                     |   | 35402406          | 0      | 2022-02-16 21:15:15 |  |        | 35402406 | 0      | 2022-02-16 21:15:15 |              |
|                     |   | 39692020          | 1      | 2022-02-18 11:57:51 |  |        | 39692020 | 0      | 2022-03-17 13:04:59 |              |
|                     |   | 43990890          | 0      | 2022-02-19 12:45:42 |  |        | 43990890 | 0      | 2022-02-19 12:45:42 |              |
|                     |   | 49374797          | 1      | 2022-02-16 21:14:47 |  |        | 49374797 | 0      | 2022-03-17 13:04:59 |              |
|                     |   | 88888888          | 0      | 2022-02-17 12:56:41 |  |        | 88888888 | 0      | 2022-02-17 12:56:41 |              |
|                     |   | Changes required? | No     |                     |  |        |          |        |                     |              |
| Changes Made        |   |                   |        |                     |  |        |          |        |                     |              |

| Test 17             |   |  |  | Test Passed? |  |  |  |  |  |     |          |       |    |  |     |          |       |    |  |      |         |    |    |  |      |         |      |    |  |     |        |         |    |  |     |       |      |    |  |     |        |       |    |  |     |        |        |    |  |
|---------------------|---|--|--|--------------|--|--|--|--|--|-----|----------|-------|----|--|-----|----------|-------|----|--|------|---------|----|----|--|------|---------|------|----|--|-----|--------|---------|----|--|-----|-------|------|----|--|-----|--------|-------|----|--|-----|--------|--------|----|--|
| Objective           | 13) Ensure the "Emergency" function operates correctly                            |  |  |              |  |  |  |  |  |     |          |       |    |  |     |          |       |    |  |      |         |    |    |  |      |         |      |    |  |     |        |         |    |  |     |       |      |    |  |     |        |       |    |  |     |        |        |    |  |
| Purpose of Test     | Test that the csv file is correct   |  |  |              |  |  |  |  |  |     |          |       |    |  |     |          |       |    |  |      |         |    |    |  |      |         |      |    |  |     |        |         |    |  |     |       |      |    |  |     |        |       |    |  |     |        |        |    |  |
| Description of Test | Test that the csv file is named the current timestamp                             | Test that the csv file is both downloaded and saved to a separate backup folder    | Test that the csv file is properly ordered and correct   |              |  |  |  |  |  |     |          |       |    |  |     |          |       |    |  |      |         |    |    |  |      |         |      |    |  |     |        |         |    |  |     |       |      |    |  |     |        |       |    |  |     |        |        |    |  |
| Test Data           | Pressing the emergency button   | Pressing the emergency button  | Pressing the emergency button  |              |  |  |  |  |  |     |          |       |    |  |     |          |       |    |  |      |         |    |    |  |      |         |      |    |  |     |        |         |    |  |     |       |      |    |  |     |        |       |    |  |     |        |        |    |  |
| Expected Result     | The csv file is named the current timestamp                                       | The csv file is both downloaded and saved to a separate backup folder              | The csv file is properly ordered and correct   |              |  |  |  |  |  |     |          |       |    |  |     |          |       |    |  |      |         |    |    |  |      |         |      |    |  |     |        |         |    |  |     |       |      |    |  |     |        |       |    |  |     |        |        |    |  |
| Actual Result       |  |  | <p>This is all students with the current status of IN</p> <table border="1"> <tbody> <tr><td></td><td></td><td></td><td></td><td></td></tr> <tr><td>6.1</td><td>Tompkins</td><td>James</td><td>13</td><td></td></tr> <tr><td>6.1</td><td>Mackness</td><td>Ellis</td><td>12</td><td></td></tr> <tr><td>6.11</td><td>Shipway</td><td>AJ</td><td>13</td><td></td></tr> <tr><td>6.12</td><td>Bamford</td><td>Alex</td><td>12</td><td></td></tr> <tr><td>6.2</td><td>Silver</td><td>Charlie</td><td>12</td><td></td></tr> <tr><td>6.2</td><td>Smart</td><td>Adam</td><td>13</td><td></td></tr> <tr><td>6.5</td><td>Brophy</td><td>Rowan</td><td>12</td><td></td></tr> <tr><td>6.6</td><td>Davies</td><td>Samuel</td><td>13</td><td></td></tr> </tbody> </table> |              |  |  |  |  |  | 6.1 | Tompkins | James | 13 |  | 6.1 | Mackness | Ellis | 12 |  | 6.11 | Shipway | AJ | 13 |  | 6.12 | Bamford | Alex | 12 |  | 6.2 | Silver | Charlie | 12 |  | 6.2 | Smart | Adam | 13 |  | 6.5 | Brophy | Rowan | 12 |  | 6.6 | Davies | Samuel | 13 |  |
|                     |   |  |  |              |  |  |  |  |  |     |          |       |    |  |     |          |       |    |  |      |         |    |    |  |      |         |      |    |  |     |        |         |    |  |     |       |      |    |  |     |        |       |    |  |     |        |        |    |  |
| 6.1                 | Tompkins  | James  | 13   |              |  |  |  |  |  |     |          |       |    |  |     |          |       |    |  |      |         |    |    |  |      |         |      |    |  |     |        |         |    |  |     |       |      |    |  |     |        |       |    |  |     |        |        |    |  |
| 6.1                 | Mackness  | Ellis  | 12   |              |  |  |  |  |  |     |          |       |    |  |     |          |       |    |  |      |         |    |    |  |      |         |      |    |  |     |        |         |    |  |     |       |      |    |  |     |        |       |    |  |     |        |        |    |  |
| 6.11                | Shipway   | AJ   | 13   |              |  |  |  |  |  |     |          |       |    |  |     |          |       |    |  |      |         |    |    |  |      |         |      |    |  |     |        |         |    |  |     |       |      |    |  |     |        |       |    |  |     |        |        |    |  |
| 6.12                | Bamford   | Alex   | 12   |              |  |  |  |  |  |     |          |       |    |  |     |          |       |    |  |      |         |    |    |  |      |         |      |    |  |     |        |         |    |  |     |       |      |    |  |     |        |       |    |  |     |        |        |    |  |
| 6.2                 | Silver  | Charlie  | 12   |              |  |  |  |  |  |     |          |       |    |  |     |          |       |    |  |      |         |    |    |  |      |         |      |    |  |     |        |         |    |  |     |       |      |    |  |     |        |       |    |  |     |        |        |    |  |
| 6.2                 | Smart   | Adam   | 13   |              |  |  |  |  |  |     |          |       |    |  |     |          |       |    |  |      |         |    |    |  |      |         |      |    |  |     |        |         |    |  |     |       |      |    |  |     |        |       |    |  |     |        |        |    |  |
| 6.5                 | Brophy  | Rowan  | 12   |              |  |  |  |  |  |     |          |       |    |  |     |          |       |    |  |      |         |    |    |  |      |         |      |    |  |     |        |         |    |  |     |       |      |    |  |     |        |       |    |  |     |        |        |    |  |
| 6.6                 | Davies  | Samuel   | 13   |              |  |  |  |  |  |     |          |       |    |  |     |          |       |    |  |      |         |    |    |  |      |         |      |    |  |     |        |         |    |  |     |       |      |    |  |     |        |       |    |  |     |        |        |    |  |
| Changes required?   | No  |  |  |              |  |  |  |  |  |     |          |       |    |  |     |          |       |    |  |      |         |    |    |  |      |         |      |    |  |     |        |         |    |  |     |       |      |    |  |     |        |       |    |  |     |        |        |    |  |
| Changes Made        |   |  |  |              |  |  |  |  |  |     |          |       |    |  |     |          |       |    |  |      |         |    |    |  |      |         |      |    |  |     |        |         |    |  |     |       |      |    |  |     |        |       |    |  |     |        |        |    |  |

## 4.3) Post Testing Analysis and Fixes

### Test 6 Fix

From the testing table, it is clear that there is an error when the variable “show” in the URL for the page is neither a 1 nor a 0. To fix this should be simple; an IF statement will allow me to check whether the \$\_GET variable is valid and if not, then I can display a better error message.

```
if(isset($_GET['show'])){
    $show = $_GET['show'];
    if ($show == 1){
        $status = "IN";
        $show = 1;
    }elseif ($show == 0){
        $status = "OUT";
        $show = 0;
    }else{
        $status = "IN";
        $show = 1;
    }
}
else{
    $status = "IN";
    $show = 1;
}
```

Test 6 fixed.

### Test 8 Fix

From the testing table, it shows that no confirmation check occurs when clicking the “Delete All History” button on the “Status History” page. This is a very easy change, as I can insert a java onclick command for the button which brings up a confirmation dialogue box.

```
<a style="float:right; margin-top: 5px;" class="btn btn-sm btn-dark"
    href="deletehistory.php?all=1" onclick="return confirm('Are you sure you want to
    delete all status history?');">Delete All History</a>
```

Test 8 fixed.

### Test 9 Fix

From test 9, it is apparent that the “Update Student Status” page does not function fully as intended. The record is updated in the `status` table however a new record is not created in `signinhistory`. This turned out to be a very simple fix as the SQL string I passed into the mySQLi Query was \$sql however it should have been \$sql2. A very simple to fix logic error.

Test 9 fixed.



## 5) Appraisal

### 5.1) Objective Fulfilment

| Client-Side Application   |  |  |   |
|---|--|--|---|
| 1   | Can be initialised easily and efficiently  | <i>When setting up the app, it is a simple and easy task. All that is required is the input of two fields and the rest is all done automatically. The fields are saved locally on the device ensuring that the app will not have to be reinitialised for the same student.</i>   | ✓ |
| 2   | Can only be setup for a specific student, students cannot falsely sign in another student  | <i>Once a student is signed in, they require an admin password to reset their app, thus preventing a student from being able to swap accounts easily. However, there is currently no authorisation of the student account on setup: the student could select any candidate number they wish. If an admin were to oversee each setup there would be no issue however this is not practical.</i>   | ~ |
| 3   | Once initialised, has no further user functionality  | <i>All a user sees once the app had been setup is a screen with a thank you message and a button to reset the account with an administrator's help. The app is fully functional behind this screen, with location services running and a website being accessed non-viewably.</i>  | ✓ |
| 4   | Runs a location service with little load to the client device  | <i>The client runs a location check every 15 minutes and then visits a website. With the processing power of current mobile devices, this will be of no stress to the students' mobiles. I would however have liked for the checks to only occur during certain times of the day.</i>  | ✓ |
| 5   | Uses only small amounts of data and uses the fields: <ul style="list-style-type: none"> <li>❖ CandidateNumber</li> <li>❖ Status</li> </ul> | <i>When setting up the app, only the CardID and CandidateNumber fields are required. From there, only the candidate number and status are used when visiting the signinout.php URL.</i>  | ✓ |
| 6   | Uses encryption to protect the fields when signing in and out  | <i>This proved to be very difficult, pretty much impossible when using appinventor. I would have liked to code using other programming languages however this would have doubled my workload, as I would have to create separate apps for iOS and Android rather than just one same application. However, the limitations of using this tool were that some functions were not usable because of their different implementations on different devices.</i> | ✗ |
| 7   | Updates location every 15 minutes and updates status if necessary  | <i>Every 15 minutes, the device checks its location and visits the singinout.php URL. When the device is between the longitude/latitude address of Highcliffe School, the status field is set to "IN" and vice versa.</i>  | ✓ |
| <b>Overall, the client-side application is the part of the project with which I am least happy. With a bit more time, resources and experience I would have aimed to create two applications: one for IOS and one for Android. These applications would be more customised to the individual functions of these different operating systems, meaning that a PHP send/receive can be more tailored. For example, authorising that the student account exists before allowing the user to continue to the Signed_In page.</b> |  |  |   |

| Server-Side Application  |   |   |   |
|--|---|---|---|
| 8  | Maintains statuses of all students                  | <i>The database is secure and reliable. The database has been normalised and the status table keeps a solid, safe record of the statuses of each student.</i>   | ✓ |
| 9  | Resets temporary statuses at the end of each day    | <i>overnight.php works very effectively. The scheduled task runs the program each day at midnight to ensure that the statuses of all students is set to OUT for the next day. The only thing I would potentially change dependant on staff feedback is that the DateChange field is updated during this process meaning that the last status change time is not consistent. However, since no signinhistory record is created, this information can be found elsewhere.</i> | ✓ |
| 10   | Keeps a record of each status change                | <i>The status field can only be changed via sign inout.php or websigninout.php so they will be valid and consistent to whether each student is on or off site.</i>  | ✓ |
| 11   | Deletes old status changes at the end of each month | <i>I chose to add an optional function to delete previous status history, however if the staff decide it can easily become a scheduled task as well. This works really well in the online admin portal, with the option to either delete just the previous month's history or the full history. A confirmation is required too so that the possibility of accidentally making a mistake is lower.</i>   | ✓ |
| <b>Overall, I am very happy with the server side of this project. The database is secure, with a username and password which can be edited in one place for the whole admin portal to be changed, meaning that no direct database access is required. All functionality is provided through the online portal.</b> |   |   |   |

| Admin-Side Application   |   |   |   |
|--|---|---|---|
| 12   | Has a login system  | <i>The login system is robust and does not store passwords in a readable format. The hashing algorithm is not easily reversible. The login system initiates the php session meaning that checks can be made on the login status of the user and the access level.</i>   | ✓ |
| 13   | Utilises Two-Factor Authentication  | <i>This would not be incredibly difficult to implement in the project, however it did not seem necessary as the php login system is secure enough. If the staff requested for it to be used, I would have a php random number generator which sent the code to another device which can be checked on input.</i>  | ✗ |
| 14   | Has the ability to update an individual student status  | <i>The websigninout.php page works very effectively. It is effectively just the signinout.php page with a user interface.</i>   | ✓ |
| 15   | Can display different databases depending on access level   | <i>When signing in, the access level of the user is assigned. The regular staff can check current student statuses and the admin staff can see both student and staff accounts.</i>   | ✓ |
| 16   | Can sort the database by different fields: <ul style="list-style-type: none"> <li>❖ Tutor Group</li> <li>❖ Year Group</li> <li>❖ Surname</li> <li>❖ Status</li> </ul>   | <i>The use of DataTables has been a very successful choice as the tables are easily formattable and have built-in ordering and searching functions. The tables are easily fillable using a mix of php and html.</i>   | ✓ |
| 17   | Has functions to manage user accounts   | <i>When the user is signed in as an administrator, they have access to the accounts pages, including the create and delete accounts functions. These work really effectively, using validity checks and regular expression.</i>   | ✓ |
| 18   | Has an emergency function <ul style="list-style-type: none"> <li>❖ Collates all students who are supposed to be on site</li> <li>❖ Compiles them into a CSV file</li> <li>❖ Downloads the file to the client</li> </ul> | <i>The emergency works very well. It quickly requests for the server to send the full list of students with a status of IN. These are collected into a 2-dimensional array which is in order of tutor group and then alphabetically within those tutor groups. This array is then iterated through, and the details of the students is placed into a Comma-Separated-Value file which is force downloaded to the client with the current datetime as the file name.</i> | ✓ |
| <p><b>I am pleased with the admin-side application as most of the functionality I intended to build into it has been implemented. The portal is very protected by a good quality login system which takes advantage of hashing to improve security even more. The user interface is clean and easy to understand, as well being themed the Highcliffe School purple. The accounts are easy to delete and create and tables are easy to access, order and search through.</b></p> |   |   |   |

## 5.2) User Feedback

### Staff Feedback

All members of staff said that they really liked the user interface design. Especially how closely the design links to some of their pre-existing infrastructures such as their school website. They said the use of the Highcliffe purple was something they valued highly. "Everything is very self-explanatory and easy to find" one teacher commented. The toggleable sidebar was a great success in the eyes of many staff as they felt it both allowed easy navigation of the system while also being able to be

hidden away if they were to be spending a longer period on one page, for example setting up new student accounts. Administrators did however say that if it were possible, they would appreciate a more efficient way to remove or create a whole year group as it could be a lengthy task for one staff member at the start of each new year to have to create a whole year groups student accounts.

The admins also commented that they appreciated the passwords being stored as a hashed value rather than plain text. They said they also appreciated the “portability” of the project, with easily changeable passwords and locations. The use of IPATH so that the database username and password only have to be changed in one file to change for the whole project was appreciated. They also stated that the use of `$_SERVER['DOCUMENT_ROOT']` was incredibly useful since all page locations were transferred automatically.

### Student Feedback

The overwhelming response from the students was that they were all very happy that the new system is automatic. This means that they don't have to consider leaving early to ensure not missing class now as many of them did before. Several students commented on the ease of initiating the mobile application.

Occasionally, students setting up their mobile application did mention that the theme was very appropriate and that they liked that there was both a light and dark theme for the application. Some however did raise concerns about location tracking and the school being able to see where they are at all times. Although, on further explanation that no location data is sent anywhere but actually there is a local check and if within the constraints of the school site, then a single boolean value for status is sent to the server. The students understood this and felt they did actually still have the security they desired.

### 5.3) Proposed Future Revisions and Feedback Analysis

The main change I would propose to make in the future would be to create a more bespoke mobile application for both iOS and Android separately. I feel this would really help to tailor the app and finish it up, rounding it out. The current app, a Javascript programme which runs on both operating systems, works well enough as long as an administrator confirms the setup of each student account. However, I would prefer to have a fully technological system wherein the setup credentials can be verified by the school server and the app cannot continue, unless receiving the confirmation from the server. This would be relatively simple to implement, coding in Swift for iOS, and in Java for Android. I would also add, to the first page, privacy notes to ensure that all students downloading the application understand that their location will be unknown to the school.

A different change I would make would be to the account creation and deletion. The administrators mentioned that it would be a tiresome task each year to have to remove the previous year group's accounts and create the new year group's accounts. For this reason, I would create a function whereby a whole year group can be removed at once, protected by an administrator password or some kind of two factor authentication so that it cannot be mistakenly used. I would also make a new, more efficient, method to create multiple accounts for a whole year group. The school also have their own email system in place currently, so integrating staff accounts in with this system could be very beneficial. This would also mean that a password reset operation would be possible as opposed to a member of staff having to request it from a system administrator.



## 5.4) Conclusion

Overall, I feel that my system is more than capable of fixing the problem that Highcliffe School have currently. While there are still some parts which could be adjusted to work more effectively, the majority of my objectives were completed. The system will function and keep track of whether the sixth form students are on or off site. I feel I have therefore sufficiently completed my project and think that it has been developed enough to go into production and full use.



