# **Mohammed mahin ibnay mamun (346584)**

**Unit 18 Assignment 2**

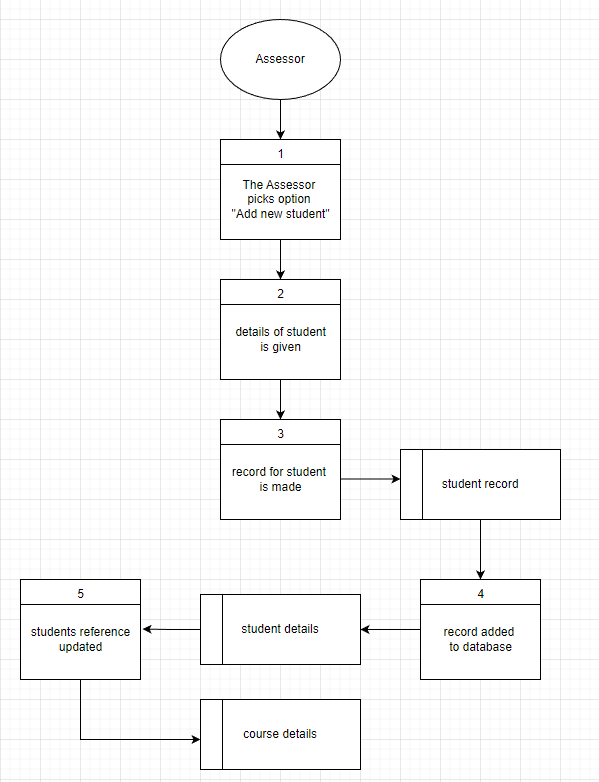
**1.0 Data Structure**

## **Data Dictionary**

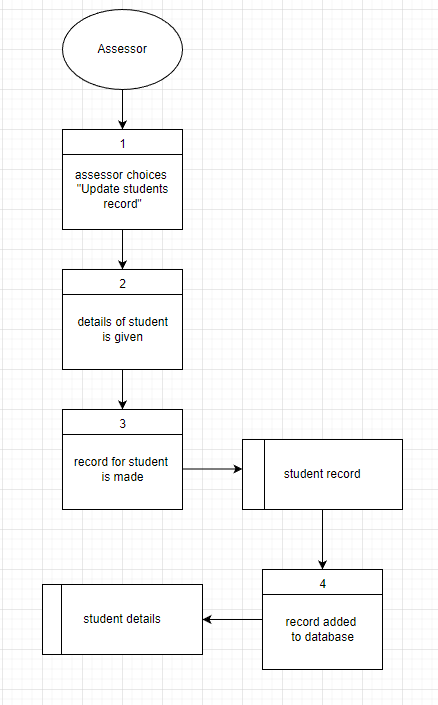
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Table** | **fields** | **Data type** | **Size / format** | **Validation** | **Input mask** | **note** |
| **Assessor details** | Assessor name | string | Easy to read text | No letters or symbols | MMIM028 |  |
| Assessor ID | Short string | integer | Is not null |  | PK |
| Assessor Address | Long string |  | Min 1 space |  |  |
| Post code | string | 7 | 6 – 8 characters | \*\*\*\*’ ‘\*\*\* |  |
| Assessor mobile number | Multiple Integer string | integer | 11 int values | 00000000000 |  |
| Assessor email address | Long String |  | Must include '@'  character | \*\*’\*@ |  |
| Course Details | Course name | string | Easy to read text | Is not null |  |  |
| Course ID | string | integer | Is not null | CS9032 | PK |
| Course cost | Int value | 2 d.p | Symbol and int values | £0,000 - £00,000 |  |
| Venue  details | Venue name | string | Easy to read text | No int values |  |  |
| Venue ID | string | integer | Is not null | 25 | PK |
| venue address | Long string |  | Must include 1 @ symbol | \*\*’\*@ |  |
| post code | string | 7 | Must include 1 space | \*\*\*\*’ ‘\*\*\* |  |
| venue mobile number | Multiple Integer string | 11 | No alphabet letters | 0000 000 0000 |  |
| manager name | string | Easy to read text | No int values |  |  |
| venue email | Long string |  | Must include 1 @ symbol | \*\*’\*@ |  |
| address | Long string |  | Must include 1 space |  |  |
| venue cost | Int | integer | Currency symbol and int values | £0,000-£000,000 |  |
| google map link | Long string | hyperlink |  |  |  |
| Course schedule details | course date | String with int | dd/mm/yy | Is not null | dd/mm/yy |  |
| course duration | Time / date | HH:MM dd/mm/yy  -  HH:MM dd/mm/yy | Is not null | HH:MM dd/mm/yy  -  HH:MM dd/mm/yy |  |
| achieved | Boolean | Y/N or YES/NO | Boolean output | Y/N or YES/NO |  |
| Student details | student name | string | Easy to read text | No int values |  |  |
| student id | string | integer | Is not null | 0000000 | FK |
| address | Long string |  | Must include 1 space |  |  |
| post code | Long string | 7 | Must include 1 space | \*\*\*\*’ ‘\*\*\* |  |
| email address | Long string |  | Must include 1 @ symbol | \*\*’\*@ |  |
| student mobile number | Multiple Integer string | 11 | No alphabet letters | 0000000000 |  |

## **Data Flow Diagram**

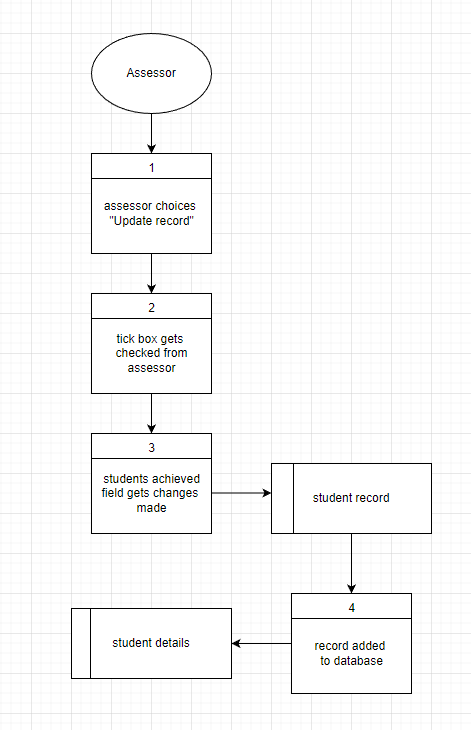
**Adding data of a new student**



**Updating Record of Students**

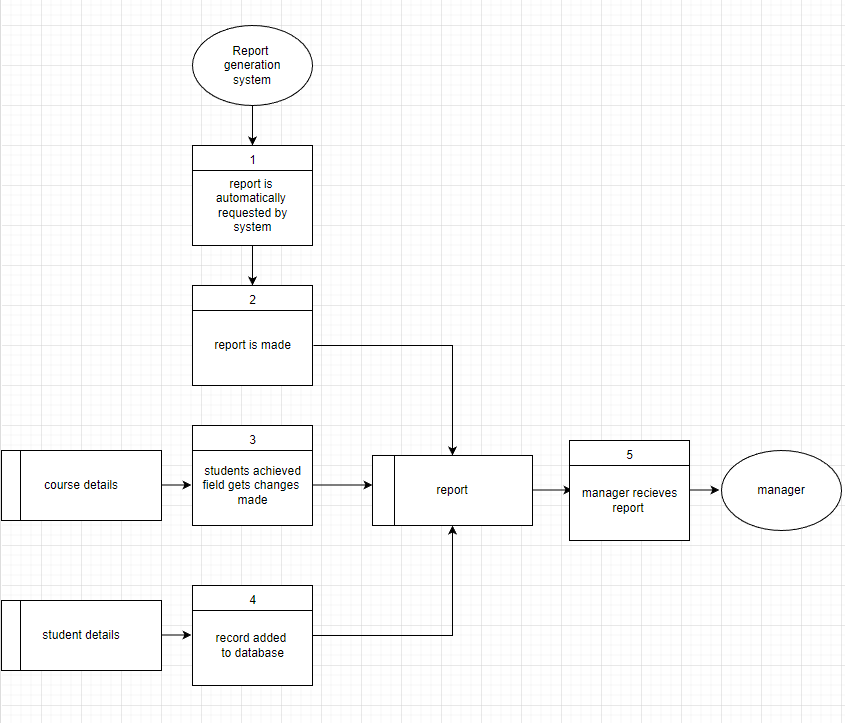


**Adding Student Achievement**

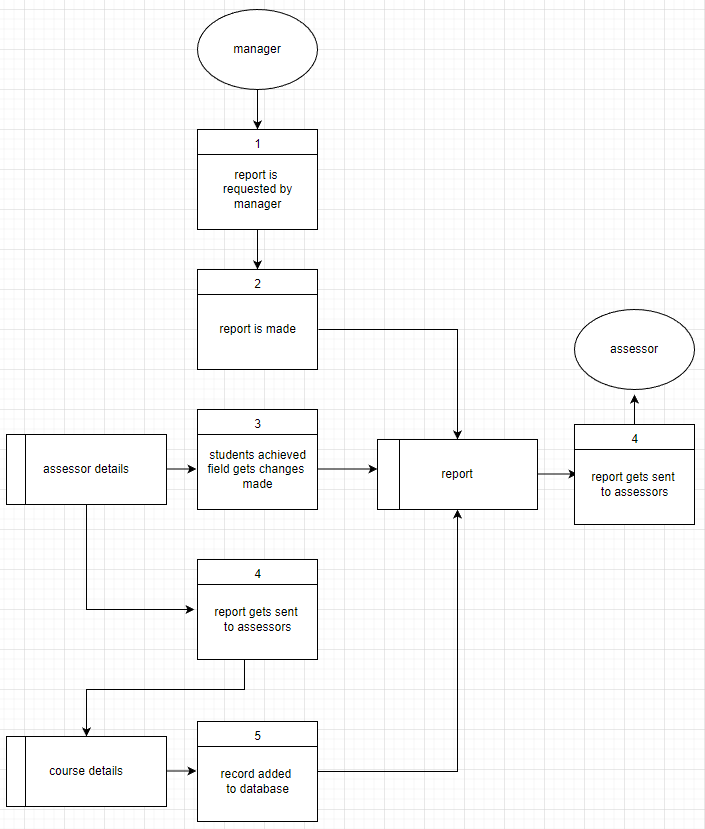


**Manager**

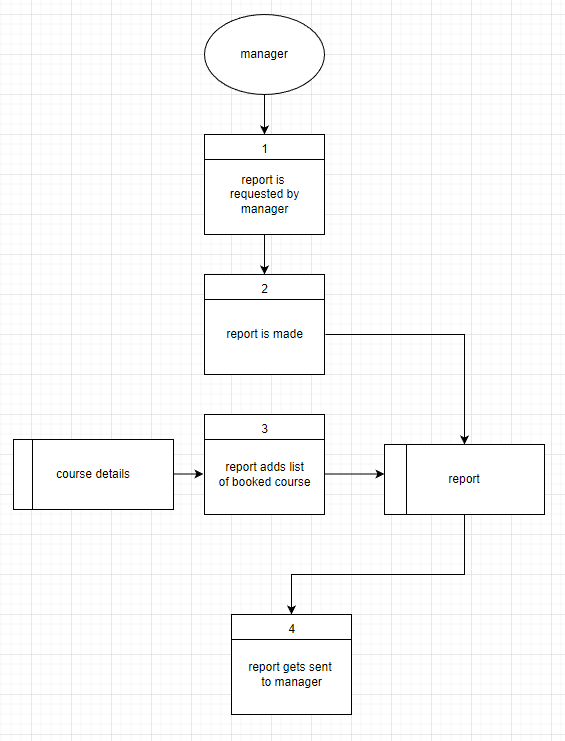
**Automatic generation of weekly course booking report**



