# 1.0 Database Data Structure

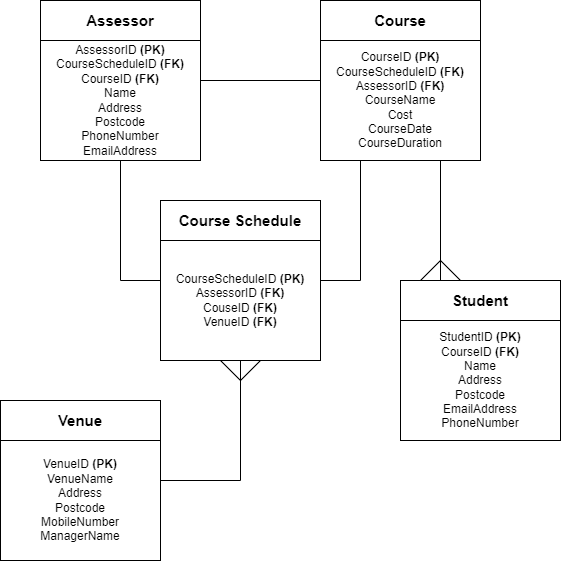
## Data Dictionary:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Data Dictionary** | | | | | | |
| **Table** | **Field** | **Type** | **Size/Format** | **Validation Rule** | **Input Mask** | **Notes** |
| **Assessor** | AssessorID | Short Text | 6 |  | 000000 | PK |
| CourseScheduleID | Short Text | 7 |  | LL0000L | FK |
| CourseID | Short Text | 6 |  | LL0000 | FK |
| Name | Long Text | <100 |  |  |  |
| Address | Long Text |  | [Chars]@[Chars].[Chars] |  |  |
| Postcode | Short Text | 5-7 |  |  |  |
| PhoneNumber | Short Text | 11 | Is Numeric | 00000000000 |  |
| EmailAddress | Long Text |  |  |  |  |
|  | | | | | | |
| **Course** | CourseID | Short Text | 6 |  | LL0000 | PK |
| CourseScheduleID | Short Text | 7 |  | LL0000L | FK |
| AssessorID | Short Text | 6 |  | 000000 | FK |
| CourseName | Long Text |  |  |  |  |
| Cost | Currency | Currency | Is In Pounds |  |  |
| CourseDate | Date/Time | Short Date |  | DD/MM/YYYY |  |
| CourseDuration | Number | Integer |  |  | Yearly |
|  | | | | | | |
| **Venue** | VenueID | Short Text | 6 |  | LLL000 | PK |
| VenueName | Long Text |  |  |  |  |
| Address | Long Text |  | [Chars]@[Chars].[Chars] |  |  |
| Postcode | Short Text | 5-7 |  |  |  |
| MobileNumber | Short Text | 11 | Is Numeric | 00000000000 |  |
| ManagerName | Long Text |  |  |  |  |
|  | | | | | | |
| **Course Schedule** | CourseScheduleID | Short Text | 7 |  | LL0000L | PK |
| AssessorID | Short Text | 6 |  | 000000 | FK |
| CourseID | Short Text | 6 |  | LL0000 | FK |
| VenueID | Short Text | 6 |  | LLL000 | FK |
|  | | | | | | |
| **Student** | StudentID | Short Text | 6 |  | LLLL00 | PK |
| CourseID | Short Text | 6 |  | LL0000 | FK |
| Name | Long Text |  |  |  |  |
| Address | Long Text |  |  |  |  |
| Postcode | Short Text | 5-7 |  |  |  |
| EmailAddress | Long Text |  | [Chars]@[Chars].[Chars] |  |  |
| PhoneNumber | Short Text | 11 | Is Numeric | 00000000000 |  |
| Achieved (Y/N) | Boolean | Boolean | Is not null |  |  |

## Data Flow Diagrams:

|  |
| --- |
| **Data Flow Diagrams** |
| **Assessor** |
| **Adding a new student** |
|  |
| **Updating a student’s record** |
|  |
| **Adding Student Achievement** |
|  |
| **Manager** |
| **Automatic generation of the weekly course booking report** |
|  |
| **Creating a trainer performance report** |
|  |
| **Creating a course performance report** |
|  |
| **Creating a course popularity report** |
|  |
| **Sending reports to assessors** |
|  |

## Entity Relationship Diagram:



## Normalisation:

### Unnormalized Data:

For the unnormalized table design and the 1st normal form table design, the layout is the same, 1st normalization will affect the data in the table itself, but not the table design. Each form of normalization (0NF and 1NF) has the same attributes.

|  |
| --- |
| **Unnormalized Data and 1st Normal Form Table Design** |
| * StudentName * StudentPhoneNUmber * StudentAddress * StudentPostcode * StudentEmailAddress * Achieved * CourseName * CourseCost * CourseDuration * CourseDate * AssessorName * AssessorAddress * AssessorPhoneNumber * AssessorPostcode * AssessorEmailAddress * VenueName * VenuePostcode * VenueAddress * VenueMobileNumber * ManagerName |

### Unnormalized Data (0NF):

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Database** | | | | | | | | | | | | | | | | | | |
| **Student Name** | **Student**  **Phone**  **Number** | **Student**  **Address** | **Student Postcode** | **Student**  **Email** | **Achieved** | **Course**  **Name** | **Cost** | **Course**  **Date** | **AssessorName** | **AssessorAddress** | **Assessor Phone** | **Assessor**  **Postcode** | **Assessor**  **Email**  **Address** | **Venue Name** | **Venue**  **Postcode** | **Venue Address** | **Venue**  **Mobile Number** | **Manager Name** |
| Cailyn Dunlap,  Mayra Cline | 07388886854,  07193853874 | 48 High Street,  62 York Road | KY112TE,  EC1A4EU | cailynd@outlook.com,  mayracline292@hotmail.com | No,  No | Business | £120.00 | 29/04/2022 | Ismail Watson | 65 Balsham Road | 07548338743 | N9 9LB | iwatson24@gmail.co.uk | Clover Hall | RG14 5NR | 116 Craven Road | 07238483285 | Jamarion Harding |
| Avery Mendoza | 07219327893 | 64 Kings Road | DL8 1TZ | avery43092@outlook.com | No | Computing | £584.50 | 09/03/2022 | Chris Woods | 46 Holgate Rd | 07287332311 | B98 0DQ | chrisw334@outlook.com | Essence Hall | CH46 7TB | 50 Town Meadow Lane | 07543854848 | Hadassah Goodman |
| Quinn Cochran | 07898698348 | 75 Mill Road | DG2 0PD | quinncochran@outlook.com | Yes | Sociology | £325.50 | 02/04/2022 | Patrick Cox | 56 Park Row | 07893342413 | SE1 4TR | patrickcox10@gmail.com | Clover Hall | RG14 5NR | 116 Craven Road | 07238483285 | Jamarion Harding |