## 6) Appendix

### 6.1) File Structure

- > Project
  - > Website
    - > assets
      - > hash
        - hash.php
        - hashverify.php
      - > img
        - error-404-monochrome.svg
        - highcliffelogo.png
      - > IPATH
        - connectDB.php
        - footer.php
        - headernav.php
    - > css
      - styles.css
      - switch.css
    - > js
      - scripts.js
  - 401.html
  - 404.html
  - 500.html
  - authenticate.php
  - connect-mysql.php
  - createStaffOOP.php
  - createStudentOOP.php
  - dashboard.php
  - deletehistory.php
  - deleteStaffOOP.php
  - deleteStudentOOP.php
  - emergency.php
  - layout-sidenav-light.html
  - login.php
  - logout.php
  - password.html
  - staffAccounts.php
  - statushistory.php
  - studentAccounts.php
  - students.php
  - updateStatus.php
  - websigninout.php
  - overnight.php
  - signinout.php
  - open screen
  - initialise
  - signed\_in

## 6.2) Code with COMMENTS

### hash.php

```
<?php
```

```
function passwordHash($input){
    $g = 31;
    $hashedPassword = 0;
    for($i=0; $i<strlen($input); $i++){
        $hashedPassword = $g * ($hashedPassword + mb_ord($input[$i]));
        #use of RECURSION as explained in the diagram above
    }

    $hashedPassword = array_map('intval', str_split($hashedPassword,1));
    #converts integer from before into an array of each digit

    $arrayLocation = 0;
    foreach($hashedPassword as $arrayNum){
        for ($x=$arrayNum-1; $x>=1; $x--){
            $arrayNum = $arrayNum * $x;
            #RECURSION to implement factorial to each element of array
        }
        $hashedPassword[$arrayLocation] = $arrayNum;
        #replaces the element with factorial of it
        $arrayLocation = $arrayLocation + 1;
    }

    $hashedPassword = implode($hashedPassword);
    #implodes the array into a string

    $hashedPassword = array_map('intval', str_split($hashedPassword,4));
    #splits string into array of 4 digits

    $length = count($hashedPassword);
    $firsthalf = array_slice($hashedPassword, 0, $length / 2);
    #splits into two evenly lengthed lists
    $secondhalf = array_slice($hashedPassword, $length / 2);

    $hashedPassword = [];

    for($i=0; $i=count($secondhalf); $i++){
        try{ #use of EXCEPTION HANDLING
            array_push($hashedPassword, array_pop($secondhalf));
            #use of QUEUE implementations and functions
        } catch (Exception $e) {
            #riffle sort the two "halves" together
        }
    }
}
```

```
        echo 'Caught exception: ', $e->getMessage(), "\n";
    }
    try{
        array_push($hashedPassword, array_pop($firsthalf));
    } catch (Exception $e) {
        echo 'Caught exception: ', $e->getMessage(), "\n";
    }
}

$hashedPassword = array_filter($hashedPassword);
    #removes any null values caused by list "halves" being different lengths

$arrayLocation = 0;
foreach($hashedPassword as $arrayNum){
    $hashedPassword[$arrayLocation] = dechex($arrayNum);
    #replaces each number in the array with its hexadecimal value
    $arrayLocation = $arrayLocation + 1;
}

$hashedPassword = implode($hashedPassword);
    #implodes the array into a string
return $hashedPassword;
}

?>
```

#### hashverify.php

```
<?php

include("hash.php");

function verifyHash($input, $hashInput){
    if(passwordHash($input) == $hashInput){
        return True;
    }else{
        return False;
    }
}

?>
```

connectDB.php

```

<?php      #this is called from IPATH each time I want to connect to my php server
    $servername = "localhost";
    $username = "username";
    $password = "password";
    $dbname = "database";

    $conn = new mysqli($servername, $username, $password, $dbname); #using SQLi
    if($conn->connect_error){
        die("Connection failed: " . $conn->connect_error);
        #If an error occurs then it is outputted
    }
?>

```

footer.php

```

<!-- This is called from IPATH at the bottom of each webpage as my footer -->

<footer class="py-4 bg-light mt-auto">
    <div class="container-fluid px-4">
        <div class="d-flex align-items-center justify-content-between small">
            <div class="text-muted">Copyright &copy; Your Website 2021</div>
            <div>
                <a href="#">Privacy Policy</a>
                &middot;
                <a href="#">Terms &amp; Conditions</a>
            </div>
        </div>
    </div>
</footer>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js" crossorigin="anonymous"></script>
<script src="js/scripts.js"></script>
<script src="https://cdn.jsdelivr.net/npm/chart.js/2.8.0/Chart.min.js" crossorigin="anonymous"></script>
<script src="assets/demo/chart-area-demo.js"></script>
<script src="assets/demo/chart-bar-demo.js"></script>
<script src="https://cdn.jsdelivr.net/npm/simple-datatables@latest" crossorigin="anonymous"></script>
<script src="js/datatables-simple-demo.js"></script>

```

headernav.php

```

<!-- This is called from IPATH at the top of each webpage as my header -->

<head>
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />

```

```
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no" />
<meta name="description" content="" />
<meta name="author" content="" />
<link href="https://cdn.jsdelivr.net/npm/simple-datatables@latest/dist/style.css" rel="stylesheet" />
<link href="css/styles.css?version=55" rel="stylesheet" />
<script src="https://cdn.jsdelivr.net/npm/font-awesome@5.15.3/js/all.min.js" crossorigin="anonymous"></script>
</head>

<style>
  body{
    overflow-y: auto !important;
  }

  body::-webkit-scrollbar {
    width: 8px;      /* width of the entire scrollbar */
  }

  body::-webkit-scrollbar-track {
    background: none; /* colour of the tracking area */
  }

  body::-webkit-scrollbar-thumb {
    background-color: #343a40; /* colour of the scroll thumb */
    border-radius: 20px; /* roundness of the scroll thumb */
    border: 3px; /* creates padding around scroll thumb */
  }

  ::-webkit-scrollbar {
    width: 8px;      /* width of the entire scrollbar */
  }

  ::-webkit-scrollbar-track {
    background: none; /* colour of the tracking area */
  }

  ::-webkit-scrollbar-thumb {
    background-color: #222a5b; /* colour of the scroll thumb */
    border-radius: 20px; /* roundness of the scroll thumb */
    border: 3px; /* creates padding around scroll thumb */
  }
</style>

<?php
function redirect($url){
  #This function is used to redirect the user throughout the website
  ob_start();
  header('Location: '. $url);
  ob_end_flush();
}
```

```

        die();
    }

    session_start();    #This is an authorisation check
    if (empty($_SESSION['username'])) {
        #If the user has signed in using login.php then $_SESSION['username'] will be set to their username
        $username = "Unknown Account";
        redirect("login.php");
    } else {
        $username = $_SESSION['username'];
        $access = $_SESSION['access'];
        #This assigns whether the current user is an admin-user or staff-user
        if ($access == "A") {
            $staffType = "Admin";
        } elseif ($access == "S") {
            $staffType = "Staff";
        } else {
            $staffType = "Unknown Staff Type";
        }
    }
}

?>

<body class="sb-nav-fixed">
    <nav class="sb-topnav navbar navbar-expand navbar-dark highcliffe">
        <!-- Navbar Brand -->
        <a class="navbar-brand ps-3" href="dashboard.php"></a>
        <!-- Sidebar Toggle -->
        <button class="btn btn-link btn-sm order-1 order-lg-0 me-4 me-lg-0" id="sidebarToggle" href="#"><i class="fas fa-bars"></i></button>
        <!-- Navbar Search -->
        <form class="d-none d-md-inline-block form-inline ms-auto me-0 me-md-3 my-2 my-md-0">
            <div class="input-group">
                <input class="form-control" type="text" placeholder="Search for..." aria-label="Search for..." aria-describedby="btnNavbarSearch" />
                <button class="btn btn-search" style="background: #343A40 !important; color: #ffffff !important;" id="btnNavbarSearch" type="button"><i
class="fas fa-search"></i></button>
            </div>
        </form>
        <!-- Navbar -->
        <ul class="navbar-nav ms-auto ms-md-0 me-3 me-lg-4">
            <li class="nav-item dropdown">
                <a class="nav-link dropdown-toggle" id="navbarDropdown" href="#" role="button" data-bs-toggle="dropdown" aria-expanded="false"><i class="fas
fa-user fa-fw"></i></a>
                <ul class="dropdown-menu dropdown-menu-end" aria-labelledby="navbarDropdown">
                    <li><a class="dropdown-item" href="#">Settings</a></li>
                    <li><a class="dropdown-item" href="#">Activity Log</a></li>
                    <li><hr class="dropdown-divider" /></li>
                    <li><a class="dropdown-item" href="logout.php">Logout</a></li>
                </ul>
            </li>
        </ul>
    </nav>

```



```

        </ul>
      </li>
    </ul>
  </nav>
</body>

<div id="layoutSidenav">
  <div id="layoutSidenav_nav">
    <nav class="sb-sidenav accordion sb-sidenav-dark bg-dark" id="sidenavAccordion">
      <div class="sb-sidenav-menu">
        <div class="nav">

          <div class="sb-sidenav-menu-heading">Core</div>

          <a class="nav-link" href="dashboard.php">
            <div class="sb-nav-link-icon"><i class="fas fa-tachometer-alt"></i></div>
            Dashboard
          </a>

          <a class="nav-link" href="emergency.php">
            <div class="sb-nav-link-icon"><i class="fas fa-first-aid"></i></div>
            Emergency
          </a>

          <div class="sb-sidenav-menu-heading">Statuses</div>

          <a class="nav-link" href="students.php?show=1">
            <div class="sb-nav-link-icon"><i class="fas fa-database"></i></div>
            Student Statuses
          </a>

          <a class="nav-link" href="statushistory.php">
            <div class="sb-nav-link-icon"><i class="fas fa-history"></i></div>
            Status History
          </a>

          <a class="nav-link" href="updateStatus.php">
            <div class="sb-nav-link-icon"><i class="fas fa-user-edit"></i></div>
            Update Student Status
          </a>

          <div class="sb-sidenav-menu-heading" <?php if ($access=="S"){?>style="display:none"<?php } ?>>Accounts</div>
          <!--Only shows these choices if signed in as an administrator-->

          <a class="nav-link" href="studentAccounts.php" <?php if ($access=="S"){?>style="display:none"<?php } ?>>

```

```

        <div class="sb-nav-link-icon"><i class="fas fa-address-card"></i></div>
        Student Accounts
    </a>
    <a class="nav-link" href="staffAccounts.php" <?php if ($access=="S"){?>style="display:none"<?php } ?>>
        <div class="sb-nav-link-icon"><i class="far fa-address-card"></i></div>
        Staff Accounts
    </a>

    <div class="sb-sidenav-menu-heading" <?php if ($access=="S"){?>style="display:none"<?php } ?>>Admin</div>
    <!-- Only shows these choices if signed in as administrator-->

    <a class="nav-link collapsed" href="#" data-bs-toggle="collapse" data-bs-target="#collapseLayouts" aria-expanded="false" aria-
controls="collapseLayouts" <?php if ($access=="S"){?>style="display:none"<?php } ?>>
        <div class="sb-nav-link-icon"><i class="fas fa-columns"></i></div>
        Layouts
        <div class="sb-sidenav-collapse-arrow"><i class="fas fa-angle-down"></i></div>
    </a>

    <div class="collapse" id="collapseLayouts" aria-labelledby="headingOne" data-bs-parent="#sidenavAccordion">
        <nav class="sb-sidenav-menu-nested nav">
            <a class="nav-link" href="">Static Navigation</a>
            <a class="nav-link" href="layout-sidenav-light.html">Light Sidenav</a>
        </nav>
    </div>

    <a class="nav-link collapsed" href="#" data-bs-toggle="collapse" data-bs-target="#collapsePages" aria-expanded="false" aria-
controls="collapsePages" <?php if ($access=="S"){?>style="display:none"<?php } ?>>
        <div class="sb-nav-link-icon"><i class="fas fa-book-open"></i></div>
        Pages
        <div class="sb-sidenav-collapse-arrow"><i class="fas fa-angle-down"></i></div>
    </a>

    <div class="collapse" id="collapsePages" aria-labelledby="headingTwo" data-bs-parent="#sidenavAccordion">
        <nav class="sb-sidenav-menu-nested nav accordion" id="sidenavAccordionPages">
            <a class="nav-link collapsed" href="#" data-bs-toggle="collapse" data-bs-target="#pagesCollapseAuth" aria-expanded="false"
aria-controls="pagesCollapseAuth">
                Authentication
                <div class="sb-sidenav-collapse-arrow"><i class="fas fa-angle-down"></i></div>
            </a>
            <div class="collapse" id="pagesCollapseAuth" aria-labelledby="headingOne" data-bs-parent="#sidenavAccordionPages">
                <nav class="sb-sidenav-menu-nested nav">
                    <a class="nav-link" href="login.php">Login</a>
                    <a class="nav-link" href="createaccount.html">Create Account</a>
                    <a class="nav-link" href="password.html">Forgot Password</a>
                </nav>
            </div>
        </div>

```

```

        <a class="nav-link collapsed" href="#" data-bs-toggle="collapse" data-bs-target="#pagesCollapseError" aria-expanded="false"
aria-controls="pagesCollapseError">
            Error
            <div class="sb-sidenav-collapse-arrow"><i class="fas fa-angle-down"></i></div>
        </a>
        <div class="collapse" id="pagesCollapseError" aria-labelledby="headingOne" data-bs-parent="#sidenavAccordionPages">
            <nav class="sb-sidenav-menu-nested nav">
                <a class="nav-link" href="401.html">401 Page</a>
                <a class="nav-link" href="404.html">404 Page</a>
                <a class="nav-link" href="500.html">500 Page</a>
            </nav>
        </div>
    </nav>
</div>

</div>
</div>

<div class="sb-sidenav-footer">
    <div class="small">Logged in as:</div>
    <?php echo $username; ?> <!-- Displays who the user is logged in as -->
    <i style="float:right"><?php echo $staffType; ?></i> <!-- Displays which level of staff this account is -->
</div>
</nav>
</div>

```

### styles.css

```

@charset "UTF-8";
/*
 * Bootstrap v5.1.3 (https://getbootstrap.com/)
 * Copyright 2011-2021 The Bootstrap Authors
 * Copyright 2011-2021 Twitter, Inc.
 * Licensed under MIT (https://github.com/twbs/bootstrap/blob/main/LICENSE)
 */

:root {
  --bs-highcliffe: #522a5b; /* This is the hex code for the highcliffe purple */
  --bs-blue: #0d6efd;
  --bs-indigo: #6610f2;
  --bs-purple: #6f42c1;
  --bs-pink: #d63384;
  --bs-red: #dc3545;
  --bs-orange: #fd7e14;
  --bs-yellow: #ffc107;
  --bs-green: #198754;
  --bs-teal: #20c997;

```

```

--bs-cyan: #0dcaf0;
--bs-white: #fff;
--bs-gray: #6c757d;
--bs-gray-dark: #343a40;
--bs-gray-100: #f8f9fa;
--bs-gray-200: #e9ecef;
--bs-gray-300: #dee2e6;
--bs-gray-400: #ced4da;
--bs-gray-500: #adb5bd;
--bs-gray-600: #6c757d;
--bs-gray-700: #495057;
--bs-gray-800: #343a40;
--bs-gray-900: #212529;
--bs-primary: #0d6efd;
--bs-secondary: #6c757d;
--bs-success: #198754;
--bs-info: #0dcaf0;
--bs-warning: #ffc107;
--bs-danger: #dc3545;
--bs-light: #f8f9fa;
--bs-dark: #212529;
--bs-primary-rgb: 13, 110, 253;
--bs-secondary-rgb: 108, 117, 125;
--bs-success-rgb: 25, 135, 84;
--bs-info-rgb: 13, 202, 240;
--bs-warning-rgb: 255, 193, 7;
--bs-danger-rgb: 220, 53, 69;
--bs-light-rgb: 248, 249, 250;
--bs-dark-rgb: 33, 37, 41;
--bs-white-rgb: 255, 255, 255;
--bs-black-rgb: 0, 0, 0;
--bs-body-color-rgb: 33, 37, 41;
--bs-body-bg-rgb: 255, 255, 255;
--bs-font-sans-serif: system-ui, -apple-system, "Segoe UI", Roboto, "Helvetica Neue", Arial, "Noto Sans", "Liberation Sans", sans-serif, "Apple Color
Emoji", "Segoe UI Emoji", "Segoe UI Symbol", "Noto Color Emoji";
--bs-font-monospace: SFMono-Regular, Menlo, Monaco, Consolas, "Liberation Mono", "Courier New", monospace;
--bs-gradient: linear-gradient(180deg, rgba(255, 255, 255, 0.15), rgba(255, 255, 255, 0));
--bs-body-font-family: var(--bs-font-sans-serif);
--bs-body-font-size: 1rem;
--bs-body-font-weight: 400;
--bs-body-line-height: 1.5;
--bs-body-color: #212529;
--bs-body-bg: #fff;
}

div#first .btn-search {
  color: #fff !important;
  background-color: #343A40 !important;

```

```
border-color: #343A40 !important;
}

.btn-search:hover{
color: #522a5b !important;
background-color: #343A40 !important;
border-color: #343A40;
}

.highcliffe{
background-color: #522a5b !important;
}

.sidebar-highcliffe{
background-color: #b987c4 !important;
}

.btn-highcliffe{
color: #fff !important;
background-color: #522a5b !important;
border-color: #522a5b !important;
}
```

The code I have included is because I have updated some colours to fit the Highcliffe School theme. There is lots more to this CSS file, too long to include but can be found online (<https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css>)

#### switch.css

```
/* The switch - the box around the slider */
.switch {
  position: relative;
  display: inline-block;
  width: 45px;
  height: 24px;
}

/* Hide default HTML checkbox */
.switch input {
  opacity: 0;
  width: 0;
  height: 0;
}

/* The slider */
```

```
.slider {
  position: absolute;
  cursor: pointer;
  top: 0;
  left: 0;
  right: 0;
  bottom: 0;
  background-color: #ccc;
  -webkit-transition: .3s;
  transition: .3s;
}

.slider:before {
  position: absolute;
  content: "";
  height: 18px;
  width: 18px;
  left: 3px;
  bottom: 3px;
  background-color: white;
  -webkit-transition: .3s;
  transition: .3s;
}

input:checked + .slider {
  background-color: #522a5b;
}

input:focus + .slider {
  box-shadow: 0 0 1px #522a5b;
}

input:checked + .slider:before {
  -webkit-transform: translateX(21px);
  -ms-transform: translateX(21px);
  transform: translateX(21px);
}

/* Rounded sliders */
.slider.round {
  border-radius: 27px;
}

.slider.round:before {
  border-radius: 50%;
}
```

scripts.js

```

window.addEventListener('DOMContentLoaded', event => {

    // Toggle the side navigation
    const sidebarToggle = document.body.querySelector('#sidebarToggle');
    if (sidebarToggle) {
        // Uncomment Below to persist sidebar toggle between refreshes
        // if (localStorage.getItem('sb|sidebar-toggle') === 'true') {
        //     document.body.classList.toggle('sb-sidenav-toggled');
        // }
        sidebarToggle.addEventListener('click', event => {
            event.preventDefault();
            document.body.classList.toggle('sb-sidenav-toggled');
            localStorage.setItem('sb|sidebar-toggle', document.body.classList.contains('sb-sidenav-toggled'));
        });
    }

});

```

401.html

```

<!-- This is the page used when a user attempts to access something unauthorized -->

<!DOCTYPE php>
<php lang="en">
    <head>
        <meta charset="utf-8" />
        <meta http-equiv="X-UA-Compatible" content="IE=edge" />
        <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no" />
        <meta name="description" content="" />
        <meta name="author" content="" />
        <title>401 Error - Admin</title>
        <link href="css/styles.css" rel="stylesheet" />
        <script src="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.3/js/all.min.js" crossorigin="anonymous"></script>
    </head>
    <body>
        <div id="layoutError">
            <div id="layoutError_content">
                <main>
                    <div class="container">
                        <div class="row justify-content-center">
                            <div class="col-lg-6">
                                <div class="text-center mt-4">
                                    <h1 class="display-1">401</h1>
                                    <p class="lead">Unauthorized</p>
                                    <p>You do not have the access required for this resource.</p>
                                </div>
                            </div>
                        </div>
                    </div>
                </main>
            </div>
        </div>
    </body>

```

```

        <a href="dashboard.php">
            <i class="fas fa-arrow-left me-1"></i>
            Return to Dashboard
        </a>
        <a href="login.php">
            <i class="fas fa-arrow-left me-1"></i>
            Return to Login
        </a>
    </div>
</div>
</div>
</div>
</main>
</div>
<div id="layoutError_footer">
    <footer class="py-4 bg-light mt-auto">
        <div class="container-fluid px-4">
            <div class="d-flex align-items-center justify-content-between small">
                <div class="text-muted">Copyright &copy; Your Website 2021</div>
                <div>
                    <a href="#">Privacy Policy</a>
                    &middot;
                    <a href="#">Terms & Conditions</a>
                </div>
            </div>
        </div>
    </div>
</div>
</div>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js" crossorigin="anonymous"></script>
<script src="js/scripts.js"></script>
</body>
</html>

```

#### 404.html

<!-- This is the page used when a user attempts to access something not located on the server -->

```

<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="utf-8" />
        <meta http-equiv="X-UA-Compatible" content="IE=edge" />
        <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no" />
        <meta name="description" content="" />
        <meta name="author" content="" />
        <title>404 Error - SB Admin</title>
    </head>

```



```

<link href="css/styles.css" rel="stylesheet" />
<script src="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.3/js/all.min.js" crossorigin="anonymous"></script>
</head>
<body>
  <div id="layoutError">
    <div id="layoutError_content">
      <main>
        <div class="container">
          <div class="row justify-content-center">
            <div class="col-lg-6">
              <div class="text-center mt-4">
                
                <p class="lead">This requested URL was not found on this server.</p>
                <a href="dashboard.php">
                  <i class="fas fa-arrow-left me-1"></i>
                  Return to Dashboard
                </a>
              </div>
            </div>
          </div>
        </div>
      </main>
    </div>
    <div id="layoutError_footer">
      <footer class="py-4 bg-light mt-auto">
        <div class="container-fluid px-4">
          <div class="d-flex align-items-center justify-content-between small">
            <div class="text-muted">Copyright &copy; Your Website 2021</div>
            <div>
              <a href="#">Privacy Policy</a>
              &middot;
              <a href="#">Terms & Conditions</a>
            </div>
          </div>
        </div>
      </footer>
    </div>
  </div>
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js" crossorigin="anonymous"></script>
  <script src="js/scripts.js"></script>
</body>
</html>