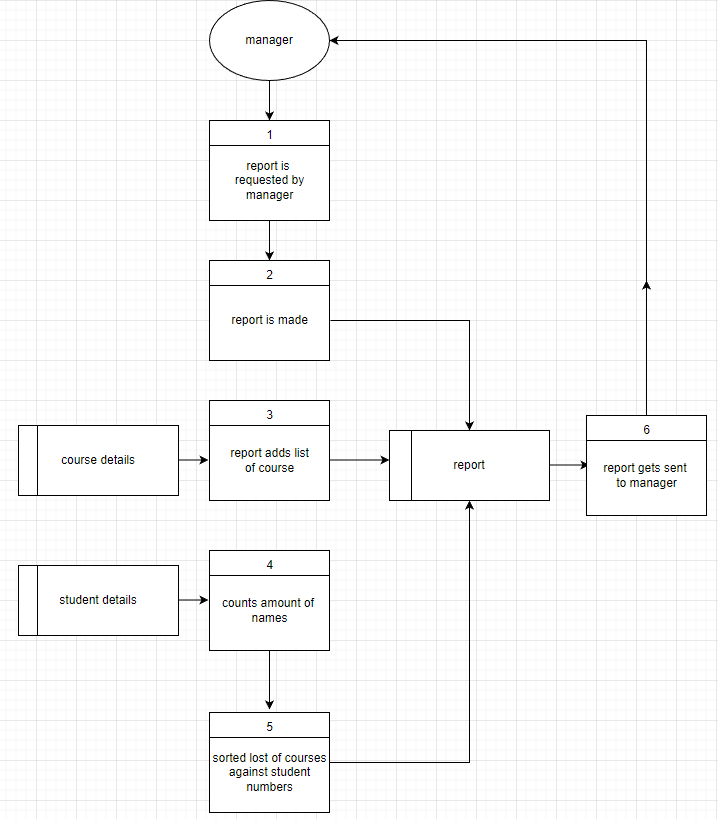
**Creating a trainer performance report**



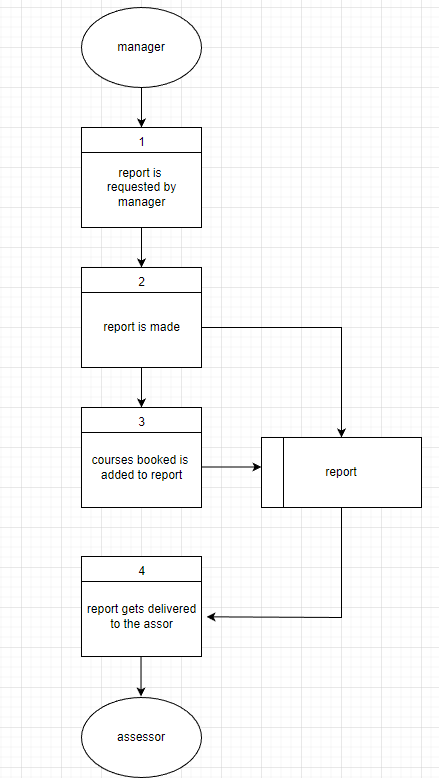
**Creating a course performance report**



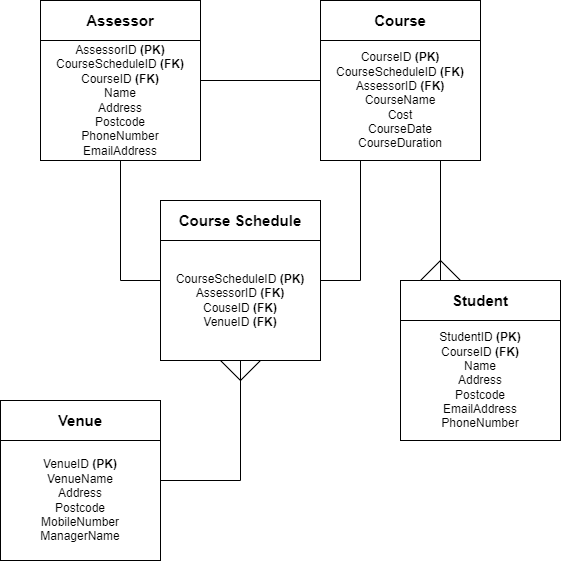
**Creating a course popularity report**



**Sending reports to assessors**



## **Entity Relationship Diagram**



## **Normalisation**

Normalisation is a technique which we use in relational databases to assist in reducing duplicated data and redundancy. This technique also simplifies in data integrity improved. By using normalisation techniques in your database, there are a few benefits such as:

* Less data redundancy
* Data is simplified
* Tables are shorter
* Searching and sorting is quicker
* Less empty values (null)

## **Un-normalised form (UNF)**

During the normalisation stages, UNF which can also be defined as 0NF is the first stage. After this from is created, first normal form (1NF) can then take place. Users who are experienced in working with forms in normalisation may tend to skip out this stage and start at 1nf but for those users who are new to database normalisation, this will be the first stage for them.

|  |
| --- |
| **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 |

**Database**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Student Name** | **Student**  **Phone**  **Number** | **Student**  **Address** | **Student Postcode** | **Student**  **Email** | **Achieved** | **Course**  **Name** | **Cost** | **Course**  **Date** | **AssessorName** |
| Jorry Hendo | 07948625438 | 74 Satin Street, | LP29,9DL | Jorry\_h19@outlook.com, | yes | Accounting | £500.49 | 21/04/22 | Fredrick Wilson |
| Calumn lord,  Steven Robert | 07481235694, 07512389427 | 153 cod Road  23 Station Road | KR2,5FA  ,  M25T,9AJ | C.L289@yahoo.com  St3vn92@gmail.com | No  ,  No | Accounting  ,  Sports | £380  ,  £600 | 21/04/22  ,  29/04/22 | Fredrick Wilson  ,  Holdan Rick |
| Jakob Linch | 07682047287 | 2 Farm Lane Road | O1L8,9AU | Jakob82674@gmail.com | yes | Sports | £600 | 29/04/22 | Holdan Rick |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **AssessorAddress** | **Assessor Phone** | **Assessor** | **Postcode** | **Assessor Email** | **Address** | **Venue Name** | **Venue** | **Postcode** | **Venue Address** | **Mobile Number** | **Manager Name** |
| 123 Main Street | 07123 456789 | John Smith | LDK4,1KD | [johnsmith@gmail.com](mailto:johnsmith@gmail.com) | 212, Pine Street | Crystal Oasis | Chester | CH20, 9AX | 73 Range Street | 07412 345678 | Jennifer Thomas |
| 789 Elm Street  ,  123 Maple Avenue | 07555 123456  ,  07123 456789 | Jane Doe  ,  Olivia bennet | M9N8O7  ,  P2Q3R4 | [Jane.doe@outlook.com](mailto:Jane.doe@outlook.com)  ,  o.bennet92@gmail.com | 53, Willow Street  ,  4, Oaked Road | Garden palm  ,  Charm lakes | Manchester  ,  Leeds | M18, AKW4  ,  LE38, 3IK | 94 commercial road  ,  1 Heaton Street | 07839205274  ,  07887 654321 | Christopher Martine  ,  Michael Brown |
| 456 Oak Street | 07777 999888 | Michael Lee | D4E5F6 | Mic.lee@outlook.com | 39 Ave drive | Heavenly hotel | Blackpool | BL29, 0FK | 5 robin lane | 07887 654321 | Andrew Miller |

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

During the first normal form stage, the main concept of this stage is to separate the database. By doing this, each value should be in one field. As the example above shows, some fields consist of more than one value. Below is a table I have designed to show this.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Student Name** | **Student**  **Phone**  **Number** | **Student**  **Address** | **Student Postcode** | **Student**  **Email** | **Achieved** | **Course**  **Name** | **Cost** | **Course**  **Date** | **AssessorName** |
| Jorry Hendo | 07948625438 | 74 Satin Street, | LP29,9DL | Jorry\_h19@outlook.com, | yes | Accounting | £500.49 | 21/04/22 | Fredrick Wilson |
| Calumn lord, | 07481235694 | 153 cod Road | KR2,5FA | C.L289@yahoo.com | No | Accounting | £380 | 21/04/22 | Fredrick Wilson |
| Steven Robert | 07512389427 | 23 Station Road | M25T,9AJ | St3vn92@gmail.com | No | Sports | £600 | 29/04/22 | Holdan Rick |
| Jakob Linch | 07682047287 | 2 Farm Lane Road | O1L8,9AU | Jakob82674@gmail.com | yes | Sports | £600 | 29/04/22 | Holdan Rick |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **AssessorAddress** | **Assessor Phone** | **Assessor** | **Postcode** | **Assessor Email** | **Address** | **Venue Name** | **Venue** | **Postcode** | **Venue Address** | **Mobile Number** | **Manager Name** |
| 123 Main Street | 07123 456789 | John Smith | LDK4,1KD | [johnsmith@gmail.com](mailto:johnsmith@gmail.com) | 212, Pine Street | Crystal Oasis | Chester | CH20, 9AX | 73 Range Street | 07412 345678 | Jennifer Thomas |
| 789 Elm Street | 07555 123456 | Jane Doe | M9N8O7 | [Jane.doe@outlook.com](mailto:Jane.doe@outlook.com) | 53, Willow Street | Garden palm | Manchester | M18, AKW4 | 94 commercial road | 07839205274 | Christopher Martine |
| 123 Maple Avenue | 07123 456789 | Olivia bennet | P2Q3R4 | o.bennet92@gmail.com | 4, Oaked Road | Charm lakes | Leeds | LE38, 3IK | 1 Heaton Street | 07887 654321 | Michael Brown |
| 456 Oak Street | 07777 999888 | Michael Lee | D4E5F6 | Mic.lee@outlook.com | 39 Ave drive | Heavenly hotel | Blackpool | BL29, 0FK | 5 robin lane | 07887 654321 | Andrew Miller |

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

After the 1nf, the 2nf begins implementing. As you can see the biggest change which we can see here is more columns.

