|  |
| --- |
| Name: S.K Udara |
| Student Reference Number: 10749130 |



|  |  |  |
| --- | --- | --- |
| Module Code: PUSL2019 | Module Name: Information Management & Retrieval | |
| Coursework Title:  Student Management System | | |
| Deadline Date: 24nd Dec 2021 | | Member of staff responsible for coursework:  Kevin Samarasinghe |
| Programme: Bsc (Hons) Software Engineering | | |
| Please note that University Academic Regulations are available under Rules and Regulations on the University website [www.plymouth.ac.uk/studenthandbook](http://www.plymouth.ac.uk/studenthandbook). | | |
| Group work: please list all names of all participants formally associated with this work and state whether the work was undertaken alone or as part of a team. Please note you may be required to identify individual responsibility for component parts.   * Project and Group Leader – S K Udara (10749130) * Planning Leader – K K D B Gimhana (10749170) * Technical Leader – M M S P Marasinghe (10750062) * Programming Leader – R S Rashmika (10749135) * Quality Leader – W M M Fernando (10749140) * Testing and Maintenance Leader – W P Y U Perera (10749131)   ***We confirm that we have read and understood the Plymouth University regulations relating to Assessment Offences and that we are aware of the possible penalties for any breach of these regulations. We confirm that this is the independent work of the group.***  Signed on behalf of the group: A picture containing text  Description automatically generated | | |
| Individual assignment: ***I confirm that I have read and understood the Plymouth University regulations relating to Assessment Offences and that I am aware of the possible penalties for any breach of these regulations. I confirm that this is my own independent work.***  Signed : A picture containing text  Description automatically generated | | |
| Use of translation software: failure to declare that translation software or a similar writing aid has been used will be treated as an assessment offence.  I \*have used/not used translation software.  If used, please state name of software………………………………………………………………… | | |
| **Overall mark \_\_\_\_\_% Assessors Initials \_\_\_\_\_\_ Date\_\_\_\_\_\_\_\_\_** | | |

**Table of Contents**

Table of Contents..........................................................................................................2

SECTION 01 ................................................................................................................3

Introduction ...............................................................................................................3

Extended Entity Relationship Diagram........................................................................4

Diagram ....................................................................................................................4

Assumptions .............................................................................................................5

Relational Mapping .....................................................................................................6

Mapping of Strong Entities.........................................................................................6

Mapping of Weak Entities .........................................................................................6

Mapping of Binary 1:1 Relationships ........................................................................7

Mapping of Binary 1:M Relationships ...................................................................... 7

Mapping of Binary M:N Relationships.......................................................................8

Mapping of Multi-Valued Attributes...........................................................................9

Mapping of N-ary Relationships...............................................................................10

Mapping of Recursive Relationships........................................................................10

Mapping of Extended-Entity Relationships .............................................................11

Normalization.............................................................................................................12

Dependencies......................................................................................................... 12

1st Normal Form (1NF)............................................................................................13

2nd Normal Form (2NF)...........................................................................................13

3rd Normal Form (3NF)............................................................................................13

Data Dictionary ........................................................................................................14

SECTION 02 ................................................................................................................17

Create Table Statements ...........................................................................................17

Database Diagram .....................................................................................................21

Database Sample Records ………………………………………………………………..22

SECTION 03…………………………………………………………………………………..28

Create Trigger Statements…………………………………………………………………28

Create Function Statements……………………………………………………………….29

Create View Statements……………………………………………………………………29

Create Procedure Statements……………………………………………………………..30

SECTION 04…………………………………………………………………………………..31

Critical Evaluation……………………………………………………………………………..31

Future Implementations………………………………………………………………………32

Workload Matrix……………………………………………………………………………….33

**Section 1**

Introduction

* We are going to plan a solution for the university to implement an effective student’s management system.

Currently, the university is operating as a file-based system which is time consuming and will require more space when managing departments in the university. Considering about the mentioned problems, we hope to categorise the departments such as.

* Examination department: This will manage all the subjects and student details separately.
* Registration department: This will maintain all the student registration details and personal details with their educational.
* Library access department: This will maintain and manage all the details of students accessing the library facility.
* IT department: This will maintain and check details of students accessing Wi-Fi and DLE and record their entries.
* In order for the university to operate an efficient student management system, a database structure needs to be implemented as a solution to categorise these departments and maintain records in an advanced manner.
* As we have introduced the solution, we are going to further discuss about the appropriate database design for the given scenario.