```

### 500.html

```
<!-- This is the page used when there is an error on the server -->
```

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no" />
    <meta name="description" content="" />
    <meta name="author" content="" />
    <title>500 Error - SB Admin</title>
    <link href="css/styles.css" rel="stylesheet" />
    <script src="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.3/js/all.min.js" crossorigin="anonymous"></script>
  </head>
  <body>
    <div id="layoutError">
      <div id="layoutError_content">
        <main>
          <div class="container">
            <div class="row justify-content-center">
              <div class="col-lg-6">
                <div class="text-center mt-4">
                  <h1 class="display-1">500</h1>
                  <p class="lead">Internal Server Error</p>
                  <a href="dashboard.php">
                    <i class="fas fa-arrow-left me-1"></i>
                    Return to Dashboard
                  </a>
                </div>
              </div>
            </div>
          </div>
        </main>
      </div>
      <div id="layoutError_footer">
        <footer class="py-4 bg-light mt-auto">
          <div class="container-fluid px-4">
            <div class="d-flex align-items-center justify-content-between small">
              <div class="text-muted">Copyright &copy; Your Website 2021</div>
              <div>
                <a href="#">Privacy Policy</a>
                &middot;
                <a href="#">Terms & Conditions</a>
              </div>
            </div>
          </div>
        </footer>
      </div>
    </div>
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js" crossorigin="anonymous"></script>

```

```
<script src="js/scripts.js"></script>
</body>
</html>
```

### authenticate.php

```
<!-- This is the page visited when a user submits the login form -->
```

```
<!DOCTYPE html>
<html lang="en">
```

```
<?php
function test_input($data) {    #This ensures that no code can be inserted by the user
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
}
```

```
function redirect($url) {
    ob_start();
    header('Location: '.$url);
    ob_end_flush();
    die();
}
?>
```

```
<!-- This is the 401-error page format -->
```

```
<head>
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no" />
    <meta name="description" content="" />
    <meta name="author" content="" />
    <title>401 Error - Admin</title>
    <link href="css/styles.css" rel="stylesheet" />
    <script src="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.3/js/all.min.js" crossorigin="anonymous"></script>
</head>
<body>
    <div id="layoutError">
        <div id="layoutError_content">
            <main>
                <div class="container">
                    <div class="row justify-content-center">
                        <div class="col-lg-6">
                            <div class="text-center mt-4">
```

```

<h1 class="display-1">401</h1>
<p class="lead">Unauthorized</p>
<p>Access to this resource is denied due to these errors:</p>
<?php
if((isset($_POST['username'])) && !empty($_POST['username']))){
    if((isset($_POST['password'])) && !empty($_POST['password']))){

        $IPATH = $_SERVER["DOCUMENT_ROOT"]. "/PROJECT/Website/assets/IPATH/";
        include($IPATH."connectDB.php");

        $username = test_input($_POST['username']);
        $result = $conn->query("SELECT * FROM `admins` WHERE username='$username'");
        $password = test_input($_POST['password']);
        $hash = password_hash($password, PASSWORD_DEFAULT);
        #Using a hashing algorithm provides extra security so that no actual passwords are being sent or stored on the database

        if (!$result){
            printf("Error: %s\n", mysqli_error($conn));
            exit();
        }else{
            $IPATH = $_SERVER["DOCUMENT_ROOT"]. "/PROJECT/Website/assets/hash/";
            include($IPATH."hashverify.php");

            while($row = mysqli_fetch_array($result)){
                if(($row['username'] == $username) && (verifyHash($password, $row['hashedpassword']))){
                    #If the input password when hashed matches the hashed password assigned to the input username then allow entry to the website
                    session_start();
                    $_SESSION['username'] = $username;
                    $_SESSION['access'] = $row['access'];
                    redirect("dashboard.php");
                }
            }
        }
    }else{
        echo "No password";
    }
}else{
    echo "No username";
}
echo "Username or Password was incorrect";
?>

<br>
<a href="login.php">
    <i class="fas fa-arrow-left me-1"></i>

```

```

        Return to Login
    </a>
</div>
</div>
</div>
</div>
</main>
</div>
<div id="layoutError_footer">
    <footer class="py-4 bg-light mt-auto">
        <div class="container-fluid px-4">
            <div class="d-flex align-items-center justify-content-between small">
                <div class="text-muted">Copyright &copy; Your Website 2021</div>
                <div>
                    <a href="#">Privacy Policy</a>
                    &middot;
                    <a href="#">Terms & Conditions</a>
                </div>
            </div>
        </div>
    </footer>
</div>
</div>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js" crossorigin="anonymous"></script>
<script src="js/scripts.js"></script>
</body>
</html>

```

### connect-mysql.php

```

<?php
    $servername = "localhost";
    $username = "username";
    $password = "password";
    $dbname = "database";

    $conn = new mysqli($servername, $username, $password, $dbname);
    if($conn->connect_error){
        die("Connection failed: " . $conn->connect_error);
    }
?>

```

### createStaffOOP.php

```

<!-- This is the page visited when the create staff form has been submitted -->

```

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no" />
    <meta name="description" content="" />
    <meta name="author" content="" />
    <title>500 Error - Admin</title>
    <link href="css/styles.css" rel="stylesheet" />
    <script src="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.3/js/all.min.js" crossorigin="anonymous"></script>
  </head>

  <body>
    <div id="layoutError">
      <div id="layoutError_content">
        <main>
          <div class="container">
            <div class="row justify-content-center">
              <div class="col-lg-6">
                <div class="text-center mt-4">
                  <h1 class="display-1">500</h1>
                  <p class="lead">Internal Server Error</p>
                  <?php
                    $IPATH = $_SERVER["DOCUMENT_ROOT"]. "/PROJECT/Website/assets/hash/";

                    include($IPATH."hash.php");

                    class staff {      #OBJECT ORIENTED PROGRAMMING in PHP
                      private $username;
                      private $password;
                      private $hash;
                      private $access;
                      public $usernameErr;
                      public $passwordErr;
                      public $accessErr;

                      function set_username($username){
                        $this->username = $username;
                      }

                      function set_password($password){
                        $this->password = $password;
                      }

                      function set_access($access){
                        $this->access = $access;
                      }

```

```

function get_username(){
    return $this->username;
}

function get_password(){
    return $this->password;
}

function get_hash(){
    return $this->hash;
}

function get_access(){
    return $this->access;
}

function get_usernameErr(){
    return $this->usernameErr;
}

function get_passwordErr(){
    return $this->passwordErr;
}

function get_accessErr(){
    return $this->accessErr;
}

function test_input($data) {
    #Ensures no code can be submitted to the form
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
}

function validateStaff(){
    $sendtoDB = True;

    #These following if statements check that the format of the inputs are correct
    if (empty($_POST["username"])) {
        $this->usernameErr = "Username Required <br>";
        $sendtoDB = False;
    }elseif (preg_match('/[\'^$%&*(){}@#~?><>,|=_+~]/', $_POST["username"])){
        $this->usernameErr = "No special characters allowed in username <br>";
        $sendtoDB = False;
    }else {

```

#REGULAR EXPRESSION

```

        $this->username = $this->test_input($_POST["username"]);
    }

    if (empty($_POST["password"])) {
        $this->passwordErr = "Password Required <br>";
        $sendtoDB = False;
    } elseif (!preg_match('/[\'^$%&*(){}@#~?><,>|_|=+-~]/', $_POST["password"])){ #REGULAR EXPRESSION
        $this->passwordErr = "Password must include a special character <br>";
        $sendtoDB = False;
    } elseif (strlen($_POST["password"]) >= 8){
        $this->password = $this->test_input($_POST["password"]);
        $this->hash = passwordHash($this->password, PASSWORD_DEFAULT);
        #Ensures the passwords are stored as a hash making them more secure
        #USE OF MY OWN HASHING ALGORITHM
        //$this->hash = password_hash($this->password, PASSWORD_DEFAULT); alternative
    } else {
        $this->passwordErr = "Password must be 8 characters or longer <br>";
        $sendtoDB = False;
    }

    if (empty($_POST["access"])) {
        $this->accessErr = "Access Level Required <br>";
        $sendtoDB = False;
    } else{
        $this->access = $_POST["access"];
    }

    return $sendtoDB;
}

function addtoDB(){
    $IPATH = $_SERVER["DOCUMENT_ROOT"]. "/PROJECT/Website/assets/IPATH/";
    include($IPATH."connectDB.php");

    $sql = "INSERT INTO `admins` (`username`, `hashedpassword`, `access`)
        VALUES ('$this->username', '$this->hash', '$this->access')";

    if ($conn->query($sql) == TRUE) {
        echo "New account created successfully";
        $conn->close();
        redirect("staffAccounts.php");
    } else {
        echo "Error: " . $conn->error . "<br>";
    }
}
}
}

```



```

function redirect($url) {
    ob_start();
    header('Location: '.$url);
    ob_end_flush();
    die();
}

$staff = new staff();
$sendtoDB = $staff->validatestaff();
if ($sendtoDB == True) {
    $staff->addtoDB();
} else {
    echo "Cannot create new staff account due to these errors: <br>";
    echo $staff->get_usernameErr();
    echo $staff->get_passwordErr();
    echo $staff->get_accessErr();
}

?>
<a href="dashboard.php">
    <i style="color:#52a5b;" class="fas fa-arrow-left me-1"></i>
    Return to Dashboard
</a>
<a href="staffAccounts.php">
    <i class="fas fa-arrow-left me-1"></i>
    Return to Staff Accounts
</a>
</div>
</div>
</div>
</div>
</main>
</div>
<div id="layoutError_footer">
    <footer class="py-4 bg-light mt-auto">
        <div class="container-fluid px-4">
            <div class="d-flex align-items-center justify-content-between small">
                <div class="text-muted">Copyright &copy; Your Website 2021</div>
                <div>
                    <a href="#">Privacy Policy</a>
                    &middot;
                    <a href="#">Terms &amp; Conditions</a>
                </div>
            </div>
        </div>
    </footer>
</div>

```

```

    </div>
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js" crossorigin="anonymous"></script>
    <script src="js/scripts.js"></script>
  </body>
</html>

```

### createStudentOOP.php

<!-- This is the page visited when the createstudent form is submitted -->

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no" />
    <meta name="description" content="" />
    <meta name="author" content="" />
    <title>500 Error - Admin</title>
    <link href="css/styles.css" rel="stylesheet" />
    <script src="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.3/js/all.min.js" crossorigin="anonymous"></script>
  </head>
  <body>
    <div id="layoutError">
      <div id="layoutError_content">
        <main>
          <div class="container">
            <div class="row justify-content-center">
              <div class="col-lg-6">
                <div class="text-center mt-4">
                  <h1 class="display-1">500</h1>
                  <p class="lead">Internal Server Error</p>
                  <?php
                    class student {      #OBJECT ORIENTED PROGRAMMING in PHP
                      private $firstname;
                      private $surname;
                      private $candidateNumber;
                      private $tutorGroup;
                      private $yearGroup;
                      private $adminNum;
                      private $cardID;
                      private $statusID;
                      public $firstnameErr;
                      public $surnameErr;
                      public $candidateNumberErr;
                      public $tutorGroupErr;
                      public $yearGroupErr;

```

```
public $adminNumErr;  
public $cardIDErr;  
public $statusIDErr;  
  
function set_firstname($firstname){  
    $this->firstname = $firstname;  
}  
  
function set_surname($surname){  
    $this->surname = $surname;  
}  
  
function set_candidateNumber($candidateNumber){  
    $this->candidateNumber = $candidateNumber;  
}  
  
function set_tutorgroup($tutorgroup){  
    $this->tutorgroup = $tutorgroup;  
}  
  
function set_yeargroup($yeargroup){  
    $this->yeargroup = $yeargroup;  
}  
  
function set_adminNum($adminNum){  
    $this->adminNum = $adminNum;  
}  
  
function set_cardID($cardID){  
    $this->cardID = $cardID;  
}  
  
function get_firstname(){  
    return $this->firstname;  
}  
  
function get_surname(){  
    return $this->surname;  
}  
  
function get_candidateNumber(){  
    return $this->candidateNumber;  
}  
  
function get_tutorgroup(){  
    return $this->tutorgroup;  
}
```

```
}

function get_yeargroup(){
    return $this->yeargroup;
}

function get_adminNum(){
    return $this->adminNum;
}

function get_cardID(){
    return $this->cardID;
}

function get_firstnameErr(){
    return $this->firstnameErr;
}

function get_surnameErr(){
    return $this->surnameErr;
}

function get_candidateNumberErr(){
    return $this->candidateNumberErr;
}

function get_tutorgroupErr(){
    return $this->tutorGroupErr;
}

function get_yeargroupErr(){
    return $this->yearGroupErr;
}

function get_adminNumErr(){
    return $this->adminNumErr;
}

function get_cardIDErr(){
    return $this->cardIDErr;
}

function test_input($data) { #Ensures no code can be submitted
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
```

```
    return $data;
}

function generateRandomString($length) {
    #This generates a random string used for cardID
    $characters = '0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ';
    $charactersLength = strlen($characters);
    $randomString = '';
    for ($i = 0; $i < $length; $i++) {
        $randomString .= $characters[rand(0, $charactersLength - 1)];
    }
    return $randomString;
}

function generateRandomNumber($length) {
    #This generates a random number used for statusID
    $characters = '0123456789';
    $charactersLength = strlen($characters);
    $randomNumber = '';
    for ($i = 0; $i < $length; $i++) {
        $randomNumber .= $characters[rand(0, $charactersLength - 1)];
    }
    return $randomNumber;
}

function validateStudent(){
    $sendtoDB = True;

    if ($_SERVER["REQUEST_METHOD"] == "POST") {

        if (empty($_POST["fname"])) {
            $this->firstnameErr = "First Name Required <br>";
            $sendtoDB = False;
        } elseif (ctype_alpha($_POST["fname"])) {
            $this->firstname = test_input($_POST["fname"]);
        } else {
            $this->firstnameErr = "Only letters allowed in first name <br>";
            $sendtoDB = False;
        }

        if (empty($_POST["sname"])) {
            $this->surnameErr = "Surname Required <br>";
            $sendtoDB = False;
        } elseif (ctype_alpha($_POST["sname"])) {
            $this->surname = test_input($_POST["sname"]);
        } else {
            $this->surnameErr = "Only letters allowed in surname <br>";
            $sendtoDB = False;
        }
    }
}
```

```
}

if (empty($_POST["canNum"])) {
    $this->candidateNumberErr = "Candidate Number Required <br>";
    $sendtoDB = False;
} elseif (strlen($_POST["canNum"]) == 4) {
    $this->candidateNumber = test_input($_POST["canNum"]);
} else {
    $this->candidateNumberErr = "Candidate number must be 4 numbers <br>";
    $sendtoDB = False;
}

if (empty($_POST["tgroup"])) {
    $this->tutorGroupErr = "Tutor Group Required <br>";
    $sendtoDB = False;
} else {
    $this->tutorGroup = test_input($_POST["tgroup"]);
}

if (empty($_POST["ygroup"])) {
    $this->yearGroupErr = "Year Group Required <br>";
    $sendtoDB = False;
} else {
    $this->yearGroup = test_input($_POST["ygroup"]);
}

if (empty($_POST["aNum"])) {
    $this->adminNumErr = "Admin Number Required <br>";
    $sendtoDB = False;
} elseif (strlen($_POST["aNum"]) == 6) {
    $this->adminNum = test_input($_POST["aNum"]);
} else {
    $this->adminNumErr = "Admin number must be 6 numbers <br>";
    $sendtoDB = False;
}

if (empty($_POST["cardID"])) {
    $this->cardID = $this->generateRandomString(11);
} elseif (strlen($_POST["cardID"]) == 11) {
    $this->cardID = $_POST["cardID"];
} else {
    $this->cardIDErr = "CardID must be 11 characters <br>";
    $sendtoDB = False;
}
}
return $sendtoDB;
}
```

```
function addtoDB(){
    $IPATH = $_SERVER["DOCUMENT_ROOT"]. "/PROJECT/Website/assets/IPATH/";
    include($IPATH."connectDB.php");

    $this->statusID = $this->generateRandomNumber(8);

    $conn->autocommit(false);    #USE OF SQL TRANSACTIONS

    $sql = "INSERT INTO `students` (`CandidateNumber`, `misID`, `Surname`, `Firstname`, `TutorGroup`, `CardID`,
`YearGroup`)
        VALUES ('$this->candidateNumber', '$this->adminNum', '$this->surname', '$this->firstname', '$this-
>tutorGroup', '$this->cardID', '$this->yearGroup')";

    $conn->query($sql);

    $sql2 = "INSERT INTO `statusids` (`CandidateNumber`, `StatusID`)
        VALUES ('$this->candidateNumber', '$this->statusID')";

    $conn->query($sql2);

    $sql3 = "INSERT INTO `status` (`StatusID`, `Status`)
        VALUES ('$this->statusID', 0)";

    $conn->query($sql3);

    if ($conn->commit() == TRUE) {
        echo "New record created successfully";
        $conn->close();
        redirect("studentAccounts.php");
    } else {
        echo "Error: " . $sql . "<br>" . $conn->error;
    }
}

function redirect($url) {
    ob_start();
    header('Location: '.$url);
    ob_end_flush();
    die();
}

function test_input($data) {
    $data = trim($data);
    $data = stripslashes($data);
}
```

```

        $data = htmlspecialchars($data);
        return $data;
    }

    $student = new student();
    $sendtoDB = $student->validateStudent();
    if ($sendtoDB == True) {
        $student->addtoDB();
    } else {
        echo "Cannot create new student account due to these errors: <br>";
        echo $student->get_firstnameErr();
        echo $student->get_surnameErr();
        echo $student->get_candidateNumberErr();
        echo $student->get_tutorGroupErr();
        echo $student->get_yearGroupErr();
        echo $student->get_adminNumErr();
        echo $student->get_cardIDErr();
    } ?>
    <a href="dashboard.php">
        <i style="color:#52a5b;" class="fas fa-arrow-left me-1"></i>
        Return to Dashboard
    </a>
    <a href="studentAccounts.php">
        <i class="fas fa-arrow-left me-1"></i>
        Return to Student Accounts
    </a>
</div>
</div>
</div>
</div>
</main>
</div>
<div id="layoutError_footer">
    <footer class="py-4 bg-light mt-auto">
        <div class="container-fluid px-4">
            <div class="d-flex align-items-center justify-content-between small">
                <div class="text-muted">Copyright &copy; Your Website 2021</div>
                <div>
                    <a href="#">Privacy Policy</a>
                    &middot;
                    <a href="#">Terms & Conditions</a>
                </div>
            </div>
        </div>
    </footer>
</div>
</div>

```



```

    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js" crossorigin="anonymous"></script>
    <script src="js/scripts.js"></script>
  </body>
</html>

```

### dashboard.php

```

<!-- This is the homepage -->

<!DOCTYPE html>
<html lang="en">
  <?php $IPATH = $_SERVER["DOCUMENT_ROOT"]. "/PROJECT/Website/assets/IPATH/"; include($IPATH."headernav.php"); ?>
  <title>Dashboard - <?php echo $username; ?></title>
  <!-- $username is set in the IPATH header when the user signs in -->

  <body>
    <div id="layoutSidenav_content">
      <main>
        <div class="container-fluid px-4">
          <h1 class="mt-4">Dashboard</h1>
          <ol class="breadcrumb mb-4">
            <li class="breadcrumb-item active">The Highcliffe Sixth Form status dashboard</li>
          </ol>
          <div class="row">
            <div class="col-xl-3 col-md-6">
              <div class="card highcliffe text-white mb-4">
                <div class="card-body">Student Statuses</div>
                <div class="card-footer d-flex align-items-center justify-content-between">
                  <a class="small text-white stretched-link" href="students.php?show=1">Visit this page</a>
                  <div class="small text-white"><i class="fas fa-angle-right"></i></div>
                </div>
              </div>
            </div>
            <div class="col-xl-3 col-md-6">
              <div class="card highcliffe text-white mb-4">
                <div class="card-body">Status History</div>
                <div class="card-footer d-flex align-items-center justify-content-between">
                  <a class="small text-white stretched-link" href="statushistory.php">Visit this page</a>
                  <div class="small text-white"><i class="fas fa-angle-right"></i></div>
                </div>
              </div>
            </div>
            <div class="col-xl-3 col-md-6">
              <div class="card highcliffe text-white mb-4">
                <div class="card-body">Update Student Status</div>
                <div class="card-footer d-flex align-items-center justify-content-between">
                  <a class="small text-white stretched-link" href="updatestatus.php">Visit this page</a>

```