|  |
| --- |
| **2nd Normal Form Table Design** |
| **Student** |
| * StudentID **(PK)** * Name * PhoneNumber * Address * Postcode * EmailAddress * Achieved |
| **Assessor** |
| * AssessorID **(PK)** * Name * Address * PhoneNumber * Postcode * EmailAddress |
| **Course** |
| * CourseID **(PK)** * AssessorID **(FK)** * CourseName * Cost * Duration * Date |
| **Venue** |
| * VenueID **(PK)** * VenueName * Postcode * Address * MobileNumber * ManagerName |

**Student table**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **StudentID** | **Name** | **PhoneNumber** | **Address** | **Postcode** | **EmailAddress** | **Achieved** | **CourseID** |
| C1224 | Jorry Hendo | 07948625438 | 74 Satin Street, | LP29,9DL | Jorry\_h19@outlook.com, | yes | M2929 |
| C5668 | Calumn lord | 07481235694 | 153 cod Road | KR2,5FA | C.L289@yahoo.com | No | M2932 |
| C9032 | Steven Robert | 07512389427 | 23 Station Road | M25T,9AJ | St3vn92@gmail.com | No | S0328 |
| C3486 | Jakob Linch | 07682047287 | 2 Farm Lane Road | O1L8,9AU | Jakob82674@gmail.com | yes | S0338 |

**Assessor table**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **AssessorID** | **Name** | **Address** | **PhoneNumber** | **Postcode** | **EmailAddress** | **CourseID** |
| C490 | Fredrick Wilson | 123 Main Street | 07123 456789 | LDK4,1KD | [johnsmith@gmail.com](mailto:johnsmith@gmail.com) | M2929 |
| C490 | Fredrick Wilson | 789 Elm Street | 07555 123456 | M9N8O7 | [Jane.doe@outlook.com](mailto:Jane.doe@outlook.com) | M2932 |
| C022 | Holdan Rick | 123 Maple Avenue | 07123 456789 | P2Q3R4 | o.bennet92@gmail.com | S0328 |
| C022 | Holdan Rick | 456 Oak Street | 07777 999888 | D4E5F6 | Mic.lee@outlook.com | S0338 |

**Course table**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **CourseID** | **CourseName** | **Cost** | **Duration** | **Date** | **AssessorID** |
| M2929 | Accounting | £500.49 | 2 | 21/04/22 | 403822 |
| M2932 | Accounting | £380 | 2 | 21/04/22 | 403831 |
| S0328 | Sports | £600 | 1 | 29/04/22 | 403847 |

**venue table**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **VenueID** | **VenueName** | **Postcode** | **Address** | **MobileNumber** | **Name** |
| VEN124 | Crystal Oasis | CH20, 9AX | 73 Range Street | 07412 345678 | Jennifer Thomas |
| VEN922 | Garden palm | M18, AKW4 | 94 commercial road | 07839205274 | Christopher Martine |

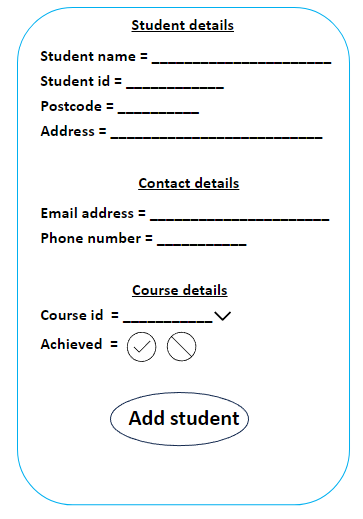
### **3rd Normal Form (3NF):**

|  |
| --- |
| **3rd Normal Form Table Design** |
| **Student** |
| * StudentID **(PK)** * Name * PhoneNumber * Address * Postcode * EmailAddress * Achieved |
| **Assessor** |
| * AssessorID **(PK)** * Name * Address * PhoneNumber * Postcode * EmailAddress |
| **Course** |
| * CourseID **(PK)** * CourseName * Cost * Duration * Date |
| **Venue** |
| * VenueID **(PK)** * VenueName * Postcode * Address * MobileNumber * ManagerName |
| **CourseSchedule** |
| * CourseScheduleID **(PK)** |

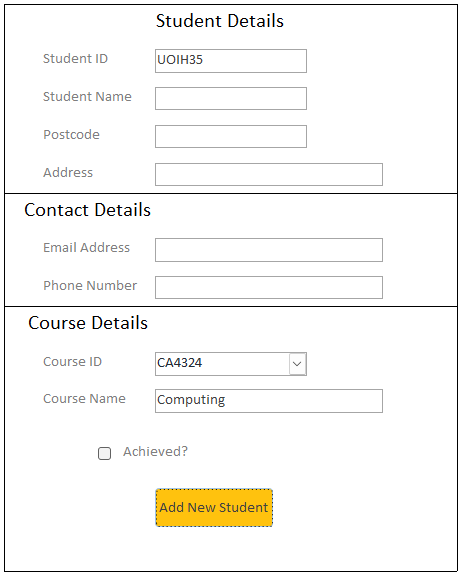
# **2.0 Interface Design**

## **Forms**

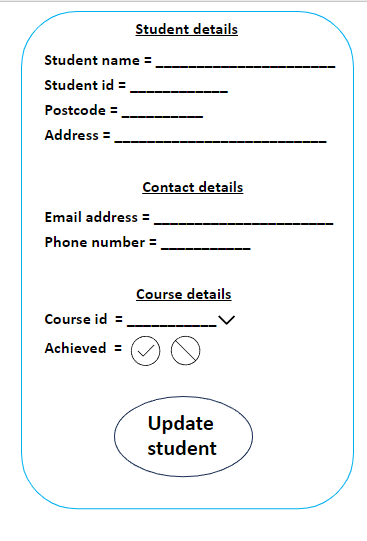
**Add student to course**



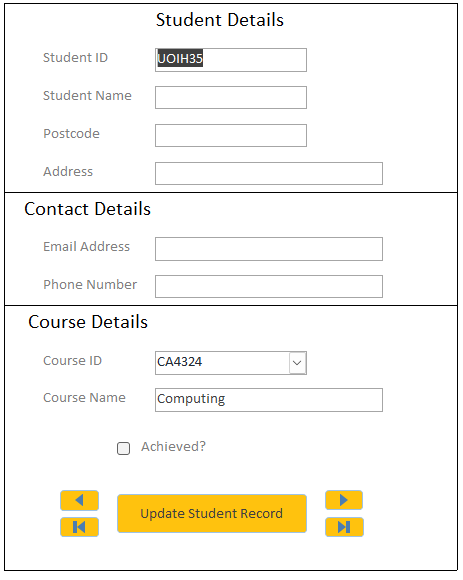
Above is a design which I created as a sketch to show how my final design may turn out. below is how the final design turned out on Microsoft Access.



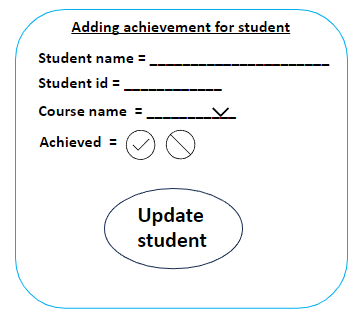
**Update student record**



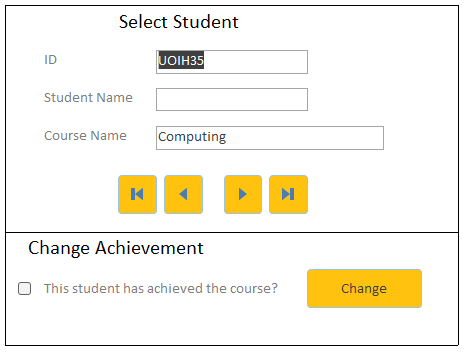
Above is a design which I created as a sketch to show how my final design may turn out. below is how the final design turned out on Microsoft Access.



**Add student achievement**

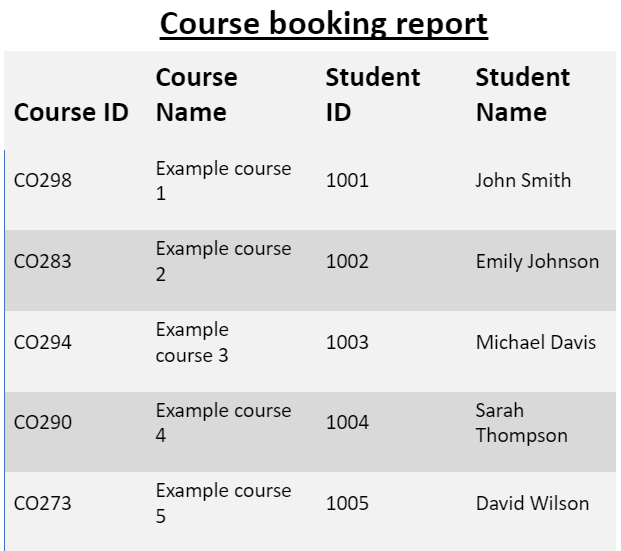


Above is a design which I created as a sketch to show how my final design may turn out. below is how the final design turned out on Microsoft Access.

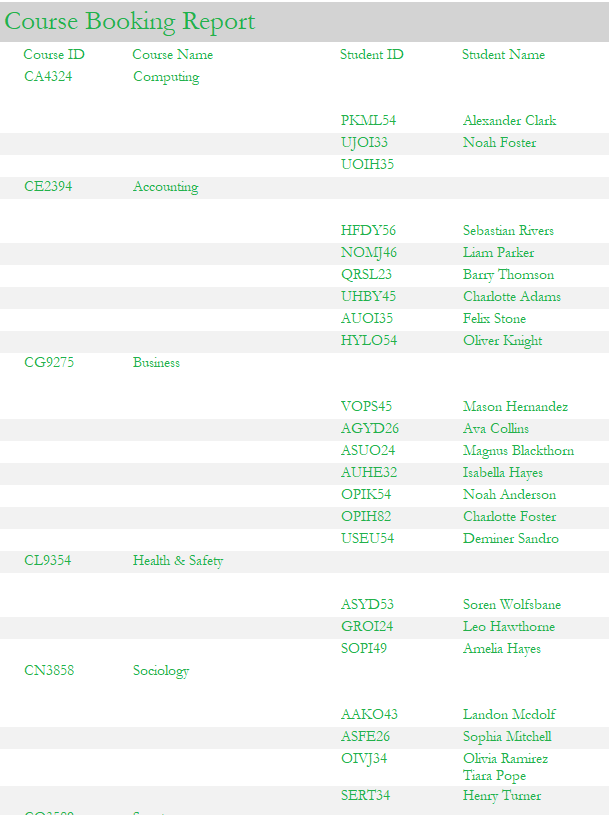


## **Reports**

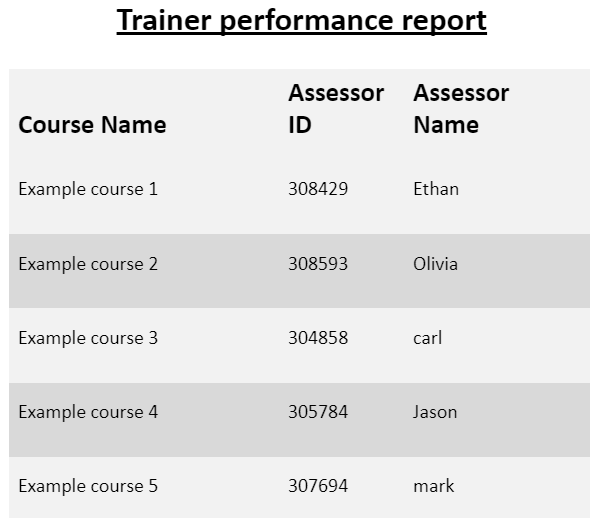
**Weekly Course Booking Report**



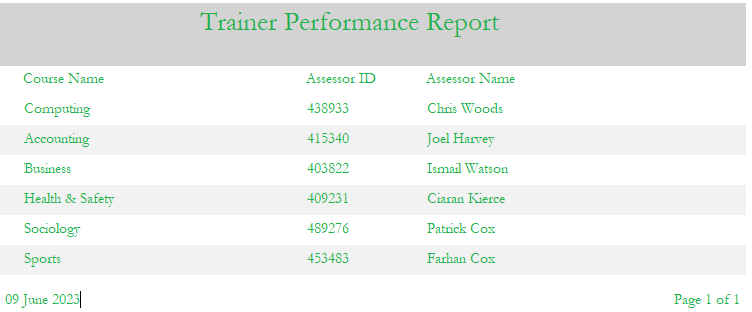
Above is a design which I created as a sketch to show how my final design may turn out. below is how the final design turned out on Microsoft Access.