**Extended Entity Relationship Diagram**

Diagram

**Diagram

Description automatically generated**

Assumptions

* We assumed that address and name attributes are composite attributes. Which address include the city, street, postal code and name include the first name, last name.
* We assumed that every student must depend on at least one guardian and therefore, we made guardian as a weak entity which is related to the student strong entity.
* We assumed that all the students must participate in learning subjects and all the subjects must have a result. Therefore, we made result as a weak entity which related to both student and subject strong entities. A subject can have many students and a student can learn many subjects and only one result is related to a specific subject and a student must have many results for many subjects but a result must be specific for a one student.
* We assumed that student can become a library member and all the book transactions must be related to only a student who is a library member. Therefore, we made a weak entity as book transaction related to the library member strong entity. Library member can have many book transaction but a specific book transaction must related only to a one library member.
* We assumed that the specialization of the access to DLE, WiFi and Gate Access sub-entities is a disjoint relationship. Therefore, dleID and dlePWD are added to the DLE entity, WiFiID and WiFiPWD are added to the WiFi entity and Gate Number is added to the Gate entity.
* A student can have many access but access can be specified only for a one student.

**Relational Mapping**

Mapping of Strong Entities

1. Student {StudentID[PK] ,First Name, Last Name, DOB, Gender, Batch Number, Degree, Email, City, Street, Postal Code}
2. Library Member {MemberID[PK], First Name, Last Name, Email}
3. Subject {SubjectID[PK], Subject Name, Subject Leader, Credits}
4. Access {AccessID[PK], Access Type, Access Time, Access Location}

Mapping of Weak Entities

1. Student {StudentID[PK] ,First Name, Last Name, DOB, Gender, Batch Number, Degree, Email, City, Street, Postal Code}
2. Guardian {(Name, StudentID[FK])[PK], Relationship, Gender, Contact Number, City, Street, Postal Code}
3. Library Member {MemberID[PK], First Name, Last Name, Email}
4. Book Transaction {(TransactionID, MemberID[FK])[PK], Borrow Date, Due Date, BookID, Book Name, Book Author, Category}
5. Subject {SubjectID[PK], Subject Name, Subject Leader, Credits}
6. Result {(ResultID, SubjectID[FK], StudentID[FK])[PK], Year, Semester, Grade, Date}
7. Access {AccessID[PK], Access Type, Access Time, Access Location}

Mapping of Binary 1:1 Relationships

1. Student {StudentID[PK] ,First Name, Last Name, DOB, Gender, Batch Number, Degree, Email, City, Street, Postal Code}
2. Guardian {(Name, StudentID[FK])[PK], Relationship, Gender, Contact Number, City, Street, Postal Code}
3. Library Member {MemberID[PK], First Name, Last Name, Email}
4. Book Transaction {(TransactionID, MemberID[FK])[PK], Borrow Date, Due Date, BookID, Book Name, Book Author, Category}
5. Subject {SubjectID[PK], Subject Name, Subject Leader, Credits}
6. Result {(ResultID, SubjectID[FK], StudentID[FK])[PK], Year, Semester, Grade, Date}
7. Access {AccessID[PK], Access Type, Access Time, Access Location}

Mapping of Binary 1:M Relationships

1. Student {StudentID[PK] ,First Name, Last Name, DOB, Gender, Batch Number, Degree, Email, City, Street, Postal Code, MemberID[FK]}
2. Guardian {(Name, StudentID[FK])[PK], Relationship, Gender, Contact Number, City, Street, Postal Code}
3. Library Member {MemberID[PK], First Name, Last Name, Email}
4. Book Transaction {(TransactionID, MemberID[FK])[PK], Borrow Date, Due Date, BookID, Book Name, Book Author, Category}
5. Subject {SubjectID[PK], Subject Name, Subject Leader, Credits, ResultID[FK]}
6. Result {(ResultID, SubjectID[FK], StudentID[FK])[PK], Year, Semester, Grade, Date}
7. Access {AccessID[PK], Access Type, Access Time, Access Location, StudentID[FK]}

Mapping of Binary M:N Relationships

1. Student {StudentID[PK] ,First Name, Last Name, DOB, Gender, Batch Number, Degree, Email, City, Street, Postal Code, MemberID[FK]}
2. Guardian {(Name, StudentID[FK])[PK], Relationship, Gender, Contact Number, City, Street, Postal Code}
3. Library Member {MemberID[PK], First Name, Last Name, Email}
4. Book Transaction {(TransactionID, MemberID[FK])[PK], Borrow Date, Due Date, BookID, Book Name, Book Author, Category}
5. Subject {SubjectID[PK], Subject Name, Subject Leader, Credits, ResultID[FK]}
6. Result {(ResultID, SubjectID[FK], StudentID[FK])[PK], Year, Semester, Grade, Date}
7. Access {AccessID[PK], Access Type, Access Time, Access Location, StudentID[FK]}
8. Learn {(StudentID[FK], SubjectID[FK])[PK],Hours}

Mapping of Multi-Valued Attributes

\*\* No Multi-Valued Attributes Found.