```

        <div class="small text-white"><i class="fas fa-angle-right"></i></div>
    </div>
</div>
<div class="col-xl-3 col-md-6">
    <div class="card bg-danger text-white mb-4">
        <div class="card-body">Emergency</div>
        <div class="card-footer d-flex align-items-center justify-content-between">
            <a class="small text-white stretched-link" href="emergency.php">Visit this page</a>
            <div class="small text-white"><i class="fas fa-angle-right"></i></div>
        </div>
    </div>
</div>
</div>
</div>

<div class="row">
    <div class="col-xl-6">
        <div class="card mb-4">
            <div class="card-header" style="font-size:20px">
                <i class="fas fa-sign-in-alt fa-lg me-1" style="color: #522a5b !important;"></i>
                Students Signed IN
                <button class="btn btn-sm btn-secondary" onClick="window.location.reload();" style="float:right; margin-top:5px; margin-
right:5px;">Refresh</button>
                <!-- window.location.reload is a javascript function to reload the page -->
            </div>
            <div class="card-body">
                <table style="width:100%">
                    <thead>
                        <tr>
                            <th>Number Of Students:</th>
                        </tr>
                    </thead>
                    <tbody>
                        <?php
                            include ($IPATH."connectDB.php");

                            $sql = "SELECT status, COUNT(*) as count FROM status
                                    WHERE status = 1
                                    GROUP BY status;";
                            #Counts the number of students signed in

                            $result = mysqli_query($conn,$sql);

                            if (!$result) {
                                printf("Error: %s\n", mysqli_error($conn));
                                exit();

```

```

    }else{
        while($row = mysqli_fetch_array($result)){
            echo "<tr>";
            echo "<td>" . $row['count'] . "</td>";
            echo "<tr>";
        }
    }
    ?>
</tbody>
</table>
</div>
</div>
</div>
</div>
<div class="col-xl-6">
    <div class="card mb-4">
        <div class="card-header" style="font-size:20px">
            <i class="fas fa-sign-out-alt fa-lg me-1" style="color: #522a5b !important;"></i>
            Students Signed OUT
            <button class="btn btn-sm btn-secondary" onClick="window.location.reload();" style="float:right; margin-top:5px; margin-
right:5px;">Refresh</button>
        </div>
        <div class="card-body">
            <table style="width:100%">
                <thead>
                    <tr>
                        <th>Number Of Students:</th>
                    </tr>
                </thead>
                <tbody>
                    <?php
                        include ($IPATH."connectDB.php");

                        $sql = "SELECT status, COUNT(*) as count FROM status
                                WHERE status = 0
                                GROUP BY status;";
                        #Counts the number of students signed in

                        $result = mysqli_query($conn,$sql);

                        if (!$result) {
                            printf("Error: %s\n", mysqli_error($conn));
                            exit();
                        }else{
                            while($row = mysqli_fetch_array($result)){
                                echo "<tr>";
                                echo "<td>" . $row['count'] . "</td>";

```



```
<p class="lead">Internal Server Error</p>
```

```
<?php
function redirect($url) {
    ob_start();
    header('Location: '.$url);
    ob_end_flush();
    die();
}

$IPATH = $_SERVER["DOCUMENT_ROOT"]. "/PROJECT/Website/assets/IPATH/"; include($IPATH."connectDB.php");

$all = $_GET['all'];
echo $all . "<br>";

if ($_GET['all'] == 1){      #If delete all
    $sql = "DELETE FROM `signinhistory`";
    if($conn->query($sql) == True){
        echo "Table Data Deleted <br>";
        redirect("statushistory.php");
    } else{
        echo "Error, Could Not Delete Data <br>";
    }
} elseif($_GET['all'] == 2){  #If delete last month
    $prevMonth = date("Y") . "-" . (date("m")-1) . "-" . date("d");
    $prevMonth = strval($prevMonth);
    $sql = "DELETE FROM `signinhistory` WHERE DateChange <= '$prevMonth'";
    if($conn->query($sql) == True){
        echo "Table Data Deleted <br>";
        redirect("statushistory.php");
    } else{
        echo "Error, Could Not Delete Data <br>";
    }
} else{
    echo "General Error <br>";
}

?>
<a href="dashboard.php">
    <i style="color:#52a5b;" class="fas fa-arrow-left me-1"></i>
    Return to Dashboard
</a>
<a href="statushistory.php">
    <i class="fas fa-arrow-left me-1"></i>
    Return to Status History
</a>
```

```

        </div>
      </div>
    </div>
  </main>
</div>
<div id="layoutError_footer">
  <footer class="py-4 bg-light mt-auto">
    <div class="container-fluid px-4">
      <div class="d-flex align-items-center justify-content-between small">
        <div class="text-muted">Copyright &copy; Your Website 2021</div>
        <div>
          <a href="#">Privacy Policy</a>
          &middot;
          <a href="#">Terms & Conditions</a>
        </div>
      </div>
    </div>
  </footer>
</div>
</div>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js" crossorigin="anonymous"></script>
<script src="js/scripts.js"></script>
</body>
</html>

```

### deleteStaffOOP.php

```
<!-- This is the page visited when the delete staff form is submitted -->
```

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no" />
    <meta name="description" content="" />
    <meta name="author" content="" />
    <title>500 Error - Admin</title>
    <link href="css/styles.css" rel="stylesheet" />
    <script src="https://cdn.jsdelivr.net/npm/font-awesome@5.15.3/js/all.min.js" crossorigin="anonymous"></script>
  </head>
  <body>
    <div id="layoutError">
      <div id="layoutError_content">
        <main>

```

```

<div class="container">
  <div class="row justify-content-center">
    <div class="col-lg-6">
      <div class="text-center mt-4">
        <h1 class="display-1">500</h1>
        <p class="lead">Internal Server Error</p>
      <?php

class staff {                                #PHP OBJECT ORIENTED PROGRAMMING
  private $username;
  public $usernameErr;
  public $accountErr;

  function set_username($username){
    $this->username = $username;
  }

  function get_username(){
    return $this->username;
  }

  function get_adminNum(){
    return $this->adminNum;
  }

  function get_usernameErr(){
    return $this->usernameErr;
  }

  function get_accountErr(){
    return $this->accountErr;
  }

  function validateStaff(){
    $sendtoDB = True;

    if ($_SERVER["REQUEST_METHOD"] == "POST") {

      if (empty($_POST["username"])) {
        $this->usernameErr = "Username Required <br>";
        $sendtoDB = False;
      } else {
        $this->username = test_input($_POST["username"]);
      }
    }
    return $sendtoDB;
  }
}

```

```
function staffExists(){
    $exists = False;

    $IPATH = $_SERVER["DOCUMENT_ROOT"]. "/PROJECT/Website/assets/IPATH/";
    include ($IPATH."connectDB.php");

    $sql = "SELECT * FROM `admins` WHERE `username`='$this->username'";

    $result = $conn->query($sql);

    if (!$result) {
        printf("Error: %s\n", mysqli_error($conn));
        exit();
    }else{
        while($row = mysqli_fetch_array($result)){
            if($row['username'] == $this->username){
                $exists = True;
                return $exists;
            }
        }
        $this->accountErr = "Staff Account Does Not Exist <br>";
    }
}

function deletefromDB(){
    $IPATH = $_SERVER["DOCUMENT_ROOT"]. "/PROJECT/Website/assets/IPATH/";
    include ($IPATH."connectDB.php");

    $sql = "DELETE FROM `admins` WHERE `username`='$this->username'";

    if ($conn->query($sql) == TRUE) {
        echo "Deleted successfully";
        $conn->close();
        redirect("staffAccounts.php");
    } else {
        echo "Error: " . $sql . "<br>" . $conn->error;
    }
}

function test_input($data){
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
}
```



```

function redirect($url) {
    ob_start();
    header('Location: '.$url);
    ob_end_flush();
    die();
}

$staff = new staff();
$valid = $staff->validatestaff();

if ($valid == True){
    $delete = $staff->staffExists();
    if($delete == True){
        $staff->deletefromDB();
    }
}
echo "Cannot delete staff account due to these errors: <br>";
echo $staff-> get_usernameErr();
echo $staff-> get_accountErr();
?>
<a href="dashboard.php">
    <i style="color:#52a5b;" class="fas fa-arrow-left me-1"></i>
    Return to Dashboard
</a>
<a href="staffAccounts.php">
    <i class="fas fa-arrow-left me-1"></i>
    Return to staff Accounts
</a>
</div>
</div>
</div>
</div>
</main>
</div>
<div id="layoutError_footer">
    <footer class="py-4 bg-light mt-auto">
        <div class="container-fluid px-4">
            <div class="d-flex align-items-center justify-content-between small">
                <div class="text-muted">Copyright &copy; Your Website 2021</div>
                <div>
                    <a href="#">Privacy Policy</a>
                    &middot;
                    <a href="#">Terms & Conditions</a>
                </div>
            </div>
        </div>
    </div>
</div>
</div>

```

```

        </div>
    </div>
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js" crossorigin="anonymous"></script>
    <script src="js/scripts.js"></script>
</body>
</html>

```

### deleteStudentOOP.php

*<!-- This is the page visited when the delete student form is submitted -->*

```

<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="utf-8" />
        <meta http-equiv="X-UA-Compatible" content="IE=edge" />
        <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no" />
        <meta name="description" content="" />
        <meta name="author" content="" />
        <title>500 Error - Admin</title>
        <link href="css/styles.css" rel="stylesheet" />
        <script src="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.3/js/all.min.js" crossorigin="anonymous"></script>
    </head>
    <body>
        <div id="layoutError">
            <div id="layoutError_content">
                <main>
                    <div class="container">
                        <div class="row justify-content-center">
                            <div class="col-lg-6">
                                <div class="text-center mt-4">
                                    <h1 class="display-1">500</h1>
                                    <p class="lead">Internal Server Error</p>
                                    <?php
                                        class student {          #PHP OBJECT ORIENTED PROGRAMMING
                                            private $firstname;
                                            private $surname;
                                            private $candidateNumber;
                                            private $adminNum;
                                            public $firstnameErr;
                                            public $surnameErr;
                                            public $candidateNumberErr;
                                            public $adminNumErr;
                                            public $accountErr;

                                            function set_firstname($firstname){
                                                $this->firstname = $firstname;

```

```
}

function set_surname($surname){
    $this->surname = $surname;
}

function set_candidateNumber($candidateNumber){
    $this->candidateNumber = $candidateNumber;
}

function set_adminNum($adminNum){
    $this->adminNum = $adminNum;
}

function get_firstname(){
    return $this->firstname;
}

function get_surname(){
    return $this->surname;
}

function get_candidateNumber(){
    return $this->candidateNumber;
}

function get_adminNum(){
    return $this->adminNum;
}

function get_firstnameErr(){
    return $this->firstnameErr;
}

function get_surnameErr(){
    return $this->surnameErr;
}

function get_candidateNumberErr(){
    return $this->candidateNumberErr;
}

function get_adminNumErr(){
    return $this->adminNumErr;
}

function get_accountErr(){
```

```
        return $this->accountErr;
    }

    function validateStudent(){
        $sendtoDB = True;

        if ($_SERVER["REQUEST_METHOD"] == "POST") {

            if (empty($_POST["fname"])) {
                $this->firstnameErr = "First Name Required <br>";
                $sendtoDB = False;
            } elseif (ctype_alpha($_POST["fname"])) {
                $this->firstname = test_input($_POST["fname"]);
            } else {
                $this->firstnameErr = "Only letters allowed in first name <br>";
                $sendtoDB = False;
            }

            if (empty($_POST["sname"])) {
                $this->surnameErr = "Surname Required <br>";
                $sendtoDB = False;
            } elseif (ctype_alpha($_POST["sname"])) {
                $this->surname = test_input($_POST["sname"]);
            } else {
                $this->surnameErr = "Only letters allowed in surname <br>";
                $sendtoDB = False;
            }

            if (empty($_POST["canNum"])) {
                $this->candidateNumberErr = "Candidate Number Required <br>";
                $sendtoDB = False;
            } elseif (strlen($_POST["canNum"]) == 4) {
                $this->candidateNumber = test_input($_POST["canNum"]);
            } else {
                $this->candidateNumberErr = "Candidate number must be 4 numbers <br>";
                $sendtoDB = False;
            }

            if (empty($_POST["aNum"])) {
                $this->adminNumErr = "Admin Number Required <br>";
                $sendtoDB = False;
            } elseif (strlen($_POST["aNum"]) == 6) {
                $this->adminNum = test_input($_POST["aNum"]);
            } else {
                $this->adminNumErr = "Admin number must be 6 numbers <br>";
                $sendtoDB = False;
            }
        }
    }
}
```

```

        return $sendtoDB;
    }

    function studentExists(){
        $exists = False;

        $IPATH = $_SERVER["DOCUMENT_ROOT"]. "/PROJECT/Website/assets/IPATH/";
        include ($IPATH."connectDB.php");

        $result = $conn->prepare("SELECT * FROM `students` WHERE `firstname`=? AND `surname`=? AND `candidateNumber`=? AND
`misID`=?)");

        $result->bind_param('ssii', $firstname, $surname, $candidateNumber, $adminNumber);      #PARAMETER BINDING

        $firstname = $this->get_firstname();
        $surname = $this->get_surname();
        $candidateNumber = $this->get_candidateNumber();
        $adminNumber = $this->get_adminNum();

        $result->execute();
        $result->store_result();

        $row = $result->num_rows();

        if (!$result) {
            printf("Error: %s\n", mysqli_error($conn));
            exit();
        }else{
            if($row > 0){
                $exists = True;
                return $exists;
            }else{
                $this->accountErr = "Student Account Does Not Exist <br>";
            }
        }
    }
}

function deletefromDB(){

    $IPATH = $_SERVER["DOCUMENT_ROOT"]. "/PROJECT/Website/assets/IPATH/";
    include ($IPATH."connectDB.php");

    $sql2 = "SELECT * FROM `statusids` WHERE `CandidateNumber`=$this->candidateNumber;";

    $result = mysqli_query($conn,$sql2);
    while($row = mysqli_fetch_array($result)){
        $this->statusID = $row['StatusID'];
    }
}

```

```
$sql = "DELETE FROM `students` WHERE `CandidateNumber`=$this->candidateNumber AND `misID`=$this->adminNum AND
`Surname`='$this->surname' AND `Firstname`='$this->firstname';";

$sql .= "DELETE FROM `status` WHERE `StatusID`=$this->statusID;";

$sql .= "DELETE FROM `statusids` WHERE `CandidateNumber`=$this->candidateNumber;";

if ($conn->multi_query($sql) == TRUE) {
    #MULTIQUERY used for multiple delete statements
    echo "Deleted successfully";
    $conn->close();
    redirect("studentAccounts.php");
} else {
    echo "Error: " . $sql . "<br>" . $conn->error;
}
}

function test_input($data){
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
}

function redirect($url) {
    ob_start();
    header('Location: '.$url);
    ob_end_flush();
    die();
}

$student = new student();
$valid = $student->validateStudent();

if ($valid == True){
    $delete = $student->studentExists();
    if($delete == True){
        $student->deletefromDB();
    }
}
echo "Cannot delete student account due to these errors: <br>";
echo $student-> get_firstnameErr();
echo $student-> get_surnameErr();
echo $student-> get_candidateNumberErr();
```



```
$username = "username";
$password = "password";
$dbname = "database";

$conn = new mysqli($servername, $username, $password, $dbname); #using SQLi
if($conn->connect_error){
    die("Connection failed: " . $conn->connect_error); #If an error occurs then it is output
}

$studentArray = array();

$sql = "SELECT students.Firstname, students.Surname, students.TutorGroup, students.YearGroup FROM `status`
LEFT JOIN `statusids` ON status.StatusID = statusids.StatusID
LEFT JOIN `students` ON statusids.CandidateNumber = students.CandidateNumber
WHERE status.status = 1
ORDER BY students.TutorGroup ASC, students.Surname;";      #USE OF SQL JOIN

$result = $conn->query($sql);

$j = 0;

while($row = mysqli_fetch_array($result)) {                  #USE OF 2D ARRAY
    $studentArray[$j]['firstname'] = $row['Firstname'];
    $studentArray[$j]['surname'] = $row['Surname'];
    $studentArray[$j]['tutorgroup'] = $row['TutorGroup'];
    $studentArray[$j]['yeargroup'] = $row['YearGroup'];
    $j+=1;
}

$filename = date("D-M-Y H.i") . ".csv";
$myfile = fopen($filename, "w") or die("Unable to open file!");      #USE OF FILE-HANDLING

$txt = "This is all students with the current status of IN \n\n";
fwrite($myfile,$txt);

for ($i = 0 ; $i < count($studentArray); $i++) {
    $txt = $studentArray[$i]['tutorgroup'] . " , " . $studentArray[$i]['surname'] . " , " . $studentArray[$i]['firstname'] . " , " .
    $studentArray[$i]['yeargroup'] . "\n";
    fwrite($myfile, $txt);
}

fclose($myfile);

#assigns metadata for the file being force downloaded
header('Content-Description: File Transfer');
header('Content-Disposition: attachment; filename='.basename($filename));
```



```

header('Expires: 0');
header('Cache-Control: must-revalidate');
header('Pragma: public');
header('Content-Length: ' . filesize($filename));
header("Content-Type: text/plain");
readfile($filename);
?>

```

### layout-sidenav-light.html

```
<!-- This is the page visited to change theme to light mode -->
```

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no" />
    <meta name="description" content="" />
    <meta name="author" content="" />
    <title>Sidenav Light - SB Admin</title>
    <link href="css/styles.css" rel="stylesheet" />
    <script src="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.3/js/all.min.js" crossorigin="anonymous"></script>
  </head>
  <body class="sb-nav-fixed">
    <nav class="sb-topnav navbar navbar-expand navbar-dark bg-dark">
      <!-- Navbar Brand-->
      <a class="navbar-brand ps-3" href="dashboard.php">Start Bootstrap</a>
      <!-- Sidebar Toggle-->
      <button class="btn btn-link btn-sm order-1 order-lg-0 me-4 me-lg-0" id="sidebarToggle" href="#"><i class="fas fa-bars"></i></button>
      <!-- Navbar Search-->
      <form class="d-none d-md-inline-block form-inline ms-auto me-0 me-md-3 my-2 my-md-0">
        <div class="input-group">
          <input class="form-control" type="text" placeholder="Search for..." aria-label="Search for..." aria-describedby="btnNavbarSearch" />
          <button class="btn btn-primary" id="btnNavbarSearch" type="button"><i class="fas fa-search"></i></button>
        </div>
      </form>
      <!-- Navbar-->
      <ul class="navbar-nav ms-auto ms-md-0 me-3 me-lg-4">
        <li class="nav-item dropdown">
          <a class="nav-link dropdown-toggle" id="navbarDropdown" href="#" role="button" data-bs-toggle="dropdown" aria-expanded="false"><i
class="fas fa-user fa-fw"></i></a>
          <ul class="dropdown-menu dropdown-menu-end" aria-labelledby="navbarDropdown">
            <li><a class="dropdown-item" href="#">Settings</a></li>
            <li><a class="dropdown-item" href="#">Activity Log</a></li>
            <li><hr class="dropdown-divider" /></li>
            <li><a class="dropdown-item" href="#">Logout</a></li>
          </ul>
        </li>
      </ul>
    </nav>

```

```

        </ul>
    </li>
</ul>
</nav>
<div id="layoutSidenav">
    <div id="layoutSidenav_nav">
        <nav class="sb-sidenav accordion sb-sidenav-light" id="sidenavAccordion">
            <div class="sb-sidenav-menu">
                <div class="nav">
                    <div class="sb-sidenav-menu-heading">Core</div>
                    <a class="nav-link" href="dashboard.php">
                        <div class="sb-nav-link-icon"><i class="fas fa-tachometer-alt"></i></div>
                        Dashboard
                    </a>
                    <div class="sb-sidenav-menu-heading">Interface</div>
                    <a class="nav-link collapsed" href="#" data-bs-toggle="collapse" data-bs-target="#collapseLayouts" aria-expanded="false" aria-
controls="collapseLayouts">
                        <div class="sb-nav-link-icon"><i class="fas fa-columns"></i></div>
                        Layouts
                        <div class="sb-sidenav-collapse-arrow"><i class="fas fa-angle-down"></i></div>
                    </a>
                    <div class="collapse" id="collapseLayouts" aria-labelledby="headingOne" data-bs-parent="#sidenavAccordion">
                        <nav class="sb-sidenav-menu-nested nav">
                            <a class="nav-link" href="layout-static.html">Static Navigation</a>
                            <a class="nav-link" href="layout-sidenav-light.html">Light Sidenav</a>
                        </nav>
                    </div>
                    <a class="nav-link collapsed" href="#" data-bs-toggle="collapse" data-bs-target="#collapsePages" aria-expanded="false" aria-
controls="collapsePages">
                        <div class="sb-nav-link-icon"><i class="fas fa-book-open"></i></div>
                        Pages
                        <div class="sb-sidenav-collapse-arrow"><i class="fas fa-angle-down"></i></div>
                    </a>
                    <div class="collapse" id="collapsePages" aria-labelledby="headingTwo" data-bs-parent="#sidenavAccordion">
                        <nav class="sb-sidenav-menu-nested nav accordion" id="sidenavAccordionPages">
                            <a class="nav-link collapsed" href="#" data-bs-toggle="collapse" data-bs-target="#pagesCollapseAuth" aria-expanded="false"
aria-controls="pagesCollapseAuth">
                                Authentication
                                <div class="sb-sidenav-collapse-arrow"><i class="fas fa-angle-down"></i></div>
                            </a>
                            <div class="collapse" id="pagesCollapseAuth" aria-labelledby="headingOne" data-bs-parent="#sidenavAccordionPages">
                                <nav class="sb-sidenav-menu-nested nav">
                                    <a class="nav-link" href="login.html">Login</a>
                                    <a class="nav-link" href="createaccount.html">Create Account</a>
                                    <a class="nav-link" href="password.html">Forgot Password</a>
                                </nav>
                            </div>
                        </nav>
                    </div>
                </div>
            </div>
        </nav>
    </div>

```