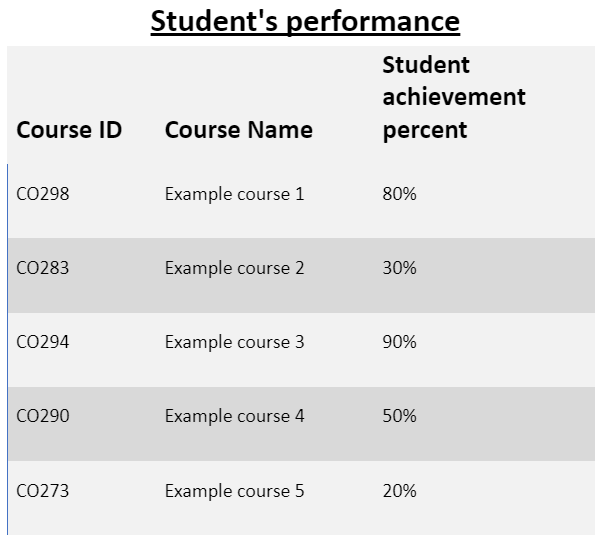
**Trainer Performance Report**



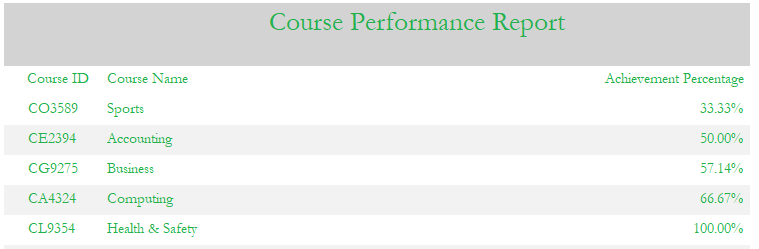
Above is a design which I created as a sketch to show how my final design may turn out. below is how the final design turned out on Microsoft Access.



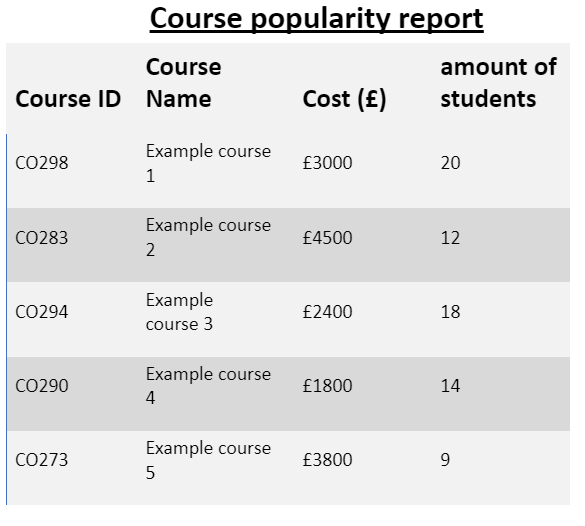
**Course Performance Report**



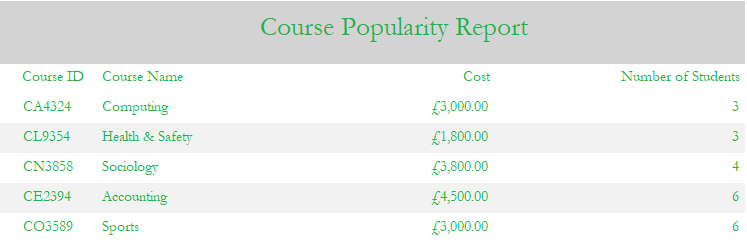
Above is a design which I created as a sketch to show how my final design may turn out. below is how the final design turned out on Microsoft Access.



**Course Popularity Report**



Above is a design which I created as a sketch to show how my final design may turn out. below is how the final design turned out on Microsoft Access.



## **Hardware and Software**

I am going to develop the database with the assistance of the software Microsoft access. Microsoft Access is a powerful graphical tool designed to run and manage databases without the need for manual coding. It offers a range of utilities that streamline database management, making certain tasks automatic and efficient. For instance, it provides features like primary/foreign keys, referencing, and referential integrity, which handle relationship management effortlessly. This means that I can focus on creating tables and defining their attributes and relationships, while the software Microsoft Access takes care of maintaining the connections between them. By automating various tasks and streamlining operations, Microsoft Access simplifies the overall process of managing databases. This results in considerable time savings, which allows me to work elsewhere.

A standard office or personal computer running Windows 10 or potentially Windows 11 with Microsoft Access is required as the database's hardware to be created and operated. Both Windows 10 and Microsoft Access have a list of necessary hardware. Microsoft Access requires a Dual Core 1.6 GHz processor, 4 GB of RAM, a graphics card that is DirectX 10 compliant, 4 GB of storage space, and Windows 10 or later to run. Like Linux, Windows just requires a 1GHz processor, 2GB of RAM, a graphics card that supports DirectX 9, and a 16GB hard drive.

These specifications will be followed by every device I use to build and test the database, with the slowest of them having a 6 core, 3.6 GHz processor, 16 GB of RAM, and a graphics card that supports DirectX 12. This is more than enough to create, test, and utilise the database because it meets Access's and Windows 10's basic needs.

## **Test Plan**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Plan** | | | | | |
| **Test No** | **Purpose** | **Test Data** | **Expected Outcome** | **Actual Outcome** | **Further Action** |
| **1** | Entering an invalid phone number or one that is too short will test the validity of the phone numbers in the student, assessor, and venue tables. | PhoneNumber: 07394723843 | An error message telling the user that the phone number entered was invalid should appear when the user tries to set the field as the number. The message box should show the following information:    An issue has happened, the given phone number is invalid. Please try again. | The database prevents the user from entering the value, and a message window stating:    An issue has happened, the given phone number is invalid. Please try again. |  |
| **2** | By entering a legitimate phone number, you can check that the student, assessor, and venue tables all have valid phone numbers. | PhoneNumber: 07928768324 | The field's current value must be changed to the new input. | The phone number field was changed to the new phone number: 07928768324 |  |
| **3** | Entering an invalid email address will check the validity of the email addresses in the assessor and student tables. | EmailAddress: including @ and. Co. Uk | The database should provide an error message that says the following to stop the user from entering the email:    "A problem occurred; the entered email address was invalid. Please try again!" | The database showed the following message:    The field was not converted to the supplied email address. |  |
| **4** | By entering a negative figure, you can check the validity of the cost and currency. | Course Cost: -£ 200 includes pound sign for business | If the cost is less than 0, the database should recognise this and stop the user from entering a value. The database will indicate the following error:    The price entered was invalid; all costs must be positive, hence there was a mistake. | The database showed the following message of an error:  The value was not changed. |  |
| **5** | By providing a valid positive figure, you can check the cost or currency's validity. | Course Cost has changed. | The value shown should be implemented to the table | Old value:    New Value: |  |
| **6** | All fields are correct and displayed when a new student is added, this is shown in the table. | New Student:  ID - QRSL23  Name - Barry Thomson  Postcode - ML24 KL3  Address - 82 Chester Lane  EmailAddress - B.thom92@gmail.co  PhoneNumber - 07423849232  CourseID - CE2394  Course - Accounting  Achieved - No/False | In the student table, a new row of data shall be inserted, and the table must be updated and show the new values. | 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 navigation for the updated student form record. | N/A | The form should scroll through the list of student records in the student table when the record navigation buttons are pressed. | Before:  Pressing the right navigation button:    After: |  |
| **8** | verifying that the fields on the update student form are updated for the right student. | Student to update: OPUI43  Data Changed: Address – 2 Landon Road  Postcode – Z45 0SP | When the update button is pushed, a notice should appear warning the user of the action that will be taken; if the user chooses to proceed, the information of the selected record should be altered. | Before:  After: |  |
| **9** | testing the record navigation for adding student achievement. | N/A | The form should scroll through the list of student records in the student table when the record navigation buttons are pressed. | Before:  Pressing the right navigation button: |  |
| **10** | The selected students' record is updated during the add student achievement form test. | Student to update: GROI24  Data Changed:  Achieved – Yes | If the user chooses to proceed after reading the notification that appears when the update button is hit, the chosen record's accomplished field should be changed. | Before:  After: |  |
| **11** | examining the course booking report to ensure that the information is accurate. | N/A | When you click the "course booking report" button on the report form, a freshly created report that lists every course in the courses table along with every student enrolled in that course should appear. | A new report is generated and displayed:    As you can see the new student Barry Thomson is added to the report |  |
| **12** | examining the course performance report to ensure that the information is accurate. | N/A | When you click the "course performance report" button on the report form, a freshly created report with a % column and the name and ID of each course should appear.    The percentage of students in each course who received a grade is shown in the percentage column for that course. | A new report is generated and displayed:    This is updated to real time and updated to when the most recent changes have been modified. |  |
| **13** | examining the course popularity report to ensure that the information is accurate. | N/A | Each course should be listed in the course popularity report along with how many students are enrolled in it. | Below shows a screenshot of the new updated report:    This is updated to real time and updated to when the most recent changes have been modified. |  |
| **14** | examining the trainer performance report to ensure that the information is accurate. | N/A | Each assessor should be listed in the assessors table of the report created by clicking the "trainer performance report" option on the report form, along with the course they are teaching. | Another report is generated and displayed:    This is updated to real time and updated to when the most recent changes have been modified. |  |
| **15** | Checking the functionality of each view report button on the report form. | N/A | A button should display the report it is related to right away when pressed. The buttons that read "View Course Booking," "View Trainer Performance," "View Course Performance," and "View Course Popularity" are all there on the website. | As observed in the earlier testing, pressing each button causes a print preview of each report to appear. |  |

## **Implementation Plan**

I will need to make an implementation strategy to lay out how I will build the database in accordance with the standards established by Workplace4Training to create it and make sure it is completely functional. In the beginning, I will design and plan the database's layout, including its overall structure, its tables' layouts, and its relationship structures. I will go through the normalisation procedure for the supplied set of attributes and arrange them in the proper table according to the client's needs after receiving the set of data on which to establish the database.