### 1st Normal Form (1NF):

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Database** | | | | | | | | | | | | | | | | | | |
| **Student Name** | **Student**  **Phone**  **Number** | **Student**  **Address** | **Student Postcode** | **Student**  **Email** | **Achieved** | **Course**  **Name** | **Cost** | **Course**  **Date** | **AssessorName** | **AssessorAddress** | **Assessor Phone** | **Assessor**  **Postcode** | **Assessor**  **Email**  **Address** | **Venue Name** | **Venue**  **Postcode** | **Venue Address** | **Venue**  **Mobile Number** | **Manager Name** |
| Cailyn Dunlap | 07388886854 | 48 High Street | KY112TE | cailynd@outlook.com | No | Business | £120.00 | 29/04/2022 | Ismail Watson | 65 Balsham Road | 07548338743 | N9 9LB | iwatson24@gmail.co.uk | Clover Hall | RG14 5NR | 116 Craven Road | 07238483285 | Jamarion Harding |
| Mayra Cline | 07193853874 | 62 York Road | EC1A4EU | mayracline292@hotmail.com | No | Business | £120.00 | 29/04/2022 | Ismail Watson | 65 Balsham Road | 07548338743 | N9 9LB | iwatson24@gmail.co.uk | Clover Hall | RG14 5NR | 116 Craven Road | 07238483285 | Jamarion Harding |
| Avery Mendoza | 07219327893 | 64 Kings Road | DL8 1TZ | avery43092@outlook.com | No | Computing | £584.50 | 09/03/2022 | Chris Woods | 46 Holgate Rd | 07287332311 | B98 0DQ | chrisw334@outlook.com | Essence Hall | CH46 7TB | 50 Town Meadow Lane | 07543854848 | Hadassah Goodman |
| Quinn Cochran | 07898698348 | 75 Mill Road | DG2 0PD | quinncochran@outlook.com | Yes | Sociology | £325.50 | 02/04/2022 | Patrick Cox | 56 Park Row | 07893342413 | SE1 4TR | patrickcox10@gmail.com | Clover Hall | RG14 5NR | 116 Craven Road | 07238483285 | Jamarion Harding |

### 2nd Normal Form (2NF):

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **2nd Normal Form Table Design** | | | | | | | | | | | | |
| **Student** | | | | | | | | | | | | |
| * StudentID **(PK)** * CourseID **(FK)** * Name * PhoneNumber * Address * Postcode * EmailAddress * Achieved | | | | | | | | | | | | |
| **Assessor** | | | | | | | | | | | | |
| * AssessorID **(PK)** * CourseID **(FK)** * VenueID **(FK)** * Name * Address * PhoneNumber * Postcode * EmailAddress | | | | | | | | | | | | |
| **Course** | | | | | | | | | | | | |
| * CourseID **(PK)** * AssessorID **(FK)** * VenueID **(FK)** * CourseName * Cost * Duration * Date | | | | | | | | | | | | |
| **Venue** | | | | | | | | | | | | |
| * VenueID **(PK)** * AssessorID **(FK)** * VenueName * Postcode * Address * MobileNumber * ManagerName | | | | | | | | | | | | |
| **Student Table** | | | | | | | | | | | | | | | | | | | | | | | | | | |
| **StudentID** | **Name** | | | | **PhoneNumber** | | **Address** | | | | | **Postcode** | | | **EmailAddress** | | | | | **Achieved** | | | | **CourseID** | | |
| AGYD26 | Cailyn Dunlap | | | | 07388886854 | | 48 High Street | | | | | KY112TE | | | cailynd@outlook.com | | | | | No | | | | CG9275 | | |
| OPIH82 | Mayra Cline | | | | 07193853874 | | 62 York Road | | | | | EC1A4EU | | | mayracline292@hotmail.com | | | | | No | | | | CG9275 | | |
| SDOP02 | Avery Mendoza | | | | 07219327893 | | 64 Kings Road | | | | | DL8 1TZ | | | avery43092@outlook.com | | | | | No | | | | CA4324 | | |
| OIVJ34 | Quinn Cochran | | | | 07898698348 | | 75 Mill Road | | | | | DG2 0PD | | | quinncochran@outlook.com | | | | | Yes | | | | CN3858 | | |
| **Assessor Table** | | | | | | | | | | | | | | | | | | | | | | | | | |
| **AssessorID** | | | | **Name** | | **Address** | | | **PhoneNumber** | | | | | **Postcode** | | | **EmailAddress** | | | | **CourseID** | | | | **VenueID** |
| 403822 | | | | Ismail Watson | | 65 Balsham Road | | | 07548338743 | | | | | N9 9LB | | | iwatson24@gmail.co.uk | | | | CG9275 | | | | VEN124 |
| 438933 | | | | Chris Woods | | 46 Holgate Rd | | | 07287332311 | | | | | B98 0DQ | | | chrisw334@outlook.com | | | | CA4324 | | | | VEN922 |
| 489276 | | | | Patrick Cox | | 56 Park Row | | | 07893342413 | | | | | SE1 4TR | | | patrickcox10@gmail.com | | | | CN3858 | | | | VEN124 |
| **Course Table** | | | | | | | | | | | | | | | | | | | | | | | | | |
| **CourseID** | | | **CourseName** | | | | | **Cost** | | | **Duration** | | | | | **Date** | | | **AssessorID** | | | | **VenueID** | | |
| CG9275 | | | Business | | | | | £120.00 | | | 2 | | | | | 29/04/2022 | | | 403822 | | | | VEN124 | | |
| CA4324 | | | Computing | | | | | £584.50 | | | 2 | | | | | 09/03/2022 | | | 438933 | | | | VEN922 | | |
| CN3858 | | | Sociology | | | | | £325.50 | | | 1 | | | | | 02/04/2022 | | | 489276 | | | | VEN124 | | |
| **Venue Table** | | | | | | | | | | | | | | | | | | | | | | | | | |
| **VenueID** | | **VenueName** | | | | | | **Postcode** | | **Address** | | | | | | | | **MobileNumber** | | | | **Name** | | | |
| VEN124 | | Clover Hall | | | | | | RG14 5NR | | 116 Craven Road | | | | | | | | 07238483285 | | | | Jamarion Harding | | | |
| VEN922 | | Essence Hall | | | | | | CH46 7TB | | 50 Town Meadow Lane | | | | | | | | 07543854848 | | | | Hadassah Goodman | | | |