```

        <a class="nav-link collapsed" href="#" data-bs-toggle="collapse" data-bs-target="#pagesCollapseError" aria-
expanded="false" aria-controls="pagesCollapseError">
            Error
            <div class="sb-sidenav-collapse-arrow"><i class="fas fa-angle-down"></i></div>
        </a>
        <div class="collapse" id="pagesCollapseError" aria-labelledby="headingOne" data-bs-parent="#sidenavAccordionPages">
            <nav class="sb-sidenav-menu-nested nav">
                <a class="nav-link" href="401.html">401 Page</a>
                <a class="nav-link" href="404.html">404 Page</a>
                <a class="nav-link" href="500.html">500 Page</a>
            </nav>
        </div>
    </nav>
</div>
<div class="sb-sidenav-menu-heading">Addons</div>
<a class="nav-link" href="charts.html">
    <div class="sb-nav-link-icon"><i class="fas fa-chart-area"></i></div>
    Charts
</a>
<a class="nav-link" href="tables.php">
    <div class="sb-nav-link-icon"><i class="fas fa-table"></i></div>
    Tables
</a>
</div>
</div>
<div class="sb-sidenav-footer">
    <div class="small">Logged in as:</div>
    Start Bootstrap
</div>
</nav>
</div>
<div id="layoutSidenav_content">
    <main>
        <div class="container-fluid px-4">
            <h1 class="mt-4">Sidenav Light</h1>
            <ol class="breadcrumb mb-4">
                <li class="breadcrumb-item"><a href="dashboard.php">Dashboard</a></li>
                <li class="breadcrumb-item active">Sidenav Light</li>
            </ol>
            <div class="card mb-4">
                <div class="card-body">
                    This page uses the light side navigation option. By appending the
                    <code>.sb-sidenav-light</code>
                    class to the
                    <code>.sb-sidenav</code>
                    class, the side navigation will take on a light color scheme. The
                    <code>.sb-sidenav-dark</code>
                    is also available for a darker option.
                </div>
            </div>
        </div>
    </main>
</div>

```

```

        </div>
      </div>
    </div>
  </main>
  <footer class="py-4 bg-light mt-auto">
    <div class="container-fluid px-4">
      <div class="d-flex align-items-center justify-content-between small">
        <div class="text-muted">Copyright &copy; Your Website 2021</div>
        <div>
          <a href="#">Privacy Policy</a>
          &middot;
          <a href="#">Terms & Conditions</a>
        </div>
      </div>
    </div>
  </footer>
</div>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js" crossorigin="anonymous"></script>
<script src="js/scripts.js"></script>
</body>
</html>

```

### login.php

```
<!-- This is the page visited for the user to log in -->
```

```

<!DOCTYPE html>
<html lang="en">
  <?php
    session_abort();          #Restarts session
    clearstatcache();
  ?>

  <head>
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no" />
    <meta name="description" content="" />
    <meta name="author" content="" />
    <title>Login - Highcliffe Admin</title>
    <link href="css/styles.css" rel="stylesheet" />
    <script src="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.3/js/all.min.js" crossorigin="anonymous"></script>
  </head>

```

```
<style>
  .input {
    position: relative;
    display: block;
    background: #ddd;

    &::after {
      content: "▼";
      position: absolute;
      right: 1.5rem;
      top: 1.6rem;
      transition: 300ms transform 200ms;
    }
    &:active::after {
      transform: rotate(-180deg);
    }
    &__field {
      display: block;
      opacity: 1;
      width: 100%;
      box-sizing: border-box;
      border: 0;
      background: transparent;
      appearance: none;
      padding: 2rem 1.5rem 1rem;
      border-bottom: 2px solid purple;
      border-radius: 0;
      transition-delay: 0ms;
      will-change: color;
      transition: 200ms color linear;

      &:invalid {
        color: transparent;
        transition: 200ms color linear 100ms;
      }

      &:valid + .input__label,
      &:focus:valid + .input__label {
        transform: scale(0.75) translate(0.5em, -10%);
        transition: 200ms transform ease-out;
        color: purple;
      }
    }
    &__label {
      position: absolute;
      left: 0;
      top: 0;
    }
  }

```

```
    right: 0;
    padding: 1.5rem 1.5rem 0;
    pointer-events: none;
    transform-origin: 0 0;
    transition: 200ms transform ease-out 200ms;
    will-change: transform;
  }
}
.form-control:focus, .dataTable-input:focus {
  color: #212529;
  background-color: #fff;
  border-color: #522a5b;
  outline: 0;
  box-shadow: 0 0 0 0.2rem rgba(82, 42, 91, 0.65);
}

.form-check-input:checked {
  background-color: #522a5b !important;
  border-color: #522a5b !important;
}

.form-check-input:focus {
  border-color: #522a5b;
  outline: 0;
  box-shadow: 0 0 0 0.15rem rgba(82, 42, 91, 0.65);
}
}
</style>

<script>
function showPassword() {
  //Javascript function used to toggle between password view and text view
  var x = document.getElementById("inputPassword");
  if (x.type === "password") {
    x.type = "text";
  } else {
    x.type = "password";
  }
}
</script>

<body class="highcliffe">
  <div id="layoutAuthentication">
    <div></div>
    <div id="layoutAuthentication_content">
      <main>
```

```

<div class="container">
  <div class="row justify-content-center">
    <div class="col-lg-5">
      <form method="post" action="authenticate.php">
        <div class="card shadow-lg border-0 rounded-lg mt-5">
          <div class="card-header"><h3 class="text-center font-weight-light my-4">Login</h3></div>
          <div class="card-body">
            <div class="login">
              <div class="form-floating mb-3">
                <input class="form-control" id="inputUsername" name="username" type="username" placeholder="Username"
autocomplete="off" required/>
                <label for="inputUsername">Username</label>
              </div>
              <div class="form-floating mb-3">
                <input class="form-control" id="inputPassword" name="password" type="password" placeholder="Password"
required/>
                <label for="inputPassword">Password</label>
              </div>
              <div class="form-check mb-3">
                <input class="form-check-input" id="inputRememberPassword" type="checkbox" value="" />
                <label class="form-check-label" for="inputRememberPassword">Remember Password</label>
                <div style="float:right;">
                  <label class="form-check-label" for="inputShowPassowrd">Show Password</label>
                  <input class="form-check-input" id="inputShowPassowrd" type="checkbox" onclick="showPassword()">
                </div>
              </div>
            </div>
          </div>
          <div class="card-footer text-center py-3">
            <div class="d-flex align-items-center justify-content-between mt-4 mb-0">
              <a class="small" href="password.html">Forgot Password?</a>
              <button type="submit" id="submitButton" class="btn btn-highcliffe" value="Submit">Login</button>
            </div>
          </div>
        </div>
      </form>
    </div>
  </div>
</div>
</main>
</div>
<div id="layoutAuthentication_footer">
  <footer class="py-4 bg-light mt-auto">
    <div class="container-fluid px-4">
      <div class="d-flex align-items-center justify-content-between small">
        <div class="text-muted">Copyright &copy; Your Website 2021</div>
      </div>
    </div>
  </footer>
</div>

```

```

        <a href="#">Privacy Policy</a>
        &middot;
        <a href="#">Terms & Conditions</a>
    </div>
</div>
</div>
</footer>
</div>
</div>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js" crossorigin="anonymous"></script>
<script src="js/scripts.js"></script>
</body>
</html>

```

### logout.php

<!-- This is the page visited when the user clicks the logout button -->

```

<?php
    Session_start();          #Destroys the current session to remove signed in variables
    Session_destroy();
    header('Location: ' . "login.php");    #redirects the user to the login screen

?>

```

### password.html

<!-- This is the page visited for a staff member to reset their password -->

```

<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="utf-8" />
        <meta http-equiv="X-UA-Compatible" content="IE=edge" />
        <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no" />
        <meta name="description" content="" />
        <meta name="author" content="" />
        <title>Password Reset - SB Admin</title>
        <link href="css/styles.css" rel="stylesheet" />
        <script src="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.3/js/all.min.js" crossorigin="anonymous"></script>
    </head>
    <body class="bg-primary">
        <div id="layoutAuthentication">
            <div id="layoutAuthentication_content">

```



```

<main>
  <div class="container">
    <div class="row justify-content-center">
      <div class="col-lg-5">
        <div class="card shadow-lg border-0 rounded-lg mt-5">
          <div class="card-header"><h3 class="text-center font-weight-light my-4">Password Recovery</h3></div>
          <div class="card-body">
            <div class="small mb-3 text-muted">Enter your email address and we will send you a link to reset your password.</div>
            <form>
              <div class="form-floating mb-3">
                <input class="form-control" id="inputEmail" type="email" placeholder="name@example.com" />
                <label for="inputEmail">Email address</label>
              </div>
              <div class="d-flex align-items-center justify-content-between mt-4 mb-0">
                <a class="small" href="login.html">Return to login</a>
                <a class="btn btn-primary" href="login.html">Reset Password</a>
              </div>
            </form>
          </div>
          <div class="card-footer text-center py-3">
            <div class="small">Need an account? Talk to the system administrator</div>
          </div>
        </div>
      </div>
    </div>
  </div>
</main>
</div>
<div id="layoutAuthentication_footer">
  <footer class="py-4 bg-light mt-auto">
    <div class="container-fluid px-4">
      <div class="d-flex align-items-center justify-content-between small">
        <div class="text-muted">Copyright &copy; Your Website 2021</div>
        <div>
          <a href="#">Privacy Policy</a>
          &middot;
          <a href="#">Terms & Conditions</a>
        </div>
      </div>
    </div>
  </footer>
</div>
</div>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js" crossorigin="anonymous"></script>
<script src="js/scripts.js"></script>
</body>
</html>

```

staffAccounts.php

```

<!-- This is the page visited to view and edit staff accounts -->

<!DOCTYPE html>
<?php
    function redirecter($url){
        ob_start();
        header('Location: '. $url);
        ob_end_flush();
        die();
    }

    session_start();      #This is an authorisation check
    if (empty($_SESSION['username'])){
#If the user has signed in using login.php then $_SESSION['username'] will be set to their username
        $username = "Unknown Account";
        redirecter("login.php");
    } else{
        $access = $_SESSION['access'];
#This assigns whether the current user is an admin-user or staff-user
        if ($access != "A"){
            redirecter("401.html");
#Redirects to "Unauthorised" error if not an admin account
        }
    }

?>

<html lang="en">
    <title>Staff Accounts</title>

    <?php $IPATH = $_SERVER["DOCUMENT_ROOT"]. "/PROJECT/Website/assets/IPATH/"; include($IPATH."headernav.php");?>
    <META HTTP-EQUIV="CACHE-CONTROL" CONTENT="NO-CACHE" />
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
    <link href="css/switch.css?version=55" rel="stylesheet" />
    <link href="//cdn.datatables.net/1.11.3/css/jquery.dataTables.min.css" rel="stylesheet" />
    <script src="//cdn.datatables.net/1.11.3/js/jquery.dataTables.min.js"></script>

    <script>    //Javascript functions to animate opening and closing the create and delete forms
        function createOpen(){
            deleteClose();
            $('#newText').hide();
            $('#createButton').animate({width: '0%'});
            $('#createStaffForm').slideDown("slow");
        };

```

```
function createClose(){
    $("#createStaffForm").slideUp("fast");
    $("#createButton").animate({width: '100%'});
    $('#newText').show();
};

function deleteOpen(){
    createClose();
    $('#deleteText').hide();
    $('#deleteButton').animate({width: '0%'});
    $("#deleteStaffForm").slideDown("slow");
};

function deleteClose(){
    $("#deleteStaffForm").slideUp("fast");
    $("#deleteButton").animate({width: '100%'});
    $('#deleteText').show();
};
</script>

<style>
    body {
        padding: 1rem;
    }
    .input {
        position: relative;
        display: block;
        background: #ddd;
    }
    &::after {
        content: "▼";
        position: absolute;
        right: 1.5rem;
        top: 1.6rem;
        transition: 300ms transform 200ms;
    }
    &:active::after {
        transform: rotate(-180deg);
    }
    &__field {
        display: block;
        opacity: 1;
        width: 100%;
        box-sizing: border-box;
        border: 0;
        background: transparent;
        appearance: none;
    }
```

```
padding: 2rem 1.5rem 1rem;
border-bottom: 2px solid purple;
border-radius: 0;
transition-delay: 0ms;
will-change: color;
transition: 200ms color linear;

&:invalid {
  color: transparent;
  transition: 200ms color linear 100ms;
}

&:valid + .input__label,
&:focus:valid + .input__label {
  transform: scale(0.75) translate(0.5em, -10%);
  transition: 200ms transform ease-out;
  color: purple;
}
}
&__label {
  position: absolute;
  left: 0;
  top: 0;
  right: 0;
  padding: 1.5rem 1.5rem 0;
  pointer-events: none;
  transform-origin: 0 0;
  transition: 200ms transform ease-out 200ms;
  will-change: transform;
}
}
.form-control:focus, .dataTable-input:focus {
  color: #212529;
  background-color: #fff;
  border-color: #522a5b;
  outline: 0;
  box-shadow: 0 0 0 0.2rem rgba(82, 42, 91, 0.65);
}

.form-check-input:checked {
  background-color: #522a5b !important;
  border-color: #522a5b !important;
}

.form-check-input:focus {
  border-color: #522a5b;
  outline: 0;
  box-shadow: 0 0 0 0.15rem rgba(82, 42, 91, 0.65);
}
```

```

}

.body{
  overflow-y: auto !important;
}

table.dataTable thead .sorting_asc {
  background-image: url("/assets/img/UpArrow.png")
}

table.dataTable thead .sorting_desc {
  background-image: url("/assets/img/DownArrow.png")
}

.button-group{
  display: inline-flex;
}

.button{
  padding: 10px;
  outline: none;
  border: none;
  cursor: pointer;
  border-right: 1px solid #cccccc;
  background: #522a5b;
  color: #ffffff;
  width: 100%;
}

.button:last-child{
  border-right: none;
}
</style>

<body>
  <div id="layoutSidenav_content">
    <main>
      <div class="container-fluid px-4">
        <h1 class="mt-4">Staff Accounts</h1>
        <ol class="breadcrumb mb-4">
          <li class="breadcrumb-item active">The Highcliffe Sixth Form Staff Account Home</li>
        </ol>
      </div>

      <div class="row-fluid">
        <div class="container-fluid px-4 mt-3 button-group">
          <button input="submit" id="createButton" onclick="createOpen()" class="button"><span id="newText">New Staff
Account</span></button>

```

```

        <button input="submit" id="deleteButton" onclick="deleteOpen()" class="button"><span id="deleteText">Delete Staff
Account</span></button>
    </div>
</div>

<div class="container" id="createStaffForm" style="display: none;">
    <div class="row">
        <div class="col-lg-12" style="float:centre;" style="border-radius: 2em !important;">

            <form method="post" action="createStaffOOP.php">

                <div class="card shadow-lg border-0 rounded-sm mt-3">
                    <div class="card-header highcliffe">
                        <a input="" class="btn btn-dark" style="float:left;" onclick="createClose()">Back</a>
                        <h3 class="text-center font-weight-light my-4" style="color:#ffffff !important;">Create Staff Account</h3>
                    </div>

                    <div class="card-body">
                        <div class="login">
                            <div class="row">
                                <div class="col-lg-5">
                                    <div class="form-floating mb-3">
                                        <input class="form-control" name="username" id="inputusername" type="text" placeholder="Username"
autocomplete="off" required/>

                                        <label for="inputusername">Username</label>
                                    </div>
                                </div>
                                <div class="col-lg-5">
                                    <div class="form-floating mb-3">
                                        <input class="form-control" name="password" id="inputpassword" type="text" placeholder="Password"
autocomplete="off" required/>

                                        <label for="inputpassword">Password <i class="muted" style="font-size: 10px;">(Include Special
Character)</i></label>
                                    </div>
                                </div>
                                <div class="col-lg-2">
                                    <div class="form-floating mb-3">
                                        <select class="form-control" id="access" name="access" placeholder="Choose Access Level"
autocomplete="off" required>

                                            <option value="S">Staff</option>
                                            <option value="A">Admin</option>
                                        </select>
                                        <label for="access">Access Level</span>
                                    </div>
                                </div>
                            </div>
                        </div>
                    </div>
                </div>
            </form>
        </div>
    </div>
</div>

```

```

        </div>
        <div class="card-footer text-center">
            <div class="mb-2" style="float: right !important;">
                <button type="submit" id="submitButton" class="btn btn-highcliffe" value="Submit">Create Account</button>
            </div>
        </div>
    </div>
</form>
</div>
</div>
</div>
</div>
<div class="container" id="deleteStaffForm" style="display: none;">
    <div class="row">
        <div class="col-lg-12" style="float: centre;" style="border-radius: 2em !important;">
            <form method="post" action="deleteStaffOOP.php" onsubmit="return confirm('Are you sure you want to delete this staff
account?');">

                <div class="card shadow-lg border-0 rounded-sm mt-3">
                    <div class="card-header highcliffe">
                        <a input="" class="btn btn-dark" style="float: left;" onclick="deleteClose()">Back</a>
                        <h3 class="text-center font-weight-light my-4" style="color: #ffffff !important;">Delete Staff Account</h3>
                    </div>

                    <div class="card-body">
                        <div class="login">
                            <div class="row">
                                <div class="col-lg-12">
                                    <div class="form-floating mb-3">
                                        <input class="form-control" name="username" id="inputusername" type="text" placeholder="Username"
autocomplete="off" required/>
                                        <label for="inputusername">Username</label>
                                    </div>
                                </div>
                            </div>
                        </div>
                    </div>
                    <div class="card-footer text-center">
                        <div class="mb-2" style="float: right !important;">
                            <button type="submit" id="submitButton" class="btn btn-highcliffe" value="Submit">Delete Account</button>
                        </div>
                    </div>
                </div>
            </form>
        </div>
    </div>
</div>

```

```

    </div>
</div>

<div class="row mt-5 px-4">
    <div class="col-xl-12">
        <div class="card mb-4">
            <div class="card-header" style="font-size: 20px">
                <span style="color: #522a5b"><i class="far fa-address-card me-1"></i></span>
                All Staff Accounts

                <button class="btn btn-sm btn-highcliffe" onClick="window.location.reload();" style="float:right; margin-top:5px; margin-
right:5px;">Refresh</button>
            </div>

            <div class="card-body">
                <table id="myTable" class="display" style="width:100%">
                    <thead>
                        <tr>
                            <th>Username</th>
                            <th>Staff Type</th>
                        </tr>
                    </thead>

                    <tbody>
                        <?php
                            $IPATH = $_SERVER["DOCUMENT_ROOT"]. "/PROJECT/Website/assets/IPATH/";
                            include($IPATH."connectDB.php");

                            $sql = "SELECT username, access FROM `admins`";

                            $result = mysqli_query($conn,$sql);
                            if (!$result) {
                                printf("Error: %s\n", mysqli_error($conn));
                                exit();
                            }else{
                                while($row = mysqli_fetch_array($result)){
                                    echo "<tr>";
                                    echo "<td>" . $row['username'] . "</td>";
                                    echo "<td>" . $row['access'] . "</td>";
                                    echo "</tr>";
                                }
                            }