To finish this procedure, I will produce the necessary documentation, consisting of entity relationship diagrams, data dictionaries, and tables for normalisation, taking the database through each level of normalisation (0NF, 1NF, 2NF, and 3NF). With the help of this documentation, I will be able to fully plan out the database's layout and identify any flaws early on, when they will be much easier to fix than during the implementation stage.

Extra documentation, including data flow diagrams, will be produced for the algorithms required for the operation of specific database components, such as report production. I may specify the data that each report needs to retrieve and how it will be used by using data flow diagrams. The diagrams can be used as a guide when creating additional diagrams, such as layout designs, for the forms and user interface in the database once the database has been designed.

Following completion of the necessary documentation, the database implementation can begin, making use of all previous documentation. Prior to creating the database's logic/functionality through queries, tables, relationships, and forms will all be finished. To perform tasks like granting read and write access through forms and retrieving pertinent data for reports, queries will need to be constructed related to the flow diagrams made before.

## **Maintenance/Support Plan**

The database's documentation, including the ERDs, Data Dictionaries, DFDs, and general diagrams/documentation, will also be accompanied by a technical manual so that any necessary database maintenance can be done in the future. The technical manual and the documentation can be used by upcoming maintainers to keep the database up to date. The maintainers can consult the ERD or data dictionary to determine where the data needs to be put, for instance, if the database's scope needs to change and new tables or attributes are required.

The technical manual's goal is to make database modification for upcoming maintainers simpler. Furthermore, since later developers are going to able to refer to the documentation or guide to understand how the database works, faults and issues will be far less likely to arise. The guide can also explain the rationale behind every choice created within the database, including the rationale behind a query's purpose and the reasons why specific data was chosen for reports and forms, as well as the objective of each report and form.

## Project Timescale

From the customer defining the requirements through the final testing/deployment phase, the database will be finished in 8 weeks. What will occur during the next eight weeks is depicted in the table below.

|  |  |  |
| --- | --- | --- |
| **Project Timescale** | | |
| **Week** | **Summary of work** | **Description** |
| **1** | Project start and basic table design | The tables in the database will be the most vital component since they will determine how the database is constructed and what data is stored in all other components, including forms and queries. Since the client's requested raw list of characteristics and data is unsuitable for use as a database, work will start right away to normalise the given data into the appropriate tables.    A data dictionary and entity relationship diagram will be developed as documentation so that they may be utilised as references later in the project's development phase. |
| **2** | Basic Logic/Report design | The client-specified queries and reports will be planned and developed this week, and whatever information the customer has requested for each report to include will be incorporated to the report's structure. |
| **3** | UI (User Interface) and user interaction design/plan | The way users will interact with the database must be taken into consideration. Users will do so using forms provided as a GUI. Each database function, including adding, removing, and editing data, will be incorporated into a distinct form. Once the forms have been selected, a schematic or wireframe will be used to develop each form's interface. |
| **4** | Evaluation of designs | The database design phase has been finished; the last step is to go back over each stage of the design process to ensure that every component of the database architecture makes sense. Each component of the database, including the tables, queries, reports, and forms, is interconnected, and must be reviewed to make sure there are no errors or misconceptions. |
| **5** | Creation of database and implementation of tables | The entity relationship diagram and data dictionary will be used to add the tables and their relationships to the database using the tools supplied by Microsoft Access during the first week. This will be done utilising the documentation that was prepared in the weeks prior. |
| **6** | Implementation of logic/query design for reports and forms | All required queries and reports will be implemented into the database using the data flow diagrams and report designs previously prepared. All the procedures and information required for the reports and queries are specified in the data flow diagrams. These diagrams will be used to build the queries, which will then be connected to the reports. |
| **7** | Filling in of sample data for report/query testing | Data will be required to fill in each table in the database to test its functionality. At the beginning of this week, sample data will be generated and added to the database that satisfies certain requirements (such as a minimum number of students per course or the requirement that each record reference or be referenced by at least one other record).    The forms designed to interface with the queries and reports may then be used to test the inquiries and reports when they have been introduced. |
| **8** | Final testing, fixes/improvements, and rollout | After a week of testing, any issues that were found will need to be repaired. After the issues are resolved, more testing will be required to ensure that the change is effective and that it has not affected other portions of the database.    After completing all of this, the customer will be informed that the database is complete and will be given to them with all the sample testing data deleted and ready for active usage. The client's errors or suggestions will then need to be changed. |

**Technical Constraints of Hardware/Software:**

From the client defining the requirements through the final testing/deployment phase, the database will be finished in 8 weeks. What will occur during the next eight weeks is shown in the table below.

I will be utilising Microsoft Access for the database because it offers a very straightforward and user-friendly interface for creating, managing, interacting with, and using databases. Microsoft Access must be installed on the user's device for them to utilise the database. While this is not a problem for Windows-based computers, it prevents Linux and MacOS devices as well as mobile devices from using the database.

Unlike to most other Office/Microsoft products, Microsoft Access is only compatible with versions 10 and 11, therefore users of Android, MacOS, Linux, or earlier versions of Windows will only have access to obsolete versions.

The client will be forced to use Windows on all their machines due to Microsoft Access's limitations. Given that some of the client's gadgets might not work with Access, this could be a problem. If the client owns hardware with a MacOS or Linux operating system, the hardware and operating system may need to be changed if the client wants to use the database.

## **Email to client**

|  |  |
| --- | --- |
| Rahat, the customer, will receive an email from me outlining the database's plan and all the elements that will affect it after it is implemented, such as technological limitations and user interaction. | |
| **Email** | |
| **To:** | Rahat-00@gmail.com (Client) |
| **From:** | Mahin210@outlook.com (Manager) |
| **Subject:** | Workplace4training database timelines, proposal, and plan |
| Dear Rahat,  The database's design proposal has been finished, and Microsoft Access will be used to create the database first. The database will be accessible to you and every other employee thanks to Microsoft Access. In addition to having fewer bugs and implementation issues than a manual implementation, Microsoft Access was chosen because of its many advanced features and functions that completely automate many aspects of database functionality. Access has a variety of functionality, including built-in Micro GUIs known as Forms for which database functionality (such as adding, deleting, and modifying data) can be done.  However, as Microsoft Access is only compatible with these versions of Windows, all devices and users who need to access the database must be running Windows 10 or 11 because the database cannot run on other operating systems like Linux, MacOS, or Android, thought must be given to the devices that you, your management, or your staff will use to access the database.  The database is expected to be finished in 8 weeks, during which time there will be revisions to the initial concept and planning, development/implementation, testing, and input from you. To maximise the likelihood that the database will be finished within those eight weeks, a test plan has been developed with each step or stage receiving the proper time needed to be completed.  The database will include 5 tables total, 5 for students, 5 for assessors, 5 for courses, 5 for schedules, and 5 for locations. The proper data has been found and put to the design for the database table structure for each table's characteristics. Forms will enable users to easily interact with the database through a GUI overlay over the raw tables and queries. Users will interface with the database through the forms. The reports that are supposed to be produced weekly have also been planned; they will include course booking, course performance, course popularity, and trainer performance reports, all of which were mentioned in the database's specifications.  We will respond and consider any ideas or improvements you may have if you feel the need to make them.  your sincere, Mahin | |

## Meeting notes

The project manager, client, and developers met to discuss the database design as well as other aspects of the database, including the user interface, screen design, and reports. The main points from the meeting are as follows:

* To complete the activity of the form, all the current forms will require buttons. Each form is assigned to a certain database interaction. As a result, the user will require a mechanism to launch the specific action for that the application was designed. This implies that each form must include at least one button.
* All user-facing interface components, which include documents and forms, must be simple to use and readily available for users, which calls for labels, textboxes, and buttons to be noticeable.
* The interface and forms are meant to make it simple and practical for users to access the database, so as little work as possible should be required of them. This implies that, for instance, the remaining information or fields from a record that contains the primary key will be filled in automatically as you cycle through the collection of recorded primary keys in a table of data.

Following additional examination of these notes, the following enhancements and modifications have been found:

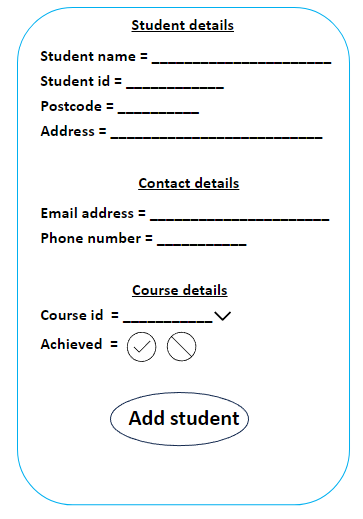
* Each form and report design has been updated with any missing titles.
* The forms now have colour, and the interface design has been divided up with more spacing.
* Users can now click links to scroll through a table's list of records thanks to the addition of record navigation capability to forms that update or alter data in tables.

## **Review of Designs**

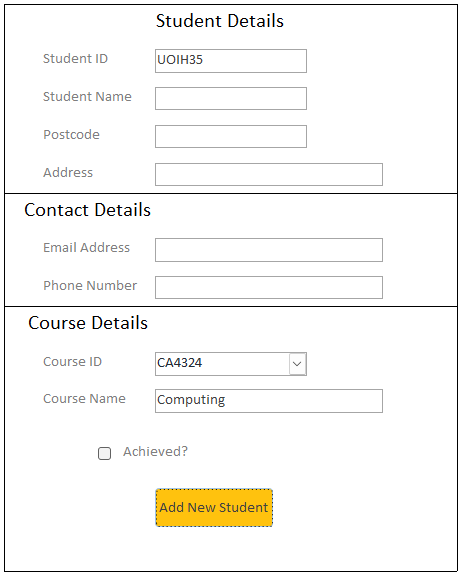
The meeting notes section's information from the meeting, together with my own analysis, have been thoroughly reviewed. As a result, some minor changes will be made to the interface's design. The database's user experience will be enhanced by these changes. Even though these are not significant changes, they will aid in the database's long-term use by reducing user confusion and error.