Mapping of N-ary Relationships

\*\* No N-ary Relationships Found.

Mapping of Extended-Entity Relationships

1. Student {StudentID[PK] ,First Name, Last Name, DOB, Gender, Batch Number, Degree, Email, City, Street, Postal Code, MemberID[FK]}
2. Guardian {(Name, StudentID[FK])[PK], Relationship, Gender, Contact Number, City, Street, Postal Code}
3. Library Member {MemberID[PK], First Name, Last Name, Email}
4. Book Transaction {(TransactionID, MemberID[FK])[PK], Borrow Date, Due Date, BookID, Book Name, Book Author, Category}
5. Subject {SubjectID[PK], Subject Name, Subject Leader, Credits, ResultID[FK]}
6. Result {(ResultID, SubjectID[FK], StudentID[FK])[PK], Year, Semester, Grade, Date}
7. Access {AccessID[PK], Access Type, Access Time, Access Location, StudentID[FK]}
8. Learn {(StudentID[FK], SubjectID[FK])[PK],Hours}
9. DLE {(AccessID[FK])[PK], dleID, dlePWD}
10. WiFi {(AccessID[FK])[PK], WiFiID, WiFiPWD}
11. Gate {(AccessID[FK])[PK], Gate Number}

**Normalization**

Dependencies

1. Student {StudentID[PK] ,First Name, Last Name, DOB, Gender, Batch Number, Degree, Email, City, Street, Postal Code, MemberID[FK]}
2. Guardian {(Name, StudentID[FK])[PK], Relationship, Gender, Contact Number, City, Street, Postal Code}
3. Library Member {MemberID[PK], First Name, Last Name, Email}
4. Book Transaction {(TransactionID, MemberID[FK])[PK], Borrow Date, Due Date, BookID, Book Name, Book Author, Category}
5. Subject {SubjectID[PK], Subject Name, Subject Leader, Credits, ResultID[FK]}
6. Result {(ResultID, SubjectID[FK], StudentID[FK])[PK], Year, Semester, Grade,Date}
7. Access {AccessID[PK], Access Type, Access Time, Access Location, StudentID[FK]}
8. Learn {(StudentID[FK], SubjectID[FK])[PK],Hours}
9. DLE {(AccessID[FK])[PK], dleID, dlePWD}
10. WiFi {(AccessID[FK])[PK], WiFiID, WiFiPWD}
11. Gate {(AccessID[FK])[PK], Gate Number }

1st Normal Form (1NF)

As there are no multi-valued attributes and has only atomic valued attributes, the above relational schema is already in First Normal Form.

2nd Normal Form (2NF)

As there are no partial dependencies, the above relational schema is already in Second Normal Form.

3rd Normal Form (3NF)

As there are no transitive dependencies, the following relational schema can be considered as in the Third Normal Form.

1. Student {StudentID[PK] ,First Name, Last Name, DOB, Gender, Batch Number, Degree, Email, City, Street, Postal Code, MemberID[FK]}
2. Guardian {(Name, StudentID[FK])[PK], Relationship, Gender, Contact Number, City, Street, Postal Code}
3. Library Member {MemberID[PK], First Name, Last Name, Email}
4. Book Transaction {(TransactionID, MemberID[FK])[PK], Borrow Date, Due Date, BookID, Book Name, Book Author, Category}
5. Subject {SubjectID[PK], Subject Name, Subject Leader, Credits, ResultID[FK]}
6. Result {(ResultID, SubjectID[FK], StudentID[FK])[PK], Year, Semester, Grade,Date}
7. Access {AccessID[PK], Access Type, Access Time, Access Location, StudentID[FK]}
8. Learn {(StudentID[FK], SubjectID[FK])[PK],Hours}
9. DLE {(AccessID[FK])[PK], dleID, dlePWD}
10. WiFi {(AccessID[FK])[PK], WiFiID, WiFiPWD}
11. Gate {(AccessID[FK])[PK], Gate Number }