                            $conn->close();
                        <?>
                    </tbody>
                </table>
            </div>
        </div>
    </div>
</div>

```



```

        </div>
    </div>
</div>

</main>

<script>
    $(document).ready(function () {
//USE OF DATATABLES, a Java plugin for sortable tables
        $('#myTable').DataTable({
            "paging": false,
            "order": [[1, 'asc'], [2, 'asc']]
        });
    });
</script>

<?php include($IPATH."footer.php"); ?>
</div>
</body>
</html>

```

### statusHistory.php

```

<!-- This is the page visited to view the sign in history -->

<!DOCTYPE html>
<html lang="en">
    <title>Student Status History</title>

    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
    <link href="css/switch.css?version=55" rel="stylesheet" />
    <link href="//cdn.datatables.net/1.11.3/css/jquery.dataTables.min.css" rel="stylesheet" />
    <script src="//cdn.datatables.net/1.11.3/js/jquery.dataTables.min.js"></script>

    <?php $IPATH = $_SERVER["DOCUMENT_ROOT"]. "/PROJECT/Website/assets/IPATH/";
    include($IPATH."headernav.php");
    ?>

    <body>
        <div id="layoutSidenav_content">
            <main>
                <div class="container-fluid px-4">

```

```

<h1 class="mt-4">Student Status History</h1>
<ol class="breadcrumb mb-4">
  <li class="breadcrumb-item active">Welcome to the hub of the Highcliffe Sixth Form Status History</li>
</ol>

<div class="row">
  <div class="col-xl-12">
    <div class="card mb-4">
      <div class="card-header" style="font-size: 20px">
        <span style="color: #522a5b"><i class="fas fa-history me-1"></i></span>
        Status History

        <a style="float:right; margin-top: 5px;" class="btn btn-sm btn-dark" href="deletehistory.php?all=1" onclick="return
confirm('Are you sure you want to delete all status history?'); //Java popup for confirming form submit">Delete All History</a>
        <a style="float:right; margin-top: 5px; margin-right: 5px;" class="btn btn-sm btn-dark" href="deletehistory.php?all=2"
onclick="return confirm('Are you sure you want to delete status history from longer than a month ago?');">Delete Previous Month</a>
        <button class="btn btn-sm btn-highcliffe" onclick="window.location.reload(); //Java reload page to update table"
style="float:right; margin-top:5px; margin-right:5px;">Refresh</button>
      </div>

      <div class="card-body">
        <table id="myTable" class="display" style="width:100%">
          <thead>
            <tr>
              <th>Candidate Number</th>
              <th>Surname</th>
              <th>Firstname</th>
              <th>Tutor Group</th>
              <th>Year Group</th>
              <th>Date Changed</th>
              <th>Time Changed</th>
              <th>Status</th>
            </tr>
          </thead>

          <tbody>
            <?php

            include ($IPATH."connectDB.php");

            $sql = "SELECT signinhistory.DateChange, signinhistory.TimeChange, students.CandidateNumber, students.Firstname,
students.Surname, students.TutorGroup, students.YearGroup, signinhistory.Status FROM `signinhistory`
LEFT JOIN `students` ON signinhistory.CandidateNumber = students.CandidateNumber
ORDER BY signinhistory.DateChange DESC;";

            $result = mysqli_query($conn,$sql);
            if (!$result) {
              printf("Error: %s\n", mysqli_error($conn));

```

```

        exit();
    }else{
        while($row = mysqli_fetch_array($result)){
            if($row['Status'] == 1){$pStatus = "IN";}elseif($row['Status'] == 0){$pStatus = "OUT";}else{$pStatus =
"Error";}};

            echo "<tr>";
            echo "<td>" . $row['CandidateNumber'] . "</td>";
            echo "<td>" . $row['Surname'] . "</td>";
            echo "<td>" . $row['Firstname'] . "</td>";
            echo "<td>" . $row['TutorGroup'] . "</td>";
            echo "<td>" . $row['YearGroup'] . "</td>";
            echo "<td>" . $row['DateChange'] . "</td>";
            echo "<td>" . $row['TimeChange'] . "</td>";
            echo "<td>" . $pStatus . "</td>";
            echo "</tr>";
        }
    }

    $conn->close();
    ?>
</tbody>
</table>
</div>
</div>
</div>
</div>
</div>
</div>
</main>

<script>
    $(document).ready(function () {
        $('#myTable').DataTable({
            "paging": false,
            "order": [[ 5 , 'desc' ], [6, 'desc']]
        });
    });
</script>

<?php include($IPATH."footer.php"); ?>
</div>
</body>
</html>

```

### studentAccounts.php

```
<!-- This is the page which displays student accounts and allows creation and deletion -->
```

```
<!DOCTYPE html>
<?php
    function redirecter($url){
        ob_start();
        header('Location: '. $url);
        ob_end_flush();
        die();
    }

    session_start();
    if (empty($_SESSION['username'])){
        $username = "Unknown Account";
        redirecter("login.php");
    } else{
        $access = $_SESSION['access'];
        if ($access != "A"){
            redirecter("401.html");
        }
    }
}

?>

<?php $IPATH = $_SERVER["DOCUMENT_ROOT"]. "/PROJECT/Website/assets/IPATH/"; include($IPATH."headernav.php");?>

<html lang="en">
    <title>Student Accounts</title>

    <META HTTP-EQUIV="CACHE-CONTROL" CONTENT="NO-CACHE" />
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
    <link href="css/switch.css?version=55" rel="stylesheet" />
    <link href="//cdn.datatables.net/1.11.3/css/jquery.dataTables.min.css" rel="stylesheet" />
    <script src="//cdn.datatables.net/1.11.3/js/jquery.dataTables.min.js"></script>

    <script>
        function createOpen(){
            updateClose();
            deleteClose();
            $('#newText').hide();
            $('#createButton').animate({width: '0%'});
            $('#createStudentForm').slideDown("slow");
        };

        function createClose(){
            $('#createStudentForm').slideUp("fast");
            $('#createButton').animate({width: '100%'});
        }
    </script>
</html>
```

```
    $('#newText').show();
};

function toggleCardID(){
    $("#inputCardID").slideToggle("fast");
};

function updateOpen(){
    createClose();
    deleteClose();
    $('#updateText').hide();
    $('#updateButton').animate({width: '0%'});
    $("#updateStudentForm").slideDown("slow");
};

function updateClose(){
    $("#updateStudentForm").slideUp("fast");
    $("#updateButton").animate({width: '100%'});
    $('#updateText').show();
};

function deleteOpen(){
    createClose();
    updateClose();
    $('#deleteText').hide();
    $('#deleteButton').animate({width: '0%'});
    $("#deleteStudentForm").slideDown("slow");
};

function deleteClose(){
    $("#deleteStudentForm").slideUp("fast");
    $("#deleteButton").animate({width: '100%'});
    $('#deleteText').show();
};

</script>

<style>
    body {
        overflow-y: auto !important;
        padding: 1rem;
    }
    .input {
        position: relative;
        display: block;
    }
```

```
background: #ddd;

&::after {
  content: "▼";
  position: absolute;
  right: 1.5rem;
  top: 1.6rem;
  transition: 300ms transform 200ms;
}
&:active::after {
  transform: rotate(-180deg);
}
&__field {
  display: block;
  opacity: 1;
  width: 100%;
  box-sizing: border-box;
  border: 0;
  background: transparent;
  appearance: none;
  padding: 2rem 1.5rem 1rem;
  border-bottom: 2px solid purple;
  border-radius: 0;
  transition-delay: 0ms;
  will-change: color;
  transition: 200ms color linear;

  &:invalid {
    color: transparent;
    transition: 200ms color linear 100ms;
  }

  &:valid + .input__label,
  &:focus:valid + .input__label {
    transform: scale(0.75) translate(0.5em, -10%);
    transition: 200ms transform ease-out;
    color: purple;
  }
}
&__label {
  position: absolute;
  left: 0;
  top: 0;
  right: 0;
  padding: 1.5rem 1.5rem 0;
  pointer-events: none;
  transform-origin: 0 0;
  transition: 200ms transform ease-out 200ms;
```

```
    will-change: transform;
  }
}
.form-control:focus, .dataTable-input:focus {
  color: #212529;
  background-color: #fff;
  border-color: #522a5b;
  outline: 0;
  box-shadow: 0 0 0 0.2rem rgba(82, 42, 91, 0.65);
}

.form-check-input:checked {
  background-color: #522a5b !important;
  border-color: #522a5b !important;
}

.form-check-input:focus {
  border-color: #522a5b;
  outline: 0;
  box-shadow: 0 0 0 0.15rem rgba(82, 42, 91, 0.65);
}

.body{
  overflow-y: visible !important;
}

table.dataTable thead .sorting_asc {
  background-image: url("/assets/img/UpArrow.png")
}

table.dataTable thead .sorting_desc {
  background-image: url("/assets/img/DownArrow.png")
}

.button-group{
  display: inline-flex;
}

.button{
  padding: 10px;
  outline: none;
  border: none;
  cursor: pointer;
  border-right: 1px solid #cccccc;
  background: #522a5b;
  color: #ffffff;
  width: 100%;
}
```

```

        .button:last-child{
            border-right: none;
        }
    </style>

    <body>
        <div id="layoutSidenav_content">
            <main>
                <div class="container-fluid px-4">
                    <h1 class="mt-4">Student Accounts</h1>
                    <ol class="breadcrumb mb-4">
                        <li class="breadcrumb-item active">The Highcliffe Sixth Form Student Account Home</li>
                    </ol>
                </div>

                <div class="row-fluid">
                    <div class="container-fluid px-4 mt-3 button-group">
                        <button input="submit" id="createButton" onclick="createOpen()" class="button"><span id="newText">New Student
Account</span></button>
                        <button input="submit" id="deleteButton" onclick="deleteOpen()" class="button"><span id="deleteText">Delete Student
Account</span></button>
                    </div>
                </div>

                <div class="container" id="createStudentForm" style="display: none;">
                    <div class="row">
                        <div class="col-lg-12" style="float:centre;" style="border-radius: 2em !important;">

                            <form method="post" action="createStudentOOP.php">

                                <div class="card shadow-lg border-0 rounded-sm mt-3">
                                    <div class="card-header highcliffe">
                                        <a input="" class="btn btn-dark" style="float:left;" onclick="createClose()">Back</a>
                                        <h3 class="text-center font-weight-light my-4" style="color:#ffffff !important;">Create Student Account</h3>
                                    </div>

                                    <div class="card-body">
                                        <div class="login">
                                            <div class="row">
                                                <div class="col-lg-6">
                                                    <div class="form-floating mb-3">
                                                        <input class="form-control" name="fname" id="inputfname" type="text" placeholder="First Name"
autocomplete="off" required/>

                                                        <label for="inputfname">First Name</label>
                                                    </div>
                                                    <div class="form-floating mb-3">

```



```

autocomplete="off" required>

        <select class="form-control" id="tgroup" name="tgroup" placeholder="Choose Tutor Group"

            <option hidden></option>
            <option value="6.1">6.1</option>
            <option value="6.2">6.2</option>
            <option value="6.3">6.3</option>
            <option value="6.4">6.4</option>
            <option value="6.5">6.5</option>
            <option value="6.6">6.6</option>
            <option value="6.7">6.7</option>
            <option value="6.8">6.8</option>
            <option value="6.9">6.9</option>
            <option value="6.10">6.10</option>
            <option value="6.11">6.11</option>
            <option value="6.12">6.12</option>
        </select>
        <label for="tgroup">Tutor Group</span>
    </div>
    <div class="form-floating mb-3">
        <input class="form-control" name="canNum" id="inputCandidateNum" type="number" pattern="[1-
9]{1}[0-9]{3}" placeholder="Candidate Number" autocomplete="off" required/>
        <label for="inputCandidateNum">Candidate Number <i class="muted" style="font-size: 10px;">(4
Numbers)</i></label>
    </div>
</div>
<div class="col-lg-6">
    <div class="form-floating mb-3">
        <input class="form-control" name="sname" id="inputsname" type="text" placeholder="Surname"
autocomplete="off" required/>
        <label for="inputsname">Surname</label>
    </div>
    <div class="form-floating mb-3">
        <select class="form-control" id="ygroup" name="ygroup" placeholder="Choose Year Group..."
autocomplete="off" required>
            <option hidden></option>
            <option value="12">Year 12</option>
            <option value="13">Year 13</option>
        </select>
        <label for="ygroup">Year Group</label>
    </div>
    <div class="form-floating mb-3">
        <input class="form-control" name="aNum" id="inputAdminNum" type="number" placeholder="Admin
Number" autocomplete="off" required/>
        <label for="inputAdminNum">Admin Number <i class="muted" style="font-size: 10px;">(6
Numbers)</i></label>
    </div>
</div>
</div>

```

```

checked/>
        <div class="form-check mb-3">
            <input class="form-check-input" id="inputRandomCardID" onclick="toggleCardID()" type="checkbox" value=""

        <label class="form-check-label" for="inputRandomCardID">Random CardID</label>
        </div>

        <div class="form-floating mb-3" id="inputCardID" style="display:none">
            <input class="form-control" id="inputCardID" name="cardID" type="text" placeholder="CardID"

autocomplete="off" />
            <label for="inputCardID">CardID <i style="font-size: 10px;">(11 Characters)</i></label>
        </div>
    </div>
</div>
<div class="card-footer text-center">
    <div class="mb-2" style="float: right !important;">
        <button type="submit" id="createsubmitButton" class="btn btn-highcliffe" value="Submit">Create
Account</button>
    </div>
</div>

</div>
</form>
</div>
</div>
</div>

<div class="container" id="deleteStudentForm" style="display: none;">
    <div class="row">
        <div class="col-lg-12" style="float:centre;" style="border-radius: 2em !important;">

            <form method="post" action="deleteStudentOOP.php" onsubmit="return confirm('Are you sure you want to delete this student
account?');">

                <div class="card shadow-lg border-0 rounded-sm mt-3">
                    <div class="card-header highcliffe">
                        <a input="" class="btn btn-dark" style="float:left;" onclick="deleteClose()">Back</a>
                        <h3 class="text-center font-weight-light my-4" style="color:#ffffff !important;">Delete Student Account</h3>
                    </div>

                    <div class="card-body">
                        <div class="login">
                            <div class="row">
                                <div class="col-lg-6">
                                    <div class="form-floating mb-3">
                                        <input class="form-control" name="fname" id="inputfname" type="text" placeholder="First Name"

autocomplete="off" required/>

```

```

        <label for="inputfname">First Name</label>
    </div>
    <div class="form-floating mb-3">
        <input class="form-control" name="canNum" id="inputCandidateNum" type="number" pattern="[1-
9]{1}[0-9]{3}" placeholder="Candidate Number" autocomplete="off" required/>
        <label for="inputCandidateNum">Candidate Number <i class="muted" style="font-size: 10px;">(4
Numbers)</i></label>
    </div>
</div>
<div class="col-lg-6">
    <div class="form-floating mb-3">
        <input class="form-control" name="sname" id="inputsname" type="text" placeholder="Surname"
autocomplete="off" required/>
        <label for="inputsname">Surname</label>
    </div>
    <div class="form-floating mb-3">
        <input class="form-control" name="aNum" id="inputAdminNum" type="number" placeholder="Admin
Number" autocomplete="off" required/>
        <label for="inputAdminNum">Admin Number <i class="muted" style="font-size: 10px;">(6
Numbers)</i></label>
    </div>
</div>
<!-- Lots of fields required to lessen the possibility of accidentally deleting the wrong account -->
</div>
</div>
</div>
<div class="card-footer text-center">
    <div class="mb-2" style="float: right !important;">
        <button type="submit" id="deletesubmitButton" class="btn btn-highcliffe" onclick="return('Are you sure you
wish to delete this account?')" value="Submit">Delete Account</button>
    </div>
</div>
</div>
</div>
</form>
</div>
</div>
</div>
<div class="row mt-5 px-4">
    <div class="col-xl-12">
        <div class="card mb-4">
            <div class="card-header" style="font-size: 20px">
                <span style="color: #522a5b"><i class="fas fa-address-card me-1"></i></span>
                All Student Accounts
            </div>
            <div class="card-body">
                <button class="btn btn-sm btn-highcliffe" onClick="window.location.reload();" style="float:right; margin-top:5px; margin-
right:5px;">Refresh</button>
            </div>
        </div>
    </div>
</div>

```

```

</div>

<div class="card-body">
  <table id="myTable" class="display" style="width:100%">
    <thead>
      <tr>
        <th>Candidate Number</th>
        <th>Surname</th>
        <th>Firstname</th>
        <th>Admin Number</th>
        <th>Tutor Group</th>
        <th>Year Group</th>
        <th>StatusID</th>
        <th>CardID</th>
      </tr>
    </thead>

    <tbody>
      <?php
        include($IPATH."connectDB.php");

        $sql = "SELECT * FROM `students`
                LEFT JOIN `statusids`           #Using a JOIN to link to another table
                ON students.CandidateNumber = statusids.CandidateNumber;";
#LEFT JOIN to collect all accounts and their linked statusID and not the other way round

        $result = mysqli_query($conn,$sql);
        if (!$result) {
          printf("Error: %s\n", mysqli_error($conn));
          exit();
        }else{
          while($row = mysqli_fetch_array($result)){
            echo "<tr>";
            echo "<td>" . $row['CandidateNumber'] . "</td>";
            echo "<td>" . $row['Surname'] . "</td>";
            echo "<td>" . $row['Firstname'] . "</td>";
            echo "<td>" . $row['misID'] . "</td>";
            echo "<td>" . $row['TutorGroup'] . "</td>";
            echo "<td>" . $row['YearGroup'] . "</td>";
            echo "<td>" . $row['StatusID'] . "</td>";
            echo "<td>" . $row['CardID'] . "</td>";
            echo "</tr>";
          }
        }

        $conn->close();
      ?>
    </tbody>
  </table>
</div>

```

```

        </table>
    </div>
</div>
</div>
</div>

</main>

<script>
    $(document).ready(function () {          //USE OF DATATABLES, a Java plugin for sortable tables
        $('#myTable').DataTable({
            "paging": false,
            "order": [[5, 'asc'], [1, 'asc']]
        });
    });
</script>

<?php include($IPATH."footer.php"); ?>
</div>
</body>
</html>

```

### students.php

<!-- This is the page which displays all current students stati -->

```

<!DOCTYPE html>
<html lang="en">
    <title>Student Statuses</title>

    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
    <link href="css/switch.css?version=55" rel="stylesheet" />
    <link href="//cdn.datatables.net/1.11.3/css/jquery.dataTables.min.css" rel="stylesheet" />
    <script src="//cdn.datatables.net/1.11.3/js/jquery.dataTables.min.js"></script>

    <?php $IPATH = $_SERVER["DOCUMENT_ROOT"]. "/PROJECT/Website/assets/IPATH/";
    include($IPATH."headernav.php");

    if(isset($_GET['show'])){
        $show = $_GET['show'];
        if ($show == 1){
            $status = "IN";
            $show = 1;
        } elseif ($show == 0){
            $status = "OUT";

```

```

        $show = 0;
    } else{
        $status = "IN";
        $show = 1;
    }
} else{
    $status = "IN";
    $show = 1;
}

?>

<body>
    <div id="layoutSidenav_content">
        <main>
            <div class="container-fluid px-4">
                <h1 class="mt-4">Student Statuses</h1>
                <ol class="breadcrumb mb-4">
                    <li class="breadcrumb-item active">Welcome to the hub of the Highcliffe Sixth Form statuses</li>
                </ol>

                <div class="row">
                    <div class="col-xl-12">
                        <div class="card mb-4">
                            <div class="card-header" style="font-size: 20px">
                                <span style="color: #522a5b"><i class="fas fa-database me-1"></i></span>
                                Students Signed <?php echo $status?>

                                <form action="students.php?show=0" method="POST" style="float:right; margin-top:2px;">
                                    <input type="submit" class="btn btn-sm btn-primary" style="background-color: #343A40 !important; border-color: #343A40 !important; max-height:60%;" name="update" value="Out">
                                </form>
                                <form action="students.php?show=1" method="POST" style="float:right; margin-top:2px; margin-right:5px;">
                                    <input type="submit" class="btn btn-sm btn-primary" style="background-color: #343A40 !important; border-color: #343A40 !important; max-height:60%;" name="update" value="In">
                                </form>
                                <button class="btn btn-sm btn-highcliffe" onClick="window.location.reload();" style="float:right; margin-top:5px; margin-right:5px;">Refresh</button>
                            </div>

                            <div class="card-body">
                                <table id="myTable" class="display" style="width:100%">
                                    <thead>
                                        <tr>
                                            <th>Candidate Number</th>

```