### 3rd Normal Form (3NF):

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **3rd Normal Form Table Design** | | | | | | | | | | | | | |
| **Student** | | | | | | | | | | | | | |
| * StudentID **(PK)** * CourseID **(FK)** * Name * PhoneNumber * Address * Postcode * EmailAddress * Achieved | | | | | | | | | | | | | |
| **Assessor** | | | | | | | | | | | | | |
| * AssessorID **(PK)** * CourseScheduleID (**FK)** * CourseID **(FK)** * Name * Address * PhoneNumber * Postcode * EmailAddress | | | | | | | | | | | | | |
| **Course** | | | | | | | | | | | | | |
| * CourseID **(PK)** * CourseScheduleID **(FK)** * CourseName * Cost * Duration * Date | | | | | | | | | | | | | |
| **Venue** | | | | | | | | | | | | | |
| * VenueID **(PK)** * VenueName * Postcode * Address * AssessorID * MobileNumber * ManagerName | | | | | | | | | | | | | |
| **CourseSchedule** | | | | | | | | | | | | | |
| * CourseScheduleID **(PK)** * AssessorID **(FK)** * CourseID **(FK)** * VenueID **(FK)** | | | | | | | | | | | | | |
| **Student Table** | | | | | | | | | | | | | | | | | | | | | | |
| **StudentID** | | | **CourseID** | | **Name** | | | | **PhoneNumber** | | | | **Address** | | **Postcode** | | | **EmailAddress** | | | | **Achieved** |
| AGYD26 | | | CG9275 | | Cailyn Dunlap | | | | 07388886854 | | | | 48 High Street | | KY112TE | | | cailynd@outlook.com | | | | No |
| OPIH82 | | | CG9275 | | Mayra Cline | | | | 07193853874 | | | | 62 York Road | | EC1A4EU | | | mayracline292@hotmail.com | | | | No |
| SDOP02 | | | CA4324 | | Avery Mendoza | | | | 07219327893 | | | | 64 Kings Road | | DL8 1TZ | | | avery43092@outlook.com | | | | No |
| OIVJ34 | | | CN3858 | | Quinn Cochran | | | | 07898698348 | | | | 75 Mill Road | | DG2 0PD | | | quinncochran@outlook.com | | | | Yes |
| **Assessor Table** | | | | | | | | | | | | | | | | | | | | | | |
| **AssessorID** | | | **CourseScheduleID** | | **CourseID** | | | | **Name** | | | | **Address** | | **PhoneNumber** | | | **Postcode** | | | **EmailAddress** | |
| 403822 | | | QJ3989A | | CG9275 | | | | Ismail Watson | | | | 65 Balsham Road | | 07548338743 | | | N9 9LB | | | iwatson24@gmail.co.uk | |
| 438933 | | | QY3784A | | CA4324 | | | | Chris Woods | | | | 46 Holgate Rd | | 07287332311 | | | B98 0DQ | | | chrisw334@outlook.com | |
| 489276 | | | QX3509D | | CN3858 | | | | Patrick Cox | | | | 56 Park Row | | 07893342413 | | | SE1 4TR | | | patrickcox10@gmail.com | |
| **Course Table** | | | | | | | | | | | | | | | | | | | | | | | | |
| **CourseID** | **CourseScheduleID** | | | | | **CourseName** | | | | **Cost** | | | | | | **Duration** | | | **Date** | | | | | |
| CG9275 | QJ3989A | | | | | Business | | | | £120.00 | | | | | | 2 | | | 29/04/2022 | | | | | |
| CA4324 | QY3784A | | | | | Computing | | | | £584.50 | | | | | | 2 | | | 09/03/2022 | | | | | |
| CN3858 | QX3509D | | | | | Sociology | | | | £325.50 | | | | | | 1 | | | 02/04/2022 | | | | | |
| **Venue Table** | | | | | | | | | | | | | | | | | | | | | | | |
| **VenueID** | | **VenueName** | | | | | **Postcode** | | | | **Address** | | | | | | **MobileNumber** | | | **Manager Name** | | | |
| VEN124 | | Clover Hall | | | | | RG14 5NR | | | | 116 Craven Road | | | | | | 07238483285 | | | Jamarion Harding | | | |
| VEN922 | | Essence Hall | | | | | CH46 7TB | | | | 50 Town Meadow Lane | | | | | | 07543854848 | | | Hadassah Goodman | | | |
| **CourseSchedule Table** | | | | | | | | | | | |
| **CourseScheduleID** | | | **CourseID** | **AssessorID** | | | | **VenueID** | | | |
| QJ3989A | | | CG9275 | 403822 | | | | VEN124 | | | |
| QY3784A | | | CA4324 | 438933 | | | | VEN922 | | | |
| QX3509D | | | CN3858 | 489276 | | | | VEN124 | | | |

# 2.0 Interface Design

|  |
| --- |
| **Data Flow Diagrams** |
| **Forms** |
| **Adding a new student** |
|  |
| **Updating a student’s record** |
|  |
| **Adding Student Achievement** |
|  |
| **Reports** |
| **Automatic generation of the weekly course booking report** |
|  |
| **Creating a trainer performance report** |
|  |
| **Creating a course performance report** |
|  |
| **Creating a course popularity report** |
|  |