## Updating of designs

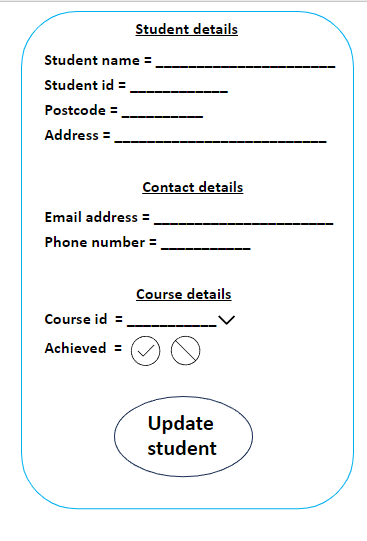
**Add student to course**



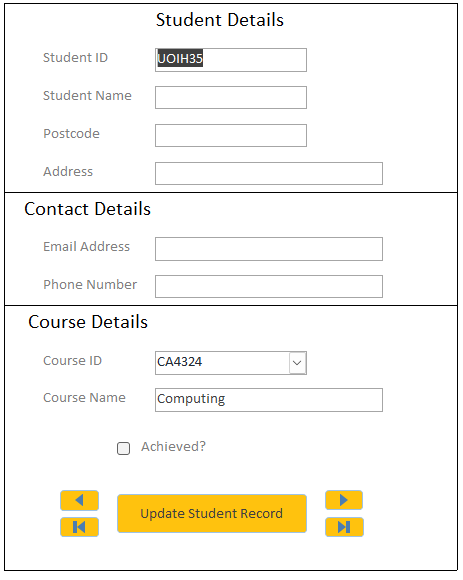
Above is a design which I created as a sketch to show how my final design may turn out. below is how the final design turned out on Microsoft Access.



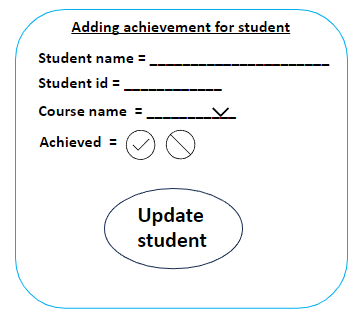
**Update student record**



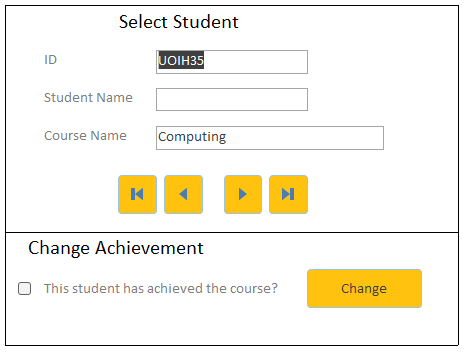
Above is a design which I created as a sketch to show how my final design may turn out. below is how the final design turned out on Microsoft Access.



**Add student achievement**

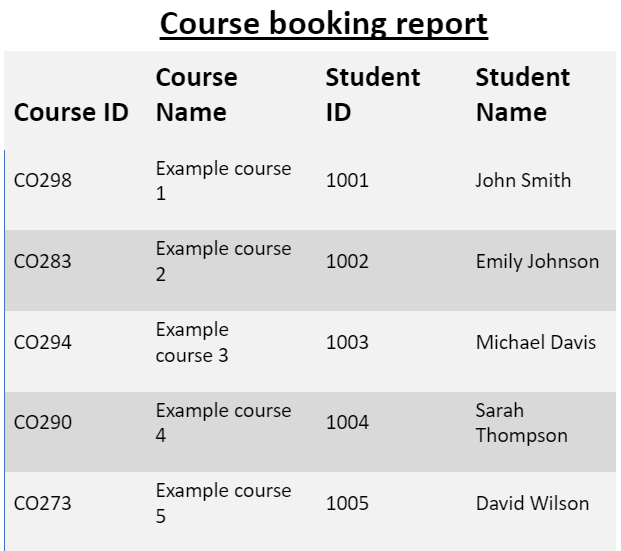


Above is a design which I created as a sketch to show how my final design may turn out. below is how the final design turned out on Microsoft Access.

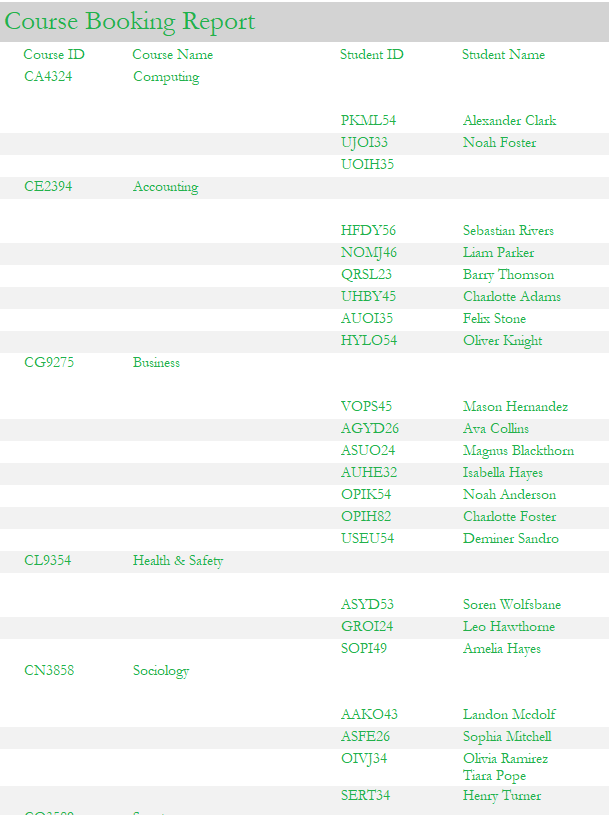


## **Reports**

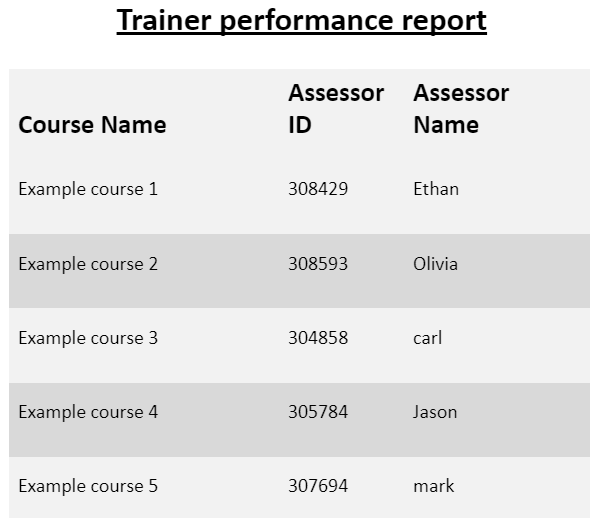
**Weekly Course Booking Report**



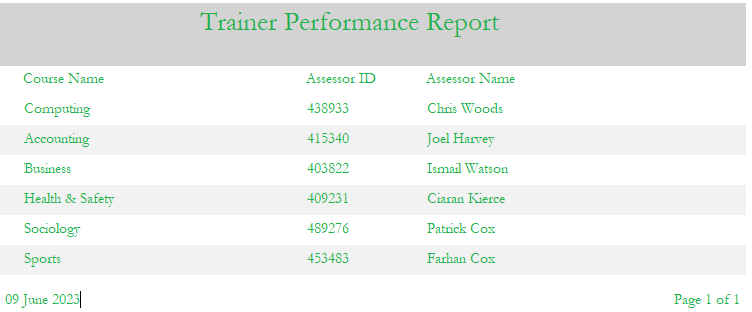
Above is a design which I created as a sketch to show how my final design may turn out. below is how the final design turned out on Microsoft Access.



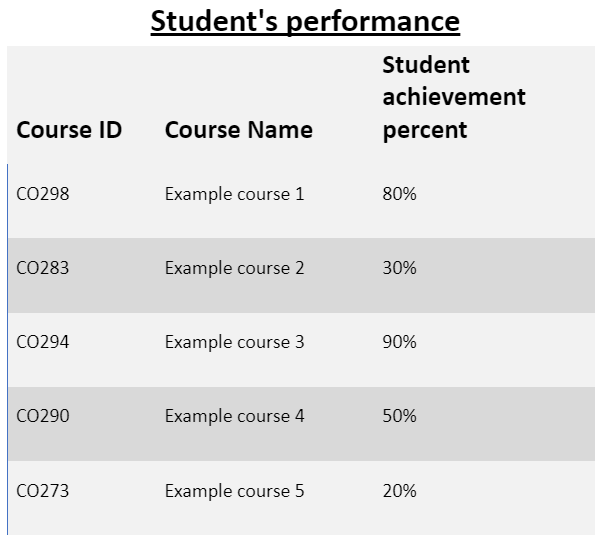
**Trainer Performance Report**



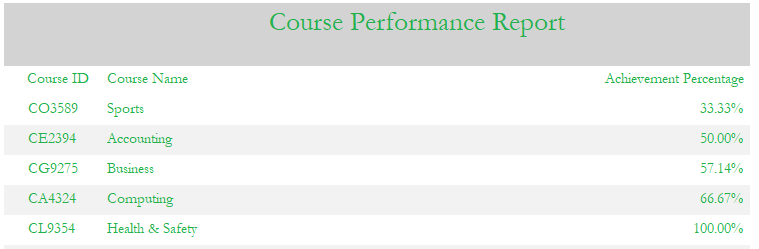
Above is a design which I created as a sketch to show how my final design may turn out. below is how the final design turned out on Microsoft Access.



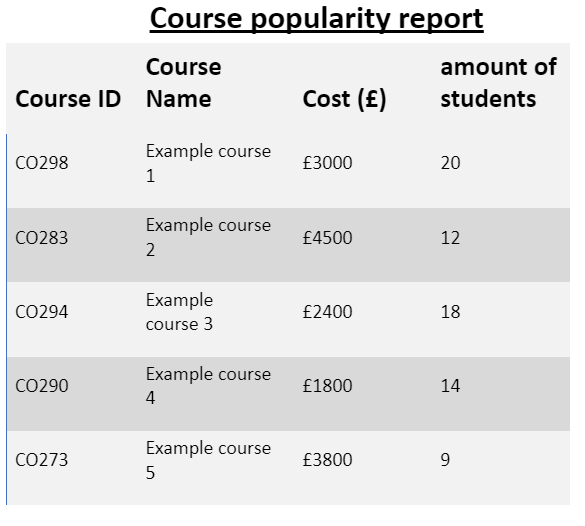
**Course Performance Report**



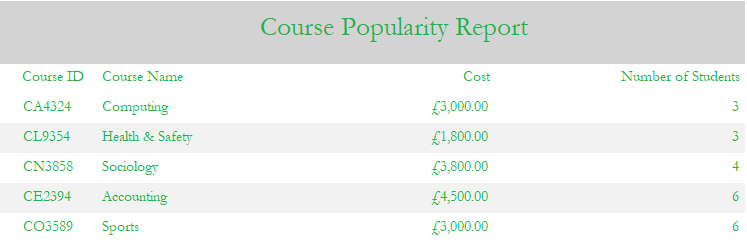
Above is a design which I created as a sketch to show how my final design may turn out. below is how the final design turned out on Microsoft Access.



**Course Popularity Report**



Above is a design which I created as a sketch to show how my final design may turn out. below is how the final design turned out on Microsoft Access.