```

        <th>Surname</th>
        <th>Firstname</th>
        <th>Tutor Group</th>
        <th>Year Group</th>
        <th>Time Changed</th>
        <th>Status</th>
    </tr>
</thead>

<tbody>
    <?php

        $IPATH = $_SERVER["DOCUMENT_ROOT"]. "/PROJECT/Website/assets/IPATH/";
        include($IPATH."connectDB.php");

        $sql = "SELECT status.DateChange, students.CandidateNumber, students.Firstname, students.Surname,
students.TutorGroup, students.YearGroup FROM `status`
LEFT JOIN `statusids` ON status.StatusID = statusids.StatusID
LEFT JOIN `students` ON statusids.CandidateNumber = students.CandidateNumber
WHERE status.status = $show;";
#USE OF SQL JOINS

        $result = mysqli_query($conn,$sql);
        if (!$result) {
            printf("Error: %s\n", mysqli_error($conn));
            exit();
        }else{
            while($row = mysqli_fetch_array($result)){
                echo "<tr>";
                echo "<td>" . $row['CandidateNumber'] . "</td>";
                echo "<td>" . $row['Surname'] . "</td>";
                echo "<td>" . $row['Firstname'] . "</td>";
                echo "<td>" . $row['TutorGroup'] . "</td>";
                echo "<td>" . $row['YearGroup'] . "</td>";
                echo "<td>" . $row['DateChange'] . "</td>";
                echo "<td>" . $status . "</td>";
                echo "</tr>";
            }
        }

        $conn->close();
    ?>
</tbody>
</table>
</div>
</div>
</div>
</div>
</div>
</div>

```

```
</main>

<script>
    $(document).ready(function () {
        $('#myTable').DataTable({
            "paging": false
        });
    });
</script>

<?php include($IPATH."footer.php"); ?>
</div>
</body>
</html>
```

### updateStatus.php

<!-- This is the page visited to update a student's status -->

```
<!DOCTYPE html>
<html lang="en">
    <title>Update Student Status</title>
    <?php $IPATH = $_SERVER["DOCUMENT_ROOT"]. "/PROJECT/Website/assets/IPATH/"; include($IPATH."headernav.php"); ?>

    <style>
        .form-control:focus, .dataTable-input:focus {
            color: #212529;
            background-color: #fff;
            border-color: #522a5b;
            outline: 0;
            box-shadow: 0 0 0 0.2rem rgba(82, 42, 91, 0.65);
        }

        .form-check-input:checked {
            background-color: #522a5b !important;
            border-color: #522a5b !important;
        }

        .form-check-input:focus {
            border-color: #522a5b;
            outline: 0;
            box-shadow: 0 0 0 0.15rem rgba(82, 42, 91, 0.65);
        }
    </style>
```



```

<body>
  <div id="layoutSidenav_content">
    <main>
      <div class="container-fluid px-4">
        <h1 class="mt-4">Update Student Status</h1>
        <ol class="breadcrumb mb-4">
          <li class="breadcrumb-item active">Update A Highcliffe Sixth Form Student's Status</li>
        </ol>

        <div class="container" id="deleteStudentForm">
          <div class="row">
            <div class="col-lg-12" style="float:centre;" style="border-radius: 2em !important;">

              <form method="post" action="websiginout.php">

                <div class="card shadow-lg border-0 rounded-sm mt-3">
                  <div class="card-header highcliffe">
                    <h3 class="text-center font-weight-light my-4" style="color:#ffffff !important;">Update A Student Status</h3>
                  </div>

                  <div class="card-body">
                    <div class="login">
                      <div class="row">
                        <div class="col-lg-6">
                          <div class="form-floating mb-3">
                            <input class="form-control" name="canNum" id="inputCandidateNum" type="number" pattern="[1-9]{1}[0-9]{3}" placeholder="Candidate Number" autocomplete="off" required/>
                            <label for="inputCandidateNum">Candidate Number <i class="muted" style="font-size: 10px;">(4 Numbers)</i></label>
                          </div>
                        </div>
                        <div class="col-lg-6">
                          <div class="form-floating mb-3">
                            <select class="form-control" id="status" name="status" placeholder="Choose Status"
                                autocomplete="off" required>
                                <option hidden></option>
                                <option value="1">IN</option>
                                <option value="0">OUT</option>
                            </select>
                            <label for="status">Status</span>
                          </div>
                        </div>
                      </div>
                    </div>
                  </div>
                  <div class="card-footer text-center">
                    <div class="mb-2" style="float: right !important;">
                      <button type="submit" id="submitButton" class="btn btn-highcliffe" value="Submit">Update</button>
                    </div>
                  </div>
                </div>
              </form>
            </div>
          </div>
        </div>
      </div>
    </main>
  </div>
</body>

```

```

        </div>
    </div>

    </div>
</form>
</div>
</div>
</div>

</div>
</main>

<?php include($IPATH."footer.php"); ?>

</div>
</body>
</html>

```

### [websigninout.php](#)

<!-- This is the page visited when the updateStatus form is submitted -->

```

<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="utf-8" />
        <meta http-equiv="X-UA-Compatible" content="IE=edge" />
        <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no" />
        <meta name="description" content="" />
        <meta name="author" content="" />
        <title>404 Error - SB Admin</title>
        <link href="css/styles.css" rel="stylesheet" />
        <script src="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.3/js/all.min.js" crossorigin="anonymous"></script>
    </head>

    <?php
function redirect($url) {
    ob_start();
    header('Location: '.$url);
    ob_end_flush();
    die();
}
?>

    <body>
        <div id="layoutError">

```

```

<div id="layoutError_content">
  <main>
    <div class="container">
      <div class="row justify-content-center">
        <div class="col-lg-6">
          <div class="text-center mt-4">
            <h1 class="display-1">500</h1>
            <p class="lead">Internal Server Error</p>

            <?php
              $IPATH = $_SERVER["DOCUMENT_ROOT"]. "/PROJECT/Website/assets/IPATH/";
              include($IPATH."connectDB.php");

              $dbCandidateNumber = $_POST["canNum"];
              $status = $_POST["status"];

              $sql = "SELECT * FROM `statusids` WHERE CandidateNumber=$dbCandidateNumber;";
              $result = $conn->query($sql);
              $row = mysqli_fetch_array($result);
              if ($row){
                $dbStatusID = $row['StatusID'];

                $conn->autocommit(FALSE);  #USE OF SQL TRANSACTIONS

                $sql = "REPLACE status (StatusID, Status)
VALUES ('$dbStatusID', '$status')";
                $conn->query($sql);

                $sql2 = "INSERT INTO `signinhistory` (`CandidateNumber`, `Status`, `Type`)
VALUES ($dbCandidateNumber, $status, 'ONLINE');";
                $conn->query($sql2);

                if ($conn->commit() == TRUE) {
                  echo "New record created successfully";
                  $conn->close();
                  redirect("updateStatus.php");
                } else {
                  echo "Error: " . $sql . "<br>" . $conn->error;
                }
              } else{
                echo "Error: The student account, Candidate Number $dbCandidateNumber, does not exist <br>";
              }

            ?>

            <a href="dashboard.php">
              <i class="fas fa-arrow-left me-1"></i>

```

```
<?php
    $servername = "localhost";
    $username = "username";
    $password = "password";
    $dbname = "database";

    $conn = new mysqli($servername, $username, $password, $dbname);
    if($conn->connect_error){
        die("Connection failed: " . $conn->connect_error);
    }
    echo"Connected successfully" . "<br>";
```

```

$sql = "UPDATE `status`
SET
`status` = 0";

if ($conn->query($sql)==True){
    echo "New record created successfully";
} else{
    echo "Error: " . $sql . "<br>" . $conn->error;
}

?>

```

### signinout.php

```

<?php
    $IPATH = $_SERVER["DOCUMENT_ROOT"]. "/PROJECT/Website/assets/IPATH/"; include($IPATH."connectDB.php");    #connects to database

    $dbCandidateNumber = $_GET["CandidateNumber"];    #assigns variables from URL
    $status = $_GET["Status"];

    if ($status == "True"){
        $dbStatus = 1;
    } elseif ($status == "False"){
        $dbStatus = 0;
    } else {
        die("Error: Invalid Status");
    }

    $sql = "SELECT * FROM `statusids` WHERE CandidateNumber=$dbCandidateNumber;";    #finds account with candidate number from URL
    $result = $conn->query($sql);
    $row = mysqli_fetch_array($result);
    if ($row){    #if account exists
        $dbStatusID = $row['StatusID'];

        $sql = "REPLACE status (StatusID, Status)
VALUES ('$dbStatusID', '$dbStatus')";

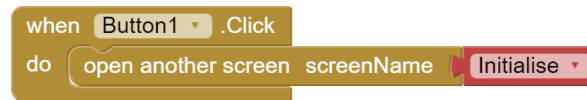
        if ($conn->query($sql)==True){    #updates current status
            echo "New currentstatus record created successfully" . "<br>";
        } else{
            echo "Error: " . $sql . "<br>" . $conn->error . "<br>";
        }

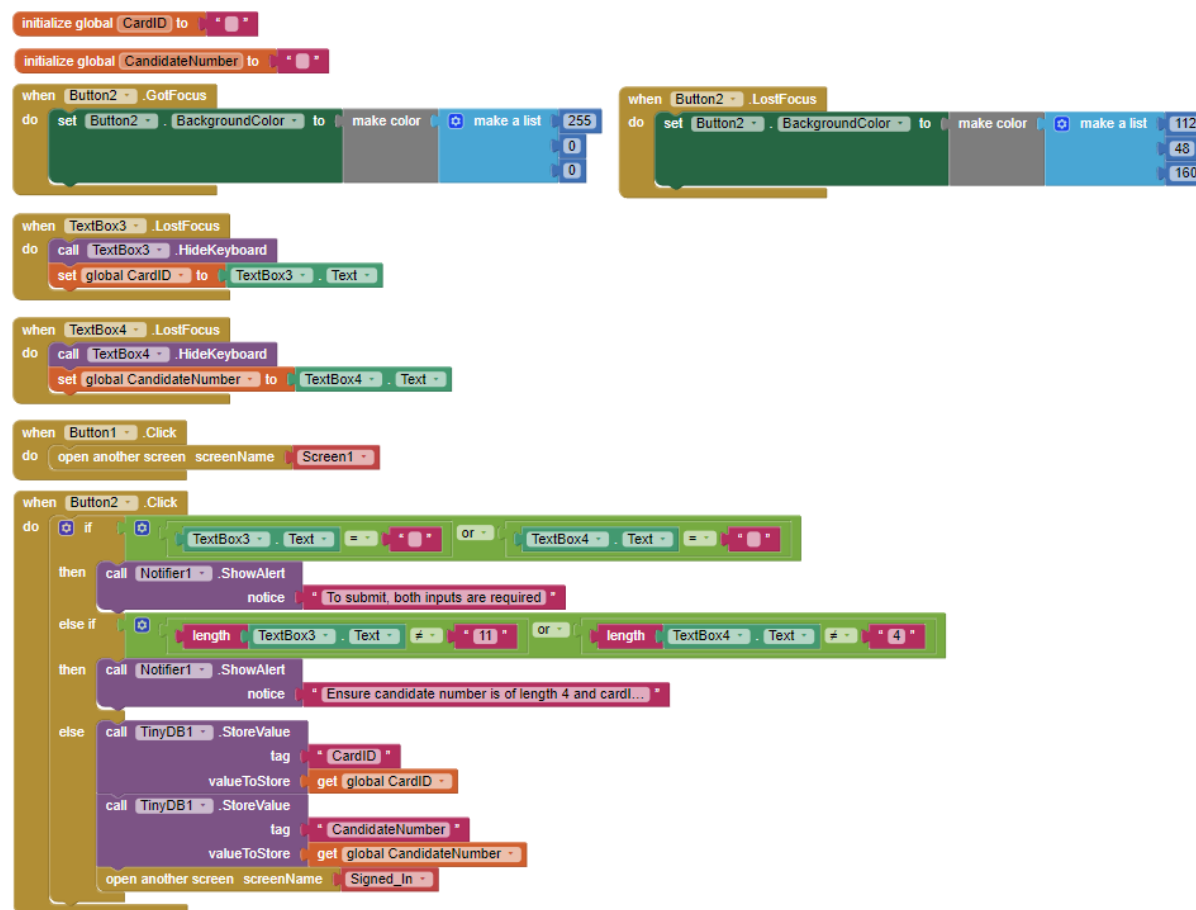
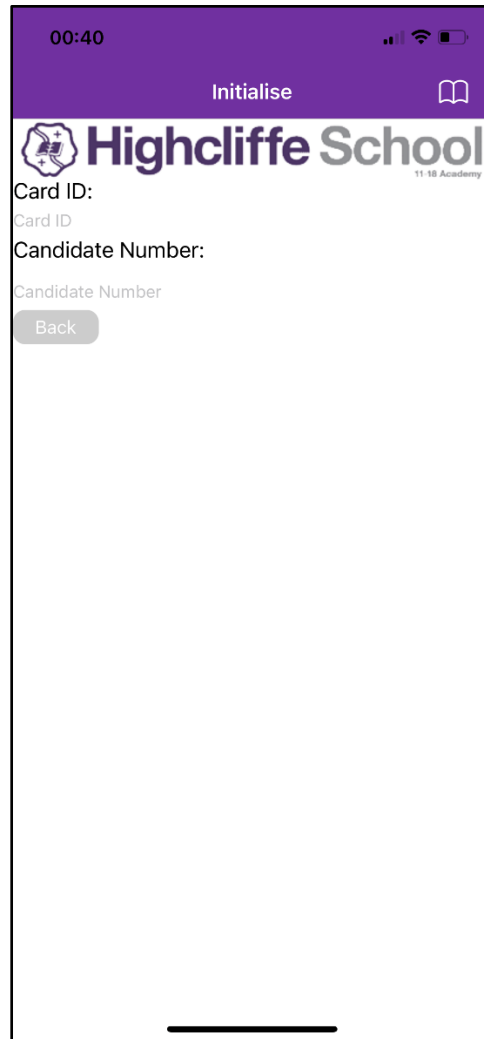
        $sql = "INSERT INTO `signinhistory` (`CandidateNumber`, `Status`, `Type`)
VALUES ($dbCandidateNumber, $dbStatus, 'ONLINE');";
    }

```

```
if ($conn->query($sql)==True){                                #adds to signhistory
    echo "New signhistory record created successfully";
} else{
    echo "Error: " . $sql . "<br>" . $conn->error;
}
} else{
    echo "Error: This student account does not exist";
}