## Hardware and Software:

|  |
| --- |
| The software I will be using to create and for users to use the database on is Microsoft Access, Microsoft Access is a graphical application for running and managing databases instead of manually through code. Additionally, Microsoft Access provides many utilities which make certain parts of the database automatic, one of these examples is primary/foreign keys, referencing and referential integrity which will take care of all the relationships automatically where the only thing I will need to do is create each table and specify the attributes and relationships in/of those tables.  Since Microsoft Access is a Microsoft application, I will be using the Windows operating system also developed by Microsoft in order to create and run the database. Users who are interfacing with the database who may want to retrieve and edit data in the database will need to have Microsoft Access installed which will only work on Windows. Using Windows also provides the advantage of familiarity, simplicity and compatibility, since the users of the database will most likely, already be using Windows, each person who needs to interact with the database such as users and database administrators only need to install Microsoft Access to use the database.  The hardware needed to create and run the database is a basic office/personal computer capable of running Microsoft Access and Windows 10 or optionally, Windows 11. Microsoft Access and Windows 10 both have a specification of minimum system requirements needed to run. The minimum requirements Microsoft Access needs to run is a Dual Core 1.6ghz processor, 4gb of RAM, DirectX 10 compatible Graphics Card, 4gb of storage space and a Windows 10/11 operating system. Windows is also similar, with Windows 10 only needing a 1ghz processor, 2gb of RAM, DirectX 9 compatible graphics card and a 16gb hard drive.  Every device I will be using to create and test the database will all follow these requirements with the slowest devices I will be using having a 6 core 3.6ghz processor, 16gb of RAM and a graphics card which supports DirectX 12. This is well above the minimum requirements of Access and Windows 10, so it is adequate to create, test and use the database. |

## Test Plan:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Plan** | | | | | |
| **Test No** | **Purpose** | **Test Data** | **Expected Outcome** | **Actual Outcome** | **Further Action** |
| **1** | Testing the validation of phone numbers in the student, assessor and venue table’s by entering an invalid phone number which is too short. | PhoneNumber: 073984 | When the user attempts to set the field as the number, an error message should popup informing the user that the phone number was not valid, the message box should display the following:  “An error occurred, the phone number entered was not a valid phone number, please try again!” | The database blocks the user entering the value, a message box also displays saying:  “An error occurred, the phone number entered was not a valid phone number, please try again!” |  |
| **2** | Testing the validation of phone numbers in the student, assessor and venue table’s by entering a valid phone number | PhoneNumber: 07733492252 | The field’s value should be overridden and set to the new input. | The phone number field was changed to the new phone number: 07733492252 |  |
| **3** | Testing the validation of email addresses in the assessor and student table’s by entering an invalid email address. | EmailAddress: emailaddress AT emails DOT co DOT uk | The database should prevent the user from entering the email by showing an error message which should display the following:  “An error occurred, the email address entered was not valid, please try again!” | The database displayed this message:    The field was not changed to the email address entered. |  |
| **4** | Testing the validation of cost/currency by entering a negative value | Course Cost: -£25.91 | The database should detect whether the cost is lesser than 0 and prevent the user from entering the value. The database will display the error message:  “An error occurred, the price entered wasn’t valid, all costs have to be positive!” | An error message is displayed:  The value was not changed. |  |
| **5** | Testing the validation of cost/currency by entering a valid positive value. | Course Cost: £259.10 | The value of the field should be changed to the new cost. | Old value:    New Value: |  |
| **6** | Testing the add student form adds a new student with all the fields correct. | New Student:  ID - IWQD23  Name - Hadley Lopez  Postcode - GL1 1UP  Address - 5 St Davids Road  EmailAddress - hlopez85@gmail.com  PhoneNumber - 07223291123  CourseID - CL9354 (Health & Safety)  Achieved - No/False | A new record should be added to the student table with all the test data. | A new record has been added to the student table with all the fields set to the data set in the test field. |  |
| **7** | Testing the update student form record navigation. | N/A | When pressing the record navigation buttons, the form should scroll through the list of student records in the student table. | Before:  Pressing the right navigation button: |  |
| **8** | Testing that the update student form updates the fields for the correct student. | Student to update: IWQD23  Data Changed: Address - 14 Green Street  Postcode - GL10 2PE | When the update button is pressed, a message should display informing the user of the action which will take place, if the user continues, the selected record should have its information changed. | Before:  After: |  |
| **9** | Testing the add student achievement form record navigation. | N/A | When pressing the record navigation buttons, the form should scroll through the list of student records in the student table. | Before:  Pressing the left navigation button: |  |
| **10** | Testing the add student achievement form updates the selected students record. | Student to update: IWQD23  Data Changed:  Achieved - True/Yes | When the update button is pressed, a message should display informing the user of the action which will take place, if the user continues, the selected record should have the achieved field updated. | Before:  After: |  |
| **11** | Viewing the course booking report to see if it displays the correct data. | N/A | Pressing the “course booking report” button on the report form should display a brand-new generated report which displays each individual course in the courses table along with each student which is on that course. | A new report is generated and displayed: |  |
| **12** | Viewing the course performance report to see if it displays the correct data. | N/A | Pressing the “course performance report” button on the report form should display a brand-new generated report which displays the name and ID of each course along with a percentage column.  This percentage column per course is the percentage of students on that course who have achieved a grade on that course. | A new report is generated and displayed: |  |
| **13** | Viewing the course popularity report to see if it displays the correct data. | N/A | The course popularity report should display each course along with the number of students within that course. | A new report is generated and displayed: |  |
| **14** | Viewing the trainer performance report to see if it displays the correct data. | N/A | The report generated from pressing the “trainer performance report” button on the report form should list each assessor in the assessors table along with the course that they are teaching on. | A new report is generated and displayed: |  |
| **15** | Testing that each view report button works on the report form. | N/A | Pressing a button should immediately bring up the report that the button was linked to. There are 4 buttons on the page corresponding with: “View Course Booking”, “View Trainer Performance”, “View Course Performance” and “View Course Popularity” | Each button when pressed displays each report in a print preview as can be seen in the previous tests. |  |

## Implementation Plan:

|  |
| --- |
| In order to create the database and ensure that it is fully functional, I will need to create an implementation plan to plan out how I will implement the database according to the requirements set out by Workplace4Training. Firstly, I will create and plan the design for the database including how the database will be structured, how the tables in the database will be structured and the relationships within the database. I have been given a set of data for which to base the database on, I will go through the normalization process for the given set of attributes and place them in the appropriate table matching the requirements of the client.  I will create the appropriate documentation in order to complete this process such as entity relationship diagrams, data dictionaries and normalisation tables taking the database through each step of normalization (0NF, 1NF, 2NF, 3NF). This documentation will be useful in creating the database as I can fully plan out what the database will look like first encountering mistakes in the design phase instead of the implementation phase which is much easier to rectify.  Additional documentation will be created for the algorithms needed for certain parts of the database to function, algorithms like report generation will be documented using data flow diagrams. Using data flow diagrams means that I can specify what data each report needs to retrieve and how it will be used. Other diagrams such as layout design will be created for the forms/user interface in the database later after the database has been designed, when the user interface needs to be added, the diagrams can be used as a reference.  Once all the relevant documentation has been completed, the implementation of the database can start where all the documentation from before will be used. Tables and relationships will be completed first along with forms, then, the logic/functionality of the database can be created through queries. Queries will need to be created based on the previously created data flow diagrams to do things such as, provide read and write access through forms and retrieve relevant data for reports. |