**Evaluation of designs**

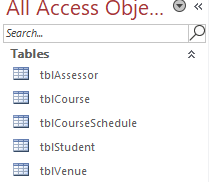
In general, the designs are acceptable for the customer and will be utilised for the user interface after rigorous design and analysis helped to produce the initial layout based on the client's needs and after revisions were made after speaking with the client. The layout offers a simple user interface with clear labelling and the ability to distinguish different portions of the interface using tools like colour. Using the color green stands out and is not too hard to read. Building the layout as a table result in making the forms and reports look neater and easier to interpret.

After looking through the initial ideas and having an interview with the administrator, I decided to revise the designs. The modified designs may be seen below. The modifications will mostly be visual for clarity and simple identification of interface elements like marking, this will boost the user experience.

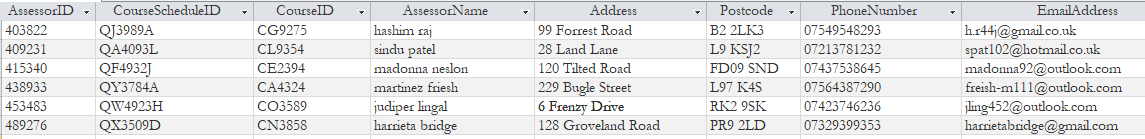
# **3.0 Implementation of database**

## **Evidence of tables**

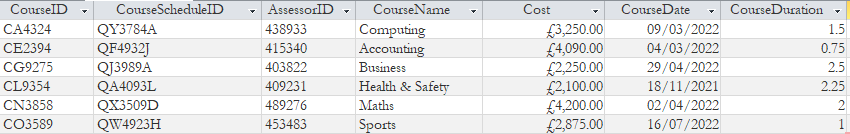
On Microsoft Access, all tables are displayed in the spreadsheet view; the attributes of each table are displayed in the yellow bar at the very beginning of the image that appears. Each image contains a vertical list of the data entries for each table. In each table, the far-left column contains the main keys, and the next-to-right column contains the foreign keys



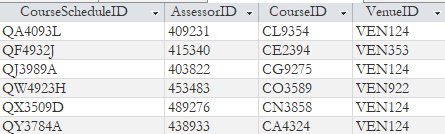
**Assessors table**



**Course table**



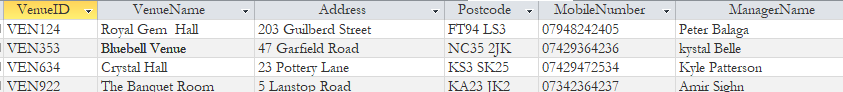
**Course Schedule table**



**Student details table**

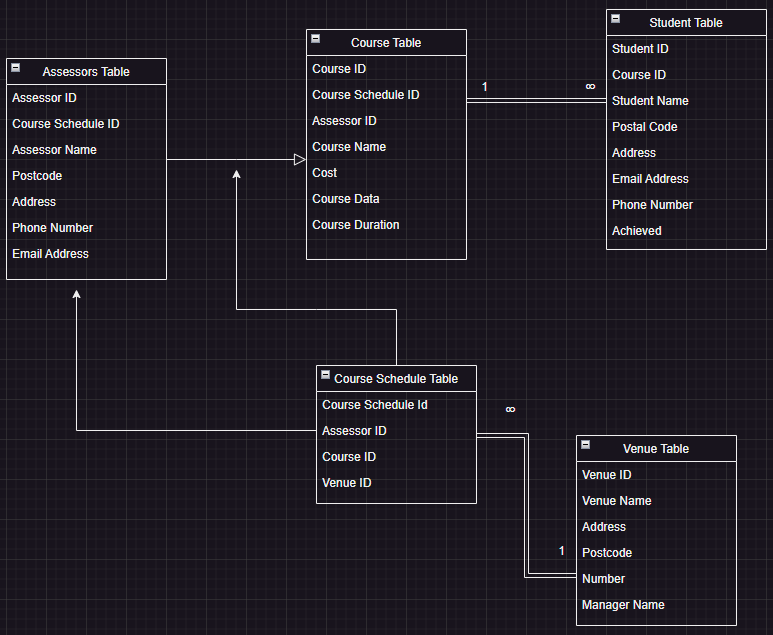


**Venue table**



## **Evidence of relationships**

we can see, the relationship tab in Microsoft Access displays identical connections as the diagram of entity relationships that was previously constructed. Each line depicts the connection between each table, and the symbols at each line's ends indicate the nature of the relationship. For example, in a one-to-many relationship, the symbol "1" denotes the "one" end of the relationship, and the infinity symbol "" denotes the "many" end. One-to-one connections are shown as a line with two small dots on each end.



## **Evidence of validation**

When a new value is entered, it must meet the criteria specified in the validation rule; if it does, the result will be changed; if it does not, a message of error will be displayed, which will display 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 images**

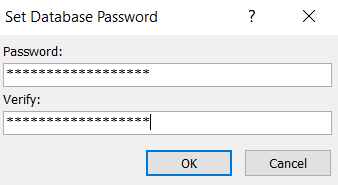
|  |  |
| --- | --- |
| **Forms** | |
| **Add Student Form** | |
| **Design View** | **Form View** |
|  |  |
| **Update Student Form** | |
| **Design View** | **Form View** |
|  |  |
| **Add Student Achievement Form** | |
| **Design View** | **Form View** |
|  |  |
| **View Report Form** | |
| **Design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

|  |  |
| --- | --- |
| **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** |
|  |  |

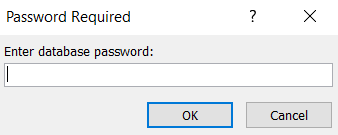
## Password protected database

I will secure the database file using a username and password that will be given to the customer to protect it from threats and breaches. By doing this, I can make sure that even if malevolent hackers gain possession of the information document, the data inside will remain secure.

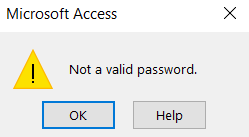
Following is an example of how I established the database password using a feature offered by Microsoft Access:



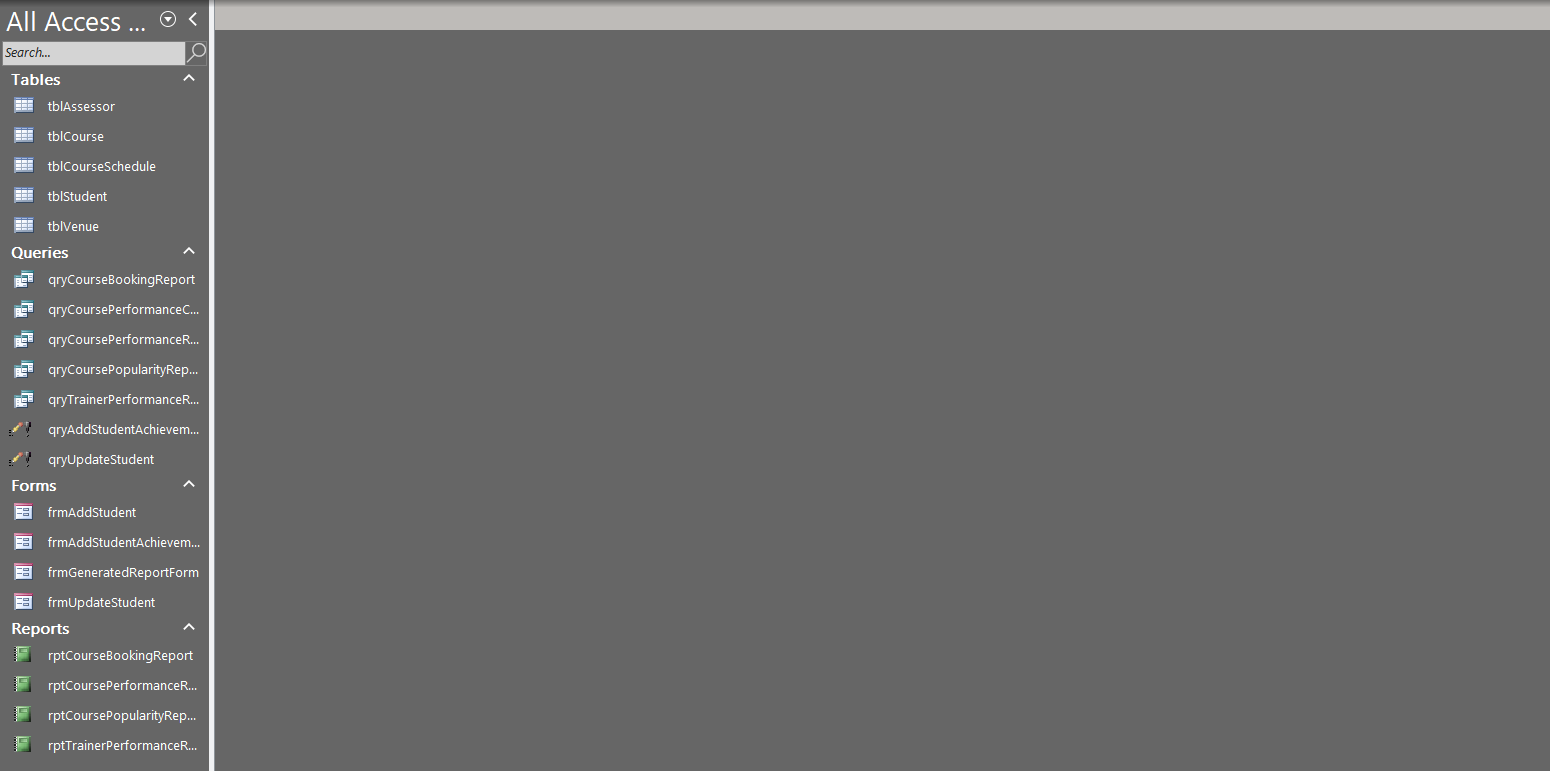
"Workplace4Password" serves as the database's password; it is a simple example password that the customer may modify later. Setting the database's password in this way causes an entry box to appear requiring the client to enter a username and password every time an attempt is made to open the database, as shown below:



This will be displayed if entering an incorrect password is applied:



And after entering the correct password which is “Workplace4Password”:



The database is entirely accessible to the user, just as if it were not encrypted.

# **4.0 Final evaluation of database**

## **User and client requirements:**

The client's requirement met in the overall database which I have completed, specified by the client that the database is to store data for Workplace4Training; data such as students, courses, assessors, and venues. The database is structured in the form of a table, which allows for the appropriate relationships between each record in each table, this indicates that the database is suitable for its role in storing and managing students and the courses they are on.

Furthermore, the clients' requirements carry out all the set-out functionality in the database, operations such as modifying a student's record either to update some fields within the record and to add the student achievement; the database preforms. For managers to be able to access reports, the database has had reporting functionality built into it, creating dedicated form for the managers. The database consists of 4 main reports, each collects data from the appropriate tables.

The 4 reports consist of the course booking, course popularity, course performance and trainer performance reports. Whatever data needs to be seen regarding the database can be seen by the manager using the 4 reports listed above.