?>
```

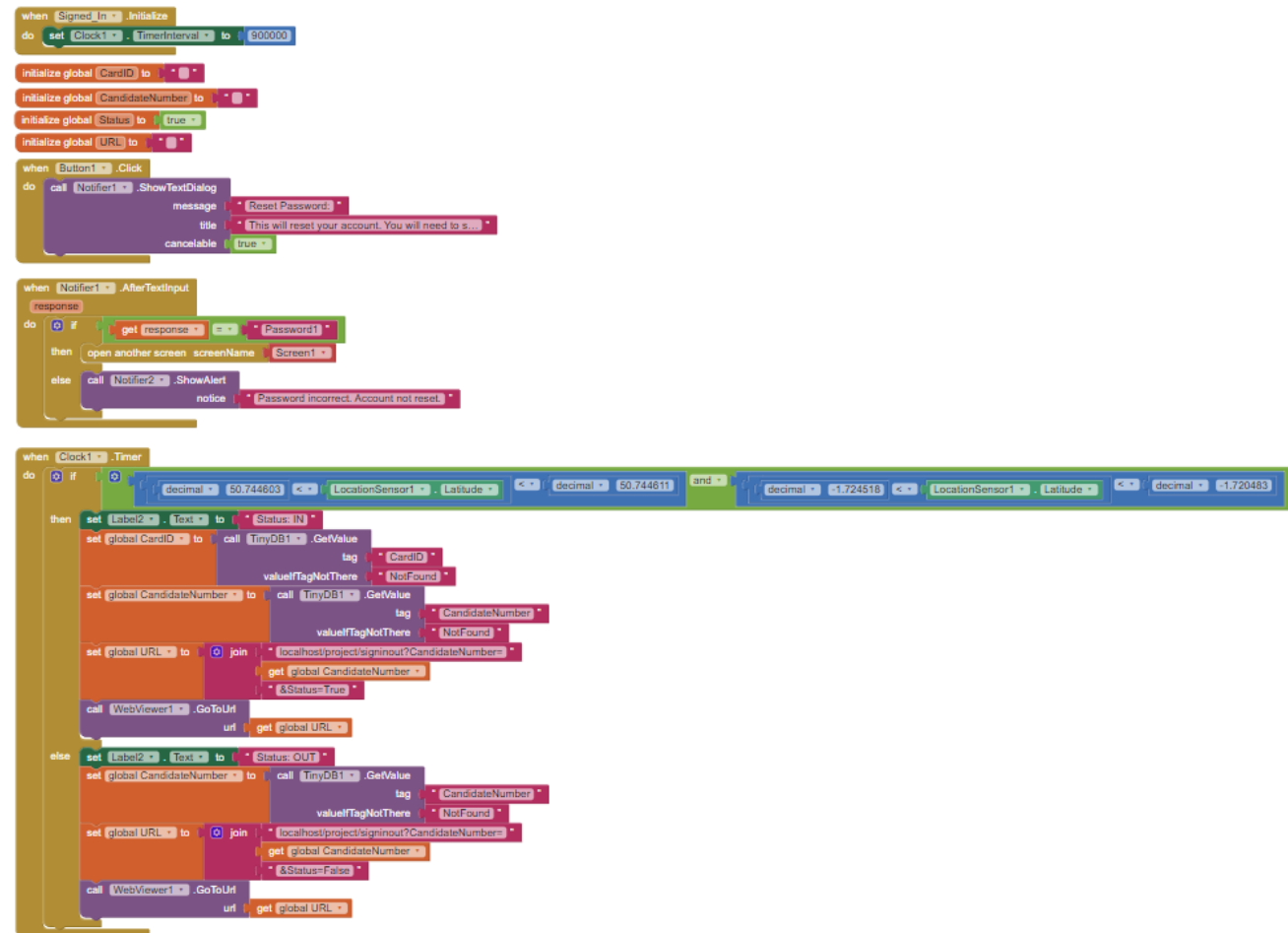
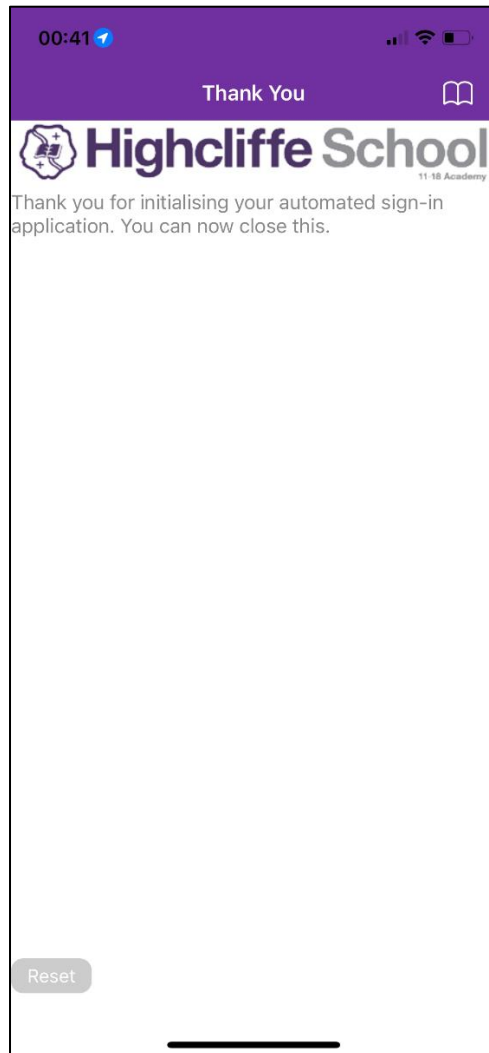
[open screen](#)

initialise

The inputs which are required are stored in a TinyDB document database. This means that the students do not have to initialise their app each time they open it.



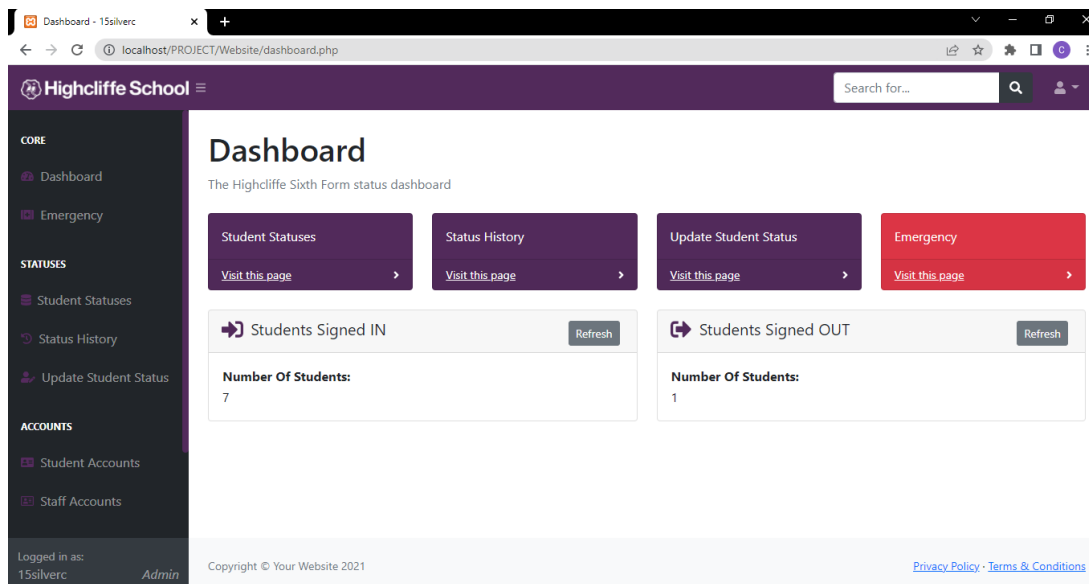
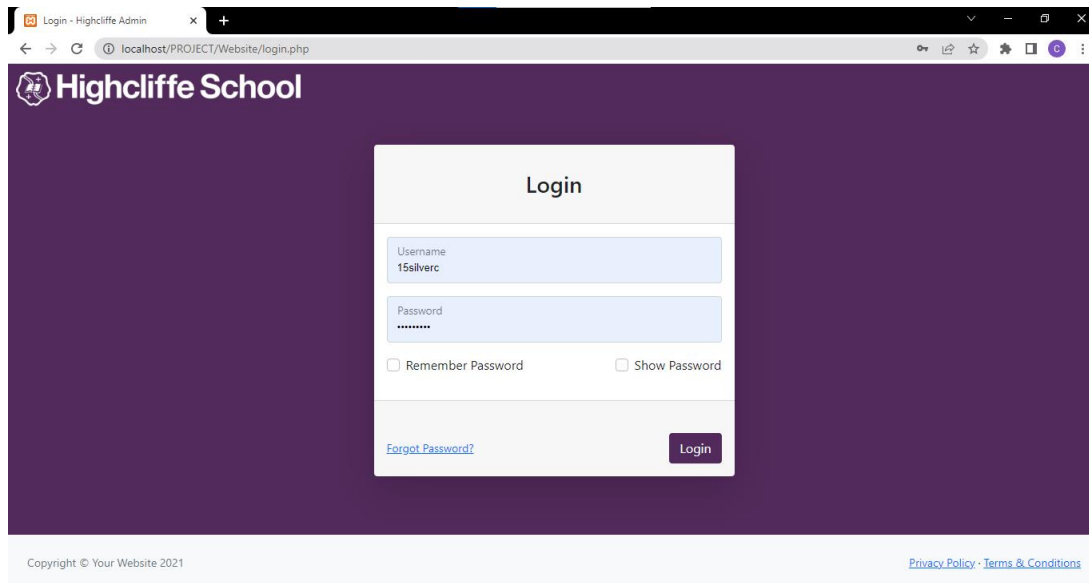
[signed in](#)

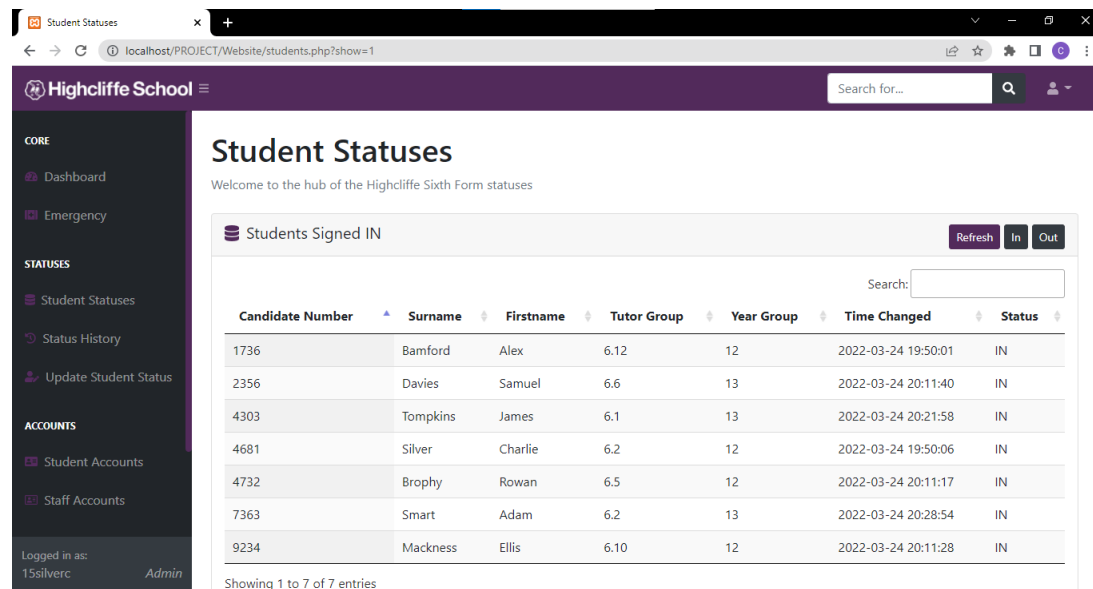


The time interval is set to 900000ms which is 15 minutes. Every 15 minutes clock1 causes the if statement to be checked. If the device is within the school grounds when that statement is checked, it visits the web address which updates this student's status.

## 6.3) Interface Screenshots

The following pictures are screenshots from the admin portal.





Student Statuses

Welcome to the hub of the Highcliffe Sixth Form statuses

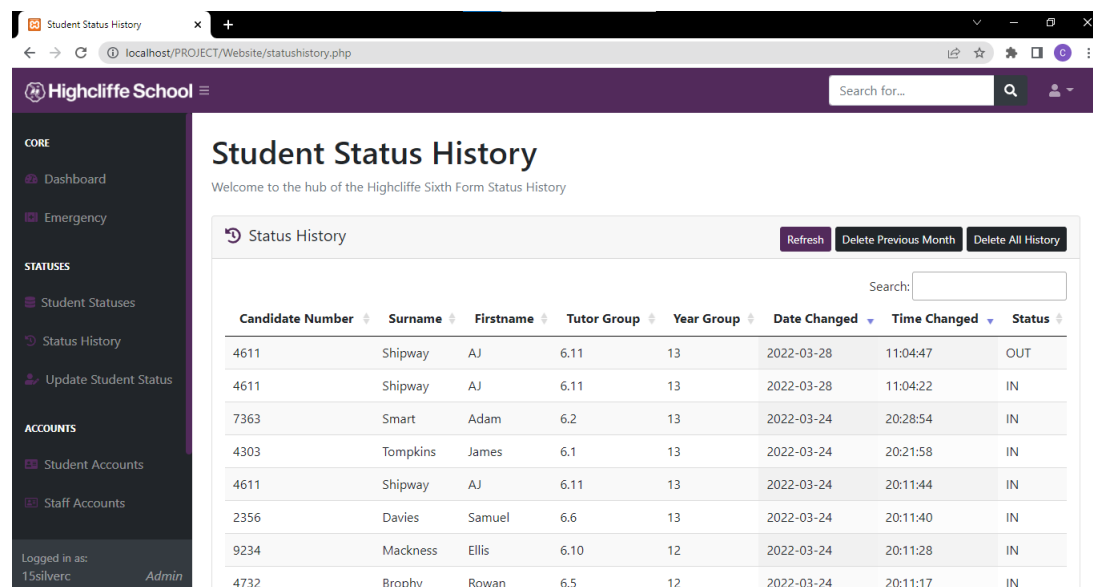
Students Signed IN

Refresh In Out

Search:

| Candidate Number | Surname  | Firstname | Tutor Group | Year Group | Time Changed        | Status |
|------------------|----------|-----------|-------------|------------|---------------------|--------|
| 1736             | Bamford  | Alex      | 6.12        | 12         | 2022-03-24 19:50:01 | IN     |
| 2356             | Davies   | Samuel    | 6.6         | 13         | 2022-03-24 20:11:40 | IN     |
| 4303             | Tompkins | James     | 6.1         | 13         | 2022-03-24 20:21:58 | IN     |
| 4681             | Silver   | Charlie   | 6.2         | 12         | 2022-03-24 19:50:06 | IN     |
| 4732             | Brophy   | Rowan     | 6.5         | 12         | 2022-03-24 20:11:17 | IN     |
| 7363             | Smart    | Adam      | 6.2         | 13         | 2022-03-24 20:28:54 | IN     |
| 9234             | Mackness | Ellis     | 6.10        | 12         | 2022-03-24 20:11:28 | IN     |

Showing 1 to 7 of 7 entries



Student Status History

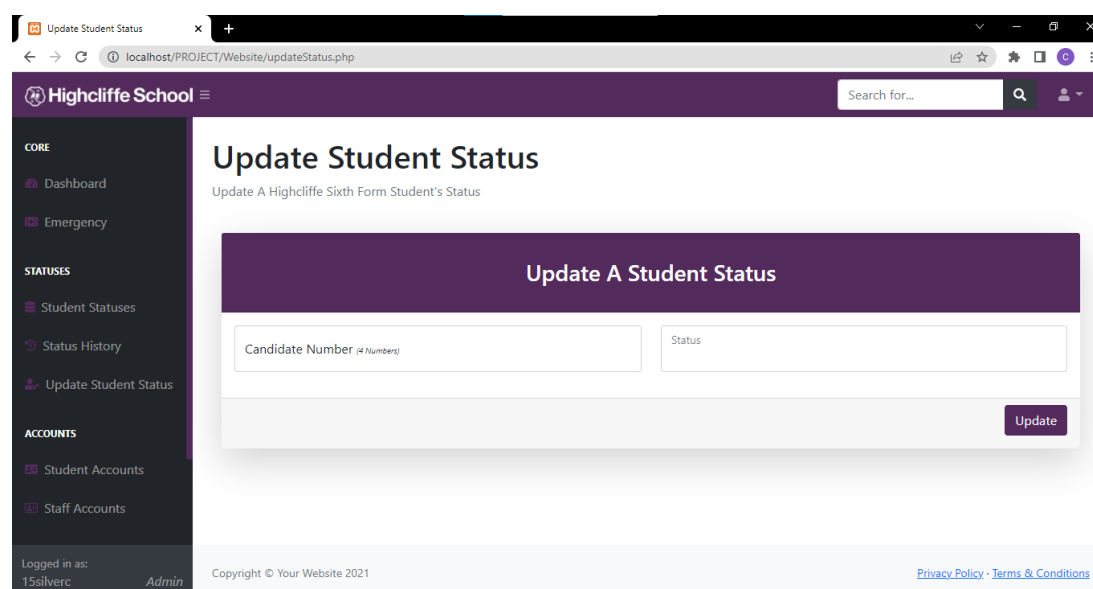
Welcome to the hub of the Highcliffe Sixth Form Status History

Status History

Refresh Delete Previous Month Delete All History

Search:

| Candidate Number | Surname  | Firstname | Tutor Group | Year Group | Date Changed | Time Changed | Status |
|------------------|----------|-----------|-------------|------------|--------------|--------------|--------|
| 4611             | Shipway  | AJ        | 6.11        | 13         | 2022-03-28   | 11:04:47     | OUT    |
| 4611             | Shipway  | AJ        | 6.11        | 13         | 2022-03-28   | 11:04:22     | IN     |
| 7363             | Smart    | Adam      | 6.2         | 13         | 2022-03-24   | 20:28:54     | IN     |
| 4303             | Tompkins | James     | 6.1         | 13         | 2022-03-24   | 20:21:58     | IN     |
| 4611             | Shipway  | AJ        | 6.11        | 13         | 2022-03-24   | 20:11:44     | IN     |
| 2356             | Davies   | Samuel    | 6.6         | 13         | 2022-03-24   | 20:11:40     | IN     |
| 9234             | Mackness | Ellis     | 6.10        | 12         | 2022-03-24   | 20:11:28     | IN     |
| 4732             | Brophy   | Rowan     | 6.5         | 12         | 2022-03-24   | 20:11:17     | IN     |



Update Student Status

Update A Highcliffe Sixth Form Student's Status

Update A Student Status

Candidate Number (4 Number) Status

Update

Copyright © Your Website 2021

[Privacy Policy](#) · [Terms & Conditions](#)

The first screenshot shows the 'Student Accounts' page. It features a sidebar with navigation links for CORE (Dashboard, Emergency), STATUSES (Student Statuses, Status History, Update Student Status), and ACCOUNTS (Student Accounts, Staff Accounts). The main content area has buttons for 'New Student Account' and 'Delete Student Account', followed by a table titled 'All Student Accounts'. The table lists student details including Candidate Number, Surname, Firstname, Admin Number, Tutor Group, Year Group, StatusID, and CardID. The second screenshot shows the 'Create Student Account' form, which includes input fields for First Name, Surname, Tutor Group, Year Group, Candidate Number, and Admin Number, along with a checkbox for 'Random CardID'. The third screenshot shows the 'Staff Accounts' page, which has a similar sidebar and main content area with buttons for 'New Staff Account' and 'Delete Staff Account', followed by a table titled 'All Staff Accounts' listing staff details like Username and Staff Type.

**Student Accounts**

The Highcliffe Sixth Form Student Account Home

New Student Account Delete Student Account

All Student Accounts Refresh

| Candidate Number | Surname  | Firstname | Admin Number | Tutor Group | Year Group | StatusID | CardID      |
|------------------|----------|-----------|--------------|-------------|------------|----------|-------------|
| 1736             | Bamford  | Alex      | 723085       | 6.12        | 12         | 35402406 | hsk392hv8v4 |
| 4732             | Brophy   | Rowan     | 356846       | 6.5         | 12         | 23434346 | h2v0MMiSnuy |
| 9234             | Mackness | Ellis     | 913502       | 6.10        | 12         | 49374797 | UyP2eoBBfOW |
| 4681             | Silver   | Charlie   | 242135       | 6.2         | 12         | 39692020 | cuvK7gGOTWU |
| 2356             | Davies   | Samuel    | 326789       | 6.6         | 13         | 23475684 | SyUtsNwQVfp |
| 4611             | Shipway  | AJ        | 124253       | 6.11        | 13         | 88888888 | 2usdxtXaXv  |

**Create Student Account**

Back

First Name Surname

Tutor Group Year Group

Candidate Number (4 Numbers) Admin Number (6 Numbers)

☒ Random CardID

Create Account

**Staff Accounts**

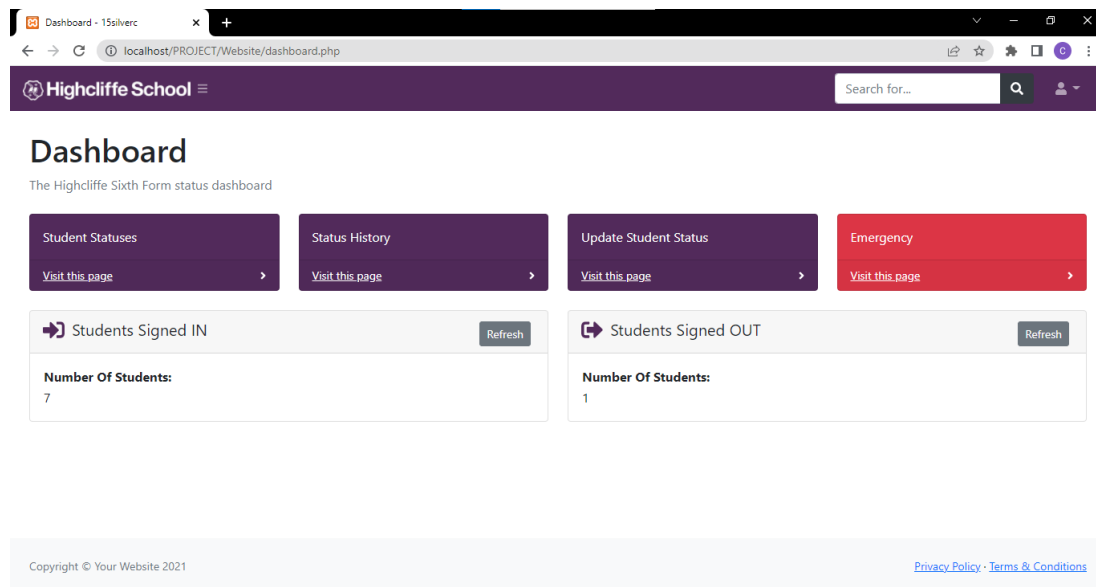
The Highcliffe Sixth Form Staff Account Home

New Staff Account Delete Staff Account

All Staff Accounts Refresh

| Username  | Staff Type |
|-----------|------------|
| 15silverc | A          |
| test      | S          |

Copyright © Your Website 2021 [Privacy Policy](#) [Terms & Conditions](#)



## 6.4) Project Resources

Public-domain resources that were used in the project:

- Font Awesome 5.15.3; icons and images – [www.fontawesome.com](http://www.fontawesome.com)
- Bootstrap 5.1.3; dynamic visual framework – [www.getbootstrap.com](http://www.getbootstrap.com)
- DataTables 1.11.5; interactable HTML tables – [www.datatables.net](http://www.datatables.net)
- MySQLi; PHP extension for SQL queries - [www.php.net/manual/en/book.mysqli.php](http://www.php.net/manual/en/book.mysqli.php)
- JQuery 3.5.1; Javascript library for animation and event-handling – [www.jquery.com](http://www.jquery.com)
- TinyDB; document-oriented database in python – [www.tinydb.readthedocs.io/en/latest/](http://www.tinydb.readthedocs.io/en/latest/)
- MIT AppInventor 2; Java app-creation tool – [www.appinventor.mit.edu/](http://www.appinventor.mit.edu/)

## 6.5) Database DDL/Schema

My database was accessed via MySQL, with a PHP server running from Apache. Both of these were run through XAMPP and phpMyAdmin.

### admins

```
CREATE TABLE `admins` (
  `aID` int(11) NOT NULL,
  `username` text NOT NULL,
  `hashedpassword` text NOT NULL,
  `access` text NOT NULL
)
ALTER TABLE `admins`
  ADD PRIMARY KEY (`aID`),
  ADD UNIQUE KEY `username` (`username`) USING HASH;
```

[signinhistory](#)

```
CREATE TABLE `signinhistory` (  
  `sihID` int(11) NOT NULL,  
  `CandidateNumber` int(4) NOT NULL,  
  `DateChange` date NOT NULL DEFAULT current_timestamp(),  
  `TimeChange` time NOT NULL DEFAULT current_timestamp(),  
  `Status` binary(1) NOT NULL,  
  `Type` text NOT NULL  
)  
ALTER TABLE `signinhistory`  
  ADD PRIMARY KEY (`sihID`);
```

[status](#)

```
CREATE TABLE `status` (  
  `StatusID` int(11) NOT NULL,  
  `status` binary(1) NOT NULL,  
  `DateChange` timestamp NOT NULL DEFAULT current_timestamp() ON UPDATE current_timestamp()  
)  
ALTER TABLE `status`  
  ADD PRIMARY KEY (`StatusID`);
```

[statusids](#)

```
CREATE TABLE `statusids` (  
  `sisID` int(11) NOT NULL,  
  `CandidateNumber` int(4) NOT NULL,  
  `StatusID` int(11) NOT NULL  
)  
ALTER TABLE `statusids`  
  ADD PRIMARY KEY (`sisID`),  
  ADD UNIQUE KEY `CandidateNumber` (`CandidateNumber`),  
  ADD UNIQUE KEY `StatusID` (`StatusID`);
```

[students](#)

```
CREATE TABLE `students` (  
  `sID` int(11) NOT NULL,  
  `CandidateNumber` varchar(4) NOT NULL,  
  `misID` varchar(6) NOT NULL,  
  `Surname` text NOT NULL,  
  `Firstname` text NOT NULL,  
  `TutorGroup` text NOT NULL,  
  `CardID` text NOT NULL,  
  `YearGroup` int(2) NOT NULL  
)  
ALTER TABLE `students`  
  ADD PRIMARY KEY (`sID`),  
  ADD UNIQUE KEY `CardID` (`CardID`) USING HASH;
```