## Maintenance/Support Plan:

|  |
| --- |
| A technical guide along with the documentation created for the database such as the ERDs, Data Dictionaries, DFDs and general diagrams/documentation will be created so that future maintenance to the database can occur. Future maintainers can use the technical guide along with the documentation to maintain the database. For example, if the scope of the database must be changed with new tables or attributes needed, the maintainers can look at the ERD or data dictionary to figure out where the data must be inserted.  The purpose of the technical guide is to provide the future maintainers with much less work when modifying the database. Additionally, bugs and issues would be much less likely to occur as the future maintainers will be able to look at the documentation/guide to figure out how the database functions. Additionally, the guide can explain the reasoning for each decision made in the database, such as to what a certain query is for and to why certain data was selected for reports and forms along with the purpose/goal of each report/form. |

## Project Timescale:

Overall, the database will take 8 weeks to complete from the client specifying the requirements to the final testing/deployment phase. The table below illustrates what will happen during the 8 weeks.

|  |  |  |
| --- | --- | --- |
| **Project Timescale** | | |
| **Week** | **Summary of work** | **Description** |
| **1** | Basic table design with project beginning | The most critical part of the database will be the tables in the database as every other part of the database such as forms and queries will depend on how the tables are structured and what data is in them. The raw list of attributes/data needed provided by the client is not suitable as a database, so work will immediately begin to normalise the data provided into their relevant tables.  Documentation will be created such as a data dictionary and entity relationship diagram to be used as reference later in the implementation phase of the project. |
| **2** | Design report | During this week, the queries and reports specified by the client will be designed and created, whatever information the client has requested for each report to contain will be added into the reports structure. |
| **3** | User interaction with the user interface | Consideration needs to be given on how users will interact with the database, the methods in which the users interact with the database will be through forms presented as a GUI. Each functionality of the database needed such as adding, deleting and modifying data will be added into their own separate form. Once the forms have been decided, the design for the interface for each form will begin in the form of a diagram/wireframe. |
| **4** | Design evaluation | The design phase of the database has been mostly completed; the last thing left to do is to go back through each stage of the design phase to check if each part of the database design makes sense. Each part of the database such as tables, queries, reports and forms are all linked together and will have to be checked to ensure that there are no mistakes/misunderstandings. |
| **5** | Tables and database production | During the 1st week, a new database on Microsoft Access will be created, using the documentation created in the weeks before, the entity relationship diagram and data dictionary will be used to add the tables and their relationships to the database using the tools provided by Microsoft Access. |
| **6** | Logic and query designs | The data flow diagrams, and report designs created before will be used to implement all the needed queries and reports into the database. The data flow diagrams list all the steps and data needed for the reports and queries. The queries will be created from these diagrams which will then be linked to the reports. |
| **7** | Testing of query and report | In order to test the databases functions, data will be needed to fill in each table, during the start of this week, sample data will be generated and added into the database meeting certain criteria (such as a minimum number of students per course or every record needs to reference or be referenced by at least one other record).  Once added, testing can begin for the queries and reports through the forms created to interact with the queries and reports. |
| **8** | Finishing overall test | After a week of testing, issues which may have been identified will need to be fixed, after fixing the issues raised, further testing will be needed to make sure that the fix itself is working, and that other parts of the database are not broken by the fix.  Once all of this is done, the client will be notified that the database has been completed and the database will be handed over to them with all the sample testing data removed ready to be actively used. errors/suggestions provided by the client will then need to be fixed/changed. |

## Technical Constraints of Hardware/Software:

|  |
| --- |
| For the database, I will be using Microsoft Access as Access provides a very easy and simplistic way to create, manage, interact with and use databases. For users to use the database, the user's device must have Microsoft Access installed, this is fine for Windows based computers but for other devices such as mobile devices or Linux and MacOS, the devices will be unable to use the database.  The latest/supported versions of Microsoft Access, unlike most other Office/Microsoft products, are only compatible on Windows 10 and 11 meaning that users who use Android, MacOS, Linux or older versions of Windows will have outdated versions.  This limitation of Microsoft Access means that the client will have to use Windows across all their computers. This may be an issue as some of the client’s devices may be incompatible with Access. If the client has devices which use MacOS or Linux as their operating system, the device/operating system may have to be swapped out if that device needs to access the database. |

## Email to client:

|  |  |
| --- | --- |
| I will send an email to the client (Rahat) informing them of the plan for the database including all the factors which will influence the database when in final use such as the technical constraints and user interaction | |
| **Email** | |
| **To:** | [rahat123@gmail.com](mailto:rahat@gmail.com) (Client) |
| **From:** | [lmcrailt@gmail.com](mailto:lmcrailt@gmail.com) (Manager) |
| **Subject:** | Proposal, timelines and plan for Workplace4Training database |
| Dear Rahat,  The design proposal for the database has been completed, firstly, the database will be implemented in Microsoft Access. Microsoft Access will allow you and any other employee to use the database. Microsoft Access was chosen because of its many advanced features and functions which make many parts of the database functionality completely automatic, in addition to less bugs and issues with a manual implementation, Access comes with a range of functionality such as built in Micro GUI’s known as Forms for which database functionality (Such as adding, deleting and modifying data) can be done.  The use of Microsoft Access however will require that all devices/users which need to use the database to be using the Windows OS versions 10 or 11 since Access is only compatible with these versions of Windows. Other operating systems like Linux, MacOS and Android cannot be used for the database, so consideration will need to be taken for what devices you or managers or employees will be using to use the database.  The database has been evaluated to be completed within 8 weeks, the 8 weeks will include initial design and planning, development/implementation, testing and feedback from you along with improvements. A test plan has been devised with each phase/stage getting the appropriate time needed to be completed to ensure the best possibility of the database being completed within those 8 weeks.  The database itself will consist of 5 tables each for students, assessors, courses, course schedules and venues which will all link to each other. The appropriate information for each tables attributes has been identified and added in the plan for the database table structure. Users will interact with the database through the previously mentioned forms, forms will allow users to interact with the database easily through a GUI layer above the raw tables/queries. The reports specified to be generated weekly have also been designed, the reports will consist of course booking, course performance, course popularity and trainer performance reports all requested in the requirements of the database.  If you have may have any suggestions or changes which you want to make, reply and consideration will be given towards those suggestions and improvements.  Kind Regards,  Anonymous | |