## Quality of the relational database:

Designed from scratch from the old system, the database uses a list specified by the client about what the database should store. Using the list provided by the client, which must be available in the database, the database table structure and the resulting relationships have been planned and documented. Normalisation, which provides a fully functioning and error free database, is a process the list of data has gone through.

When the appropriate relationships between each table can be setup the this is the use of normalisation, means that many of the processes to viewing and modifying data, meaning using relationships in the database, results in the database operating fully automatically. Records in the database can be linked together, which is seen to be an advantage.

An advantage of referential integrity and cascading updates can be seen due to the use of normalisation and relationships in the database. The database must only update the data in that record once when a record requires to have its data changed. Reducing repeated data is an aim of normalisation as if data is repeated multiple times within a database, the data can be mismatched. Due to the facts that if the change is missed for that field/data in one or more records, the data mismatch is prone to errors, which can result in confusion, frustration and more severely, lost data.

With normalisation, any record that requires access to recurring data will be provided with a link to another table that includes the necessary data. For instance, if the venue for a course needs to be adjusted, only the location record needs to be updated rather than updating the venue for each student if they each have a reference ID related to a course if the instructional material contains a description of a venue.

Additionally, the database offers a user-friendly and convenient way for users to communicate with it; they are not required to physically connect with it through Access to carry out their desired action. To create an additional, simpler layer of interaction, I have set up forms, which are miniature GUIs built into the database that link to the database functionality. There is a form designed specifically to handle inserting new pupils to the database rather than, for instance, entering a new student straight to the table, which may be perplexing to certain users.

## **Fitness for purpose:**

The original databases, which frustrated assessors, users, and managers, were to be replaced as part of this project. This project must provide a database that satisfies all requirements for functionality and usability before Workplace4Training may integrate it into their business for use. The created database satisfies all these needs by offering an improved relational database that resolves the issues of the old system that were raised in the past and adds additional capabilities like reports and forms.

## **Suitability against the original requirements:**

The client's original specifications for the database were for a relational database that holds information on students, examiners, courses, locations, and class schedules. Furthermore, the requirements include that the database must perform particular and unique tasks, such as producing reports and offering a user-friendly way to interact with forms.

The database is necessary for Workplace4Training to hold the data required for the operation of the company. Many of the issues that managers and assessors raised when attempting to use the previous system/database are resolved by the new database. The introduction of the unfamiliar relational database has reduced some of these worries, including the need to manually update every entry throughout multiple databases when a single piece of data changed. Relational databases are ideal for these kinds of requirements, and their design has already eliminated this worry.

## **Legal and ethical constraints:**

Because a company will manage the database, that company (Workplace4Training) will be governed by several data protection and privacy laws, including the Data Protection Act of 19984 The regulations set forth further principles for data protection as well as recommendations for what businesses and organisations may and may not do with consumer data. As the information in the database cannot be utilised if it is not compatible, these regulations must be obeyed.

Although users and customers now have greater access rights to their information, the Data Protection Act also require organisations to guarantee that data confidentially is maintained with relation to all user and customer data. This is where there may be a problem. This means that businesses must take all necessary precautions to prevent data leaks and breaches; otherwise, they risk fines of up to 4% of their annual revenue. If a data breach is severe enough or occurs frequently enough, more harsher penalties may be applied, such as a complete restriction on the storing of any customer or user data. While this is not ideal, it will render the database useless.

Database administrators and the company must take precautions to prevent data leaks, whether unintentional or on purpose, to prevent security breaches. This can be accomplished by using techniques like encryption and password-protecting the database. A password has been put up for the database so that anyone attempting to access it without it (with nefarious intent) will be prevented from doing so. The database password is also the encryption key, thus even if hackers were to get access to or steal the database file, the database would still be secure because it is encrypted.

## **Technology constraints:**

Microsoft Access is the technology used to develop the database. Access was selected for the database because to its advanced functions, including reports, forms, and query builders, as well as its ability to handle numerous backend tasks including referential integrity. Unlike other database programmes like SQLite or MySQL, Access is designed to be used by small organisations, making it the best choice for this application. However, because Access has faults and problems that could harm the database during development, testing, or real use, developing on Access has its own set of challenges.

Access has a minimal list of system requirements as well. It is a large application that conducts numerous complex activities and procedures; therefore, it will need a capable system. Even if a machine satisfies the minimal system requirements given by Microsoft for Access and the database as a result, Access may still be restricted by low hardware as the computer may still be slow or corrupt.

As a result, anyone using a device for accessing the database will need to make sure it complies with the minimal system requirements listed on Microsoft's website.

## **Strengths and alternative solutions that could be implemented:**

The benefit of utilising Microsoft Access is that it makes it quick and simple to establish new databases. Access will handle most of the work while the various database segments and components, such as user interfaces, are being developed. For instance, Access features an integrated SQL engine that can be applied to manually build queries or, alternatively, to automatically generate queries by entering in the data that the search needs to get.

Access features support for GUI/form micro programmes inside the database that may be configured to communicate with the database, and it also offers wonderful tools for interacting with the database. For any purpose, including data viewing, report creation and viewing, table record modification, record deletion, record addition, and many other operations, forms can be built.

Access, however, has compatibility issues as well as faults and failures. Only Windows 10 and Windows 11 are compatible with the most recent and most supported versions of Access (currently Access 2019), which means that users have a limited number of options for operating systems if they frequently access databases. Other database management systems, including MySQL and SQLite, are not affected by this problem because they were designed to work with many operating systems, such MacOS, Linux, and Android.

## **Platforms and compatibility:**

Since the database was implemented using Access, it can only be used with software and operating systems that are compatible with Access, which means that some platforms might not be supported. Microsoft has not made any attempts to migrate It to other platforms like MacOS and Linux since Access is a large, complicated application; it is accessible only on Windows 10 and 11.

As a result, customers who need to access the database on other operating systems, such as MacOS, Android, Linux, or earlier versions of Windows, will not be able to do so. To use a database developed in Microsoft Access, Access must also be used to open the database. Access is set up not only as a database creation software, but also as a management and widespread use software for Access built databases.

# **5.0 Evaluate your skills**

## **Your time management and planning:**

Over this assignment, I have used the information given to me by planning effectively and efficiently arranging to oversee the time I spent throughout the entire assignment. To successfully set up the database, while also fulfilling the client's goals and demands, I set up in advance for what the database would look like ahead of time and how I intend to do it. In the early stagiest of the assignment, I estimated the duration for how long the assignment will take, according to the many tasks I must complete for the assignment. I then developed a term table over the next 8 weeks (2 months), containing the tasks and duties that will need to be done at every stage of the 8 weeks (2 months).

By developing a timescale, this has enabled me to offer myself the greatest possible chance of concluding the assignment on schedule, establishing which areas of the overall assignment must be completed and when. By dividing each component of the assignment including; design phase, implementation phase and testing/developing phase, enables me to precisely approximate the duration for each component phase and the various parts of the component phase will take place. This has enabled me to evaluate my assignment as I devoted some time over the assignment term table to analyse what has taken place so far, allowing me to enhance it even further.

I considered and created the following documents, including a test plan, data dictionaries, entity relationship diagrams, data flow diagrams, and interface design wireframes, to build and construct this database. As I was able to use the planning documents as a guide during the implementation phase, producing these documents at the start of the assignment supported and enabled me. This showed that I was able to implement the database more effectively to satisfy the client's needs, which they had specified, as they had approved 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.

Following this, a meeting commenced between the whole team, client, and me to discuss and talk over the proposal of the database. In the meeting, issues including user interface were discussed with the client, recommending, and enhancing whether they would like to observe or add onto the ongoing strategy. Considering these guidelines was critical in strengthening the design of the user interface. Once the meeting ended, the recommendations made within the discussion were assessed and a plan of action for the user interface were modified to satisfy the requirements of the client.

These concepts are recommended from the overall suggestions, such as what they would prefer to observe and witness, within the pattern of general specific improvements after presenting them with a variation of plans and ideas. The plans and ideas were then overruled and altered to accommodate the clients' recommendations and were incorporated into the documents.

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

Throughout the duration of the assignment, I repeatedly lead the assignment suitable and in a timely productive manner. As a result, an outstanding database was created. As the on-going assignment commenced, I considered the team and client by repeatedly interacting with them by sending informal emails regarding what is taking place in the meetings. I continuously re-opened the assignment for disapproval, so that if the client or team have any further suggestions, they could inform and speak to me regarding it. On hearing any suggestions, I would evaluate them whiles modifying the assignment components or for further component of the assignment.

I remained in professionalism across the assignment by the constant contact and communication with the client. The client would be aware of all aspects of the assignment and how it is coming along. I kept the client up to date with all details at the very beginning and throughout the entirety of the assignment, by allowing the client to know and be informed of essential information for instance; how long the project will commence? Or how the term table for various component of the assignment, and how the assignment was scheduled to be applied into operation.

I also implemented and took into consideration my teams thought through my preparation and reporting. But additionally for the clients and team, but particularly for the possible maintainers of the database. I have composed an instructional manual with important information about the database and all the organisation and preparation for the user interface wireframes, diagrams and data dictionaries, as an outcome of which indicates whatever for the team or for possible maintainers to implement and understand a certain aspect of the database design and implementation they can examine through the accompanying materials about the assistant support guide.

## **Your recommendations and decisions:**

While putting together an arrangement, I had developed and created a proposal to the client, regarding the design and application for the database. The proposal consisted of many important and valuable details about the database such as; what platform the database would be implemented in? And many more. I put forward an observation about the platform, considering and noticing advantages and drawbacks of the suggested platform, Microsoft Access. I identified the advantages and disadvantages, which would impact and influence the client, so that they are able to thoroughly understand and arrange.

Contained within the proposal, I also offered and submitted a suggested time to accomplish the assignment. The assignment term table of which was 8 weeks. In addition to it all, I also proposed the completion of the concept and layout of the database, analysing and examining what data the client had requested, for the database to hold. The proposal to the client further opened and allowed us to get any recommendations from the client if they encountered any form of issue or worry with my proposal.

## **Targets to obtain insights into own performance:**

During the assignment, I required to contemplate about my individual and personal achievement and accomplishment while controlling and overseeing the leading direction of the assignment. During the assignment took place, I specified and provided the assignment goals and objectives, which would provide me with a comprehensive understanding into how I preform. Some of these goals and objectives were for instance; milestones from minor ones like preceding a particular section of the assignment term table to major ones for example as finalising and concluding the assignment on schedule, while offering and providing a flawless functional database.

There are several goals and objectives for this assignment which if fulfilled or not, could decide on my overall performance on this assignment. Failing in a handful of occasions or a substantial number of times of these goals and objectives will have a high change of leading my overall performance to be poorly acted of during this assignment. However, since the assignment so far has been effective in offering an entirely functional and operating database, with all the requirements and recommendations met of the client’s, leads to an overall conclusion that I have performed successfully and effectively. Moreover, I can use the knowledge and understanding obtained within these goals and objectives to assess and identify which aspects of my accomplishments demand improvement.