## Meeting notes:

|  |
| --- |
| A meeting has been held between the project manager, client and developers to discuss the design of the database along with other parts of the database such as the user interface, screen design and reports. These are the key notes from the meeting:   * The existing forms will all need buttons to perform the action of the form. Each form is delegated to one specific interaction within the database. Because of this, the user will need a way to start the dedicated action which the form was created for. This means that there will need to be a minimum of 1 button per form. * All user facing parts of the interface such as reports and forms will need to be easy to use and accessible for users, this means that parts of the interface such as labels, textboxes and buttons will need to be clearly identifiable. E.g. The ID box on a form should have a label pointing out that the textbox is for an ID. Easy to use interfaces also include neat and good-looking design’s including the use of colour and spacing meaning that each form and report will need to have some level of extra colour detail. * The forms should leave the least amount of work necessary to the user, the point of the interface/forms is for easy to use and convenient access to the database. This means that for example, when cycling through a list of record primary keys in a table, the rest of the data/fields from the record containing the primary key will be automatically filled in.   After further analysis from these notes, these improvements and tweaks have been identified:   * Missing titles have been added to each form and report design. * Colour has been added to the forms along with additional spacing to break up the interface design. * Record navigation functionality has been added to forms which involve updating/modifying data in tables meaning that users can click navigation buttons to cycle through the list of records within a table. |

## Review of Designs:

|  |
| --- |
| After carefully reviewing the information from the meeting which has been identified in the meeting notes section and through my own analysis, the design of the interface will have some minor adjustments added to it. These adjustments will help to improve the user experience of the database. While these are not major adjustments, they will help in long term use of the database to help avoid any user error/confusion. |

## Updating of designs:

|  |  |
| --- | --- |
| After reviewing the initial designs and after meeting with the manager, I have reviewed the designs and have decided to modify them. The modifications will be mostly visual for clarity/easy identification of parts of the interface such as labelling which will improve the user experience, the updated designs can be seen below: | |
| **Updated Form Designs** | | |
| **Add Student Form** | **Update Student Form** | |
|  |  | |
| **Add Student Achievement** | **View Report Form** | |
|  |  | |

## Evaluation of designs:

|  |
| --- |
| After the careful design and analysis which helped to create the initial design based on the client's needs, and the modifications made to the designs after speaking to the client, overall, the designs are satisfactory for the client and will be used for the user interface. The design provides an easy-to-use user interface with accurate labelling and separation of sections of the interface through techniques such as colour. |

# 3.0 Implementation of database

## Evidence of tables:

|  |  |
| --- | --- |
| All tables are presented in the spreadsheet view on Microsoft Access, each tables attributes can be seen in the yellow bar at the top of the image. The records for each table are listed vertically in each image. Each table has the primary keys as the far-left column, foreign keys are located directly adjacent to the right. | |
| **Database Tables** | | |
| **Student Table** | **Course Table** | |
|  |  | |
| **Venue Table** | **CourseSchedule Table** | |
|  |  | |
| **Assessor Table** | | |
|  | | |

## Evidence of relationships:

|  |
| --- |
| This is the relationship view in Microsoft Access on the database, as you can see, the relationship tab shows the exact same relationships as the entity relationship diagram created previously. Each line represents a relationship between each table, the symbols at the end of each line represent the type of relationship, this can be seen with the one-to-many relationships with the symbol “1” representing the “one” end of the relationship and the infinity symbol “∞” of the relationship representing the “many” end of the relationship. One-to-one relationships are represented by a line with 2 small dots at either end. |
|  |

## Evidence of validation:

Validation for fields is implemented using validation rules, a new value entered must meet the criteria set in the validation rule, if the new value meets the criteria, the value will be changed, otherwise, an error message will be displayed which displays text in the “Validation Text” box below the “Validation Rule” box.

|  |  |
| --- | --- |
| **Validation** | |
| **Email Address Validation** | |
| **Validation Rule and Error** | **Result of invalid data: “invalidemailaddress AT email.com”** |
|  |  |
| **Phone number validation** | |
| **Validation Rule and Error** | **Result of invalid data: “A3RU8”** |
|  |  |
| **Course Cost Validation** | |
| **Validation Rule and Error** | **Result of invalid data: “-£34.20”** |
|  |  |

## Evidence of form:

|  |  |
| --- | --- |
| **Forms** | |
| **Add Student Form** | |
| **Design View** | **Form View** |
|  |  |
| **Update Student Form** | |
| **Design View** | **Form View** |
|  |  |
| **Add Student Achievement Form** | |
| **Design View** | **Form View** |
|  |  |
| **View Report Form** | |
| **Design View** | **Form View** |
|  |  |

## Evidence of queries and reports:

|  |  |
| --- | --- |
| **Queries and reports** | |
| **Course Booking Report** | |
| **Query Design** | **Report View** |
|  |  |
| **Course Performance Report** | |
| **Query Design** | **Report View** |
|  |  |
| **Course Popularity Report** | |
| **Query Design** | **Report View** |
|  |  |
| **Trainer Performance Report** | |
| **Query Design** | **Report View** |
|  |  |

## Password protected database:

|  |
| --- |
| To secure the database from any threats/breaches, I will encrypt the database using a password which will be provided to the client, encrypting the database ensures that even if malicious hackers get access to the database file, the data inside the database will still be safe.  I have set the password to the database by using a function provided by Microsoft Access, this can be seen below where the password for the database will be set:    The password for this database is “Workplace4Password” which is a basic example password which can be changed later by the client. Setting the database password as this means that whenever an attempt is made at opening the database, an entry box will show asking for the user to enter a password, which can be seen below:    After entering an incorrect password:    And after entering the correct password which is “Workplace4Password”:    The user gets complete access to the database as if it were unencrypted. |

# 4.0 Final evaluation of database

## User and client requirements:

|  |
| --- |
| Overall, the completed database meets the requirements of the client, the client has specified that the database is to store data for Workplace4Training for data such as students, courses, assessors and venues. The table structure of the database allows for the appropriate relationships between each record in each table meaning that the database is suitable for its role in storing and managing students and the courses they are on.  Additionally, the database carries out all the set-out functionality in the clients' requirements, the database can perform operations such as modifying a student's record either to update some fields within the record and to add the student achievement. The database has also had reporting functionality built into it, with a dedicated form created for managers to be able to access reports. The database contains 4 main reports which each collects data from the appropriate tables.  These 4 reports are the course booking, course popularity, course performance and trainer performance reports. Using these reports, the manager view which ever data they need to see about the database. E.g. If a manager wants to see the performance of students across one course, the manager can click a button on the report form to view the “course performance report” which will list the percentage of students who have achieved the course. Using built in Access reports is very beneficial as reports can have access to queries and various other features such as printing which means that reports can be easily printed and archived. |

## Quality of the relational database:

|  |
| --- |
| The database has been designed from scratch from the old system using a list specified by the client about what the database should store. The client has provided a list of data which has to be available in the database, using this list of data, the database table structure and the resulting relationships have been planned and documented. The list of data has gone through the complete process of normalization providing a fully functioning and error free database.  Using normalization means that the appropriate relationships between each table can be setup, using relationships in the database means that many of the processes to viewing and modifying data in the database are fully automatically. This provides an advantage as records in the database can be linked together. For example, if a manager needs to find out what venue a course will be held at for an assessor, the manager can use the established relationship between the course table, course schedule table and finally the venue table to find the venue record which the course has been linked to rather than manually searching through the database, the database can automatically look up this value.  The use of normalisation and relationships in the database also comes with the advantage of referential integrity and cascading updates. When a record in the database needs to have its data changed, the database only must update the data in that record once. Normalisation aims to reduce repeating data as if data is repeated multiple times within a database, the data can be mismatched. This is because when that piece of data needs to be updated, each record which contains that repeating data will also have to be updated. This is prone to errors as if the change is missed for that field/data in one or more records, the data mismatch will occur which can result in confusion, frustration and more severely, lost data.  Instead of repeating data, normalisation will instead fill each record which needs to access that repeating data have a reference to another table which contains the data needed. E.g. If the student records each contain a reference ID to a course, and the course contains a reference to a venue, if the venue of the course must be changed, only the venue record must be changed once instead of updating the venue per student.  Additionally, the database provides an easy to use and accessible method for users to interact with the database, users do not have to directly interface with the database through Access to perform the action they want. Instead, I have setup forms which are mini-GUIs embedded in the database which connect to the database functionality to provide an extra more user-friendly layer of interaction. Instead of for example, adding a new student to the table directly, which may be confusing to some users, there is a form setup dedicated to handling adding new students to the table. The form contains widgets/controls such as textboxes, combo/dropdown boxes and buttons which link to the table where the user can enter in the data which the new student record will contain. |

## Fitness for purpose:

|  |
| --- |
| The goal of this project was to replace the original databases which assessors/users/managers had trouble with due to frustration. The result of this project needs to be a produced database which must check each criterion of specified functionality and use for Workplace4Training to be able to integrate the database into their organization for use. The produced database has met all these requirements as it provides a new relational database which addresses the previous concerns with the old system while providing extra functionality such as reports and forms. |

## Suitability against the original requirements:

|  |
| --- |
| The original requirements of the database specified by the client was a relational database which stores data about students, assessors, courses, venues and course schedules. Additionally, the requirements have also specified that the database is to carry out special and specific functionality such as creating reports and providing a user-friendly method of interaction with forms.  Workplace4Training needs the database in order to store the information needed for the organization to function. The new database eliminates many of the concerns addressed by managers and assessors when trying to use the old system/database. Some of these concerns like having to manually update each individual record across multiple databases when one singular bit of data was changed, the implementation of the new relational database has eliminated this concern in the design of the database and the type of database being relational which is perfect for requirements such as these. |

## Legal and ethical constraints:

|  |
| --- |
| Since the database will be ran by an organization, the organization (Workplace4Training) will be subject to various data protection and privacy laws such as the Data Protection Act (1998) and GDPR (2018). The laws provide guidelines on what companies and organizations are and are not allowed to do with customer data Aswell as additional rules for data protection. These laws must be complied with as the database cannot be used if it isn’t compliant.  Examples of what these rules enforce is complete transparency in what data businesses have stored about customers or people who have data on them in the database. Users/Customers (in this case students) must be able to see what data the organization has stored about them; this may mean that the record in the database stored about the student must be provided on demand. This in and of itself isn’t too much of an issue as the appropriate student's record can be retrieved and sent to the student.  However, where there may be issue is with data security and breaches, in addition to increased rights of users/customers regarding access to their data, GDPR and the DPA also need organizations to ensure that data confidentiality is held up regarding all user/customer data. This means that organizations must ensure that all breaches and data leaks are prevented at all costs, failure to do so will result in fines up to 4% of the organizations total profits. Data breaches if severe enough, or recurring enough, can also come with more severe punishments such as a total ban on any kind of data storage about customers/users, this would not be ideal and will mean that the database is unusable.  To avoid the issue of security breaches, the database administrators and organization must ensure that data is not accidently or purposely leaked. This can be done by using methods such as encryption with the database being password protected. The database has had a password setup so that anyone who tries to access the database without the password (likely for malicious purposes) will be unable to access the database. The password set to the database acts as the encryption key to the database meaning that no matter what, even if hackers get access to the database file/steal the file, the database will still be protected as it is encrypted. |

## Technology constraints:

|  |
| --- |
| The platform used to implement the database is Microsoft Access, Access was chosen for the database for its advanced functionalities such as reports, forms and query builders while also handling many backend things such as referential integrity. Access is suitable for this purpose because it its intended to be used by small organizations as opposed to other database software such as SQLite or MySQL. However, developing on Access comes with its own difficulties due to bugs and errors present in Access which may affect the database either in development, testing or live use.  Access also comes with a minimum set of system requirements; Access is a large application which performs many complex tasks and processes meaning that it will need a somewhat powerful system. Microsoft has listed a specification of minimum system requirements for Access and subsequently the database to function, even if a system meets these system requirements, Access can still be constrained by poor hardware as the database can still be slow or crash.  This will mean that any user with a device which needs to access the database will have to ensure that the device they are using meets the minimum system requirements set out on Microsoft's website. |

## Strengths and alternative solutions that could be implemented:

|  |
| --- |
| The advantage of using Microsoft Access is the quick and easy creation of new databases, Access will do most of the work while the segments of the database and components attached to the database such as user interfaces are being created. For example, Access has a built in SQL engine for creating queries, this query engine can be used to manually write queries or alternatively, queries can be generated automatically by filling in the information which the query needs to retrieve.  Access also provides great tools regarding interfacing with the database, Access has support for GUI/form mini applications within the database which can be setup to interact with the database. Forms can be created for any purpose such as viewing data, creating and viewing reports, modifying records within tables, deleting records, adding records and many other actions.  However, Access fails regarding compatibility and in errors/bugs. The latest most supported versions of Access (currently Access 2019) are only available on Windows 10 and 11, any other operating system or older version of Windows does not support Access meaning that users are limited in the choice of operating system if they need to regularly access the database. Other database management systems such as MySQL and SQLite do not face this issue as both have been written to be compatible across most operating systems including MacOS, Linux and Android. |

## Platforms and compatibility:

|  |
| --- |
| Since Access has been used to implement the database, the database is only compatible with operating systems and systems which are also compatible with Access meaning that certain platforms may not be supported. Because Access is a large complex application, Microsoft hasn’t attempted to port Access to other operating systems such as MacOS and Linux and is only available on Windows 10 and 11.  This presents an issue with the database as users who need to access the database on other operating systems such as MacOS, Android, Linux or older Windows versions will be unable to use the database. Microsoft Access is setup not only as a database creation software, but also a management and general use software for Access created databases, this means that to use a database created in Access, Access must also be used to open the database. |

# 5.0 Evaluate your skills

## Your time management and planning:

|  |
| --- |
| Throughout the project I have effectively planned and managed my time by my use of documentation and planning documents. To implement the database correctly while also meeting the clients' requirements, I planned what the database will look like beforehand and how I will do it. At the immediate start of the project, I estimated how long the project will take based on the various things that I need to do for the project. I then created a project timescale table over these 8 weeks which will list what tasks need to be done at each stage of the 8 weeks.  Using a timescale table has allowed me to give myself the best chance of completing the project on time based on what work needs to be done and my abilities. Breaking down each section of the project such as the design phase, implementation phase and testing/deployment phase allows me to more accurately estimate how long each phase and the different parts of the phase will take. It has also allowed me to assess and evaluate my work as I dedicated some time over the project timeline to evaluating what was done so far allowing me to improve it even more.  To assist me in implementing the database, I created planning documents and documentation such as Data Dictionaries, Data Flow Diagrams, Entity Relationship Diagrams, Test Plans and interface design wireframes. Creating these at the start of the project helped me during the implementation phase as I can use the planning documents as references meaning that I can implement the database accurately to what the client has specified since the client has approved of the planning. |

## How you used the feedback from your manager:

|  |
| --- |
| When planning the database, I also needed to plan how the user will interact with the database and what platform the database will be on. Therefore, when I was planning the database, I got feedback from the manager of Workplace4Training about the proposal for the database. Firstly, I sent an email to the client informing them of the proposal for the database. In the email, I discussed the plan for the database such as to what platform the database will be held on where I proposed Microsoft Access and its advantages and disadvantages related to the client and informed them of how users will interact with the database.  After this, A meeting was held between the client, Me and the rest of the team to discuss the proposal for the database. In the meeting, things such as the user interface were discussed with the client making suggestions and improvements which they would like to see or be added onto the already existing plan. Using these suggestions was vital in improving the design of the user interface and after the meeting ended, the suggestions made within the meeting were evaluated and the plans for the user interface were modified to suit the client's needs.  These suggestions spanned from general suggestions such as what they would prefer to see within the designs to general specific improvements after showing them some of the designs. The designs were changed in order to suit the client's needs and were added to the documentation. |

## How you behaved on the project - professionalism, etiquette, supportive of others, timely and appropriate leadership, accountability:

|  |
| --- |
| While the project was underway, I consistently lead the project appropriately and effectively resulting in a high-quality database being produced. Throughout the project I took consideration for team and the client by consistently communicating with them through emails and through meetings. I kept the project constantly open to criticism so that if the client/team wanted to raise any suggestions, they could mention them to me, and I would consider them in changing parts of the project or for future parts of the project.  I kept professionalism throughout the project by the consistent communication with the client, the client would know about every part of the project and how it was going. I informed the client about all the details at the start and throughout the project by letting them know important details such as how long the project was going to take/the timescales for different phases of the project and how the project was going to be implemented.  I also took consideration for others through my planning and documentation not only for the client/team but also for future maintainers of the database. I have created a support guide with information about the database and all the planning documentation such as user interface wireframes, diagrams and data dictionaries so that whenever the team or future maintainers needs to understand a certain part of the database design/implementation they can refer to the support guide and documentation. |

## Your recommendations and decisions:

|  |
| --- |
| When planning the project, I had made a proposal to the client about the design and implementation for the database, the proposal included many of the important details about the database such as what platform the database would be implemented in. I made recommendations about the platform considering the advantages and disadvantages of the platform proposed which was Microsoft Access. I listed the advantages and disadvantages which would affect the client so that they can fully understand and make a decision.  Within the proposal, I also provided the suggested timeframes to complete the project and the project timescale which was 8 weeks. Additionally, I also proposed the design implementation of the database considering what data the client asked for the database to store. The proposal to the client also allows me to get any suggestions from the client if they had any issues with my proposal/suggestions. |

## Targets to obtain insights into own performance:

|  |
| --- |
| While on the project I needed consider my own performance while managing and leading the project. When the project was underway, I gave the project targets which would give insight into my own performance. Some of these targets were for examples, milestones from minor ones such as following a part of the project timeline perfectly to major ones such as completing the project on time while delivering a perfectly functional database.  There are many targets for this project which if met or not would determine my performance on this project. Failing only a few or a significant number of these targets will mean that I have performed poorly on this project. However, since the project has been successful in delivering a fully functional database with all the client's requests and targets it means that I have performed well. Additionally, I can use the insight gained from these targets to determine what areas of my performance need improving. |