**Data Dictionary**

1. Student Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Data Type | Constraints | Constraint Name | Reference Table |
| StudentID | INTEGER | PRIMARY | pk\_StudentID |  |
| First Name | VARCHAR(30) | NOT NULL |  |  |
| Last Name | VARCHAR(30) | NOT NULL |  |  |
| DOB | DATE | NOT NULL |  |  |
| Gender | CHAR(1) | NOT NULL |  |  |
| Batch Number | VARCHAR(30) | NOT NULL |  |  |
| Degree | VARCHAR(60) | NOT NULL |  |  |
| Email | VARCHAR(60) | NOT NULL |  |  |
| City | VARCHAR(20) | NOT NULL |  |  |
| Street | VARCHAR(20) | NOT NULL |  |  |
| Postal Code | VARCHAR(10) | NOT NULL |  |  |
| MemberID | INTEGER | FOREIGN | fk\_MemberID | Library Member |

1. Guardian Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Data Type | Constraints | Constraint Name | Reference Table |
| Name | VARCHAR(60) | PRIMARY | pk\_Name |  |
| StudentID | INTEGER | PRIMARY | pk\_StudentID |  |
|  |  | FOREIGN | fk\_StudentID | Student |
| Relationship | VARCHAR(20) | NOT NULL |  |  |
| Gender | CHAR(1) | NOT NULL |  |  |
| Contact Number | VARCHAR(30) | NOT NULL |  |  |
| City | VARCHAR(20) | NOT NULL |  |  |
| Street | VARCHAR(20) | NOT NULL |  |  |
| Postal Code | VARCHAR(10) | NOT NULL |  |  |

1. Library Member Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Data Type | Constraints | Constraint Name | Reference Table |
| MemberID | INTEGER | PRIMARY |  |  |
| First Name | VARCHAR(30) |  |  |  |
| Last Name | VARCHAR(30) |  |  |  |
| Email | VARCHAR(60) |  |  |  |

1. Book Transaction Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Data Type | Constraints | Constraint Name | Reference Table |
| TransactionID | INTEGER | PRIMARY | pk\_TransactionID |  |
| MemberID | INTEGER | PRIMARY | pk\_MemberID |  |
|  |  | FOREIGN | fk\_MemberID | Library Member |
| Borrow Date | DATE | NOT NULL |  |  |
| Due Date | DATE | NOT NULL |  |  |
| BookID | VARCHAR(30) | NOT NULL |  |  |
| Book Name | VARCHAR(30) | NOT NULL |  |  |
| Book Author | VARCHAR(30) | NOT NULL |  |  |
| Category | VARCHAR(20) | NOT NULL |  |  |

1. Subject Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Data Type | Constraints | Constraint Name | Reference Table |
| SubjectID | INTEGER | PRIMARY | pk\_SubjectID |  |
| Subject Name | VARCHAR(30) | NOT NULL |  |  |
| Subject Leader | VARCHAR(30) | NOT NULL |  |  |
| Credits | INTEGER | NOT NULL |  |  |
| ResultID | INTEGER | FOREIGN | fk\_ResultID | Result |

1. Result Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Data Type | Constraints | Constraint Name | Reference Table |
| ResultID | INTEGER | PRIMARY | pk\_ResultID |  |
| SubjectID | INTEGER | PRIMARY | pk\_SubjectID |  |
|  |  | FOREIGN | fk\_SubjectID | Subject |
| StudentID | INTEGER | PRIMARY | pk\_StudentID |  |
|  |  | FOREIGN | fk\_StudentID | Student |
| Year | INTEGER |  |  |  |
| Semester | INTEGER |  |  |  |
| Grade | CHAR(1) |  |  |  |
| Date | DATE |  |  |  |

1. Access Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Data Type | Constraints | Constraint Name | Reference Table |
| AccessID | INTEGER | PRIMARY | pk\_AccessID |  |
| Access Type | VARCHAR(10) | NOT NULL |  |  |
| Access Time | TIME | NOT NULL |  |  |
| Access Location | VARCHAR(10) | NOT NULL |  |  |
| StudentID | INTEGER | FOREIGN | fk\_StudentID | Student |

1. Learn Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Data Type | Constraints | Constraint Name | Reference Table |
| StudentID | INTEGER | PRIMARY | pk\_StudentID |  |
|  |  | FOREIGN | fk\_StudentID | Student |
| SubjectID | INTEGER | PRIMARY | pk\_SubjectID |  |
|  |  | FOREIGN | fk\_SubjectID | Subject |
| Hours | INTEGER | NOT NULL |  |  |

1. DLE Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Data Type | Constraints | Constraint Name | Reference Table |
| AccessID | INTEGER | PRIMARY | pk\_AccessID |  |
|  |  | FOREIGN | fk\_AccessID | Access |
| dleID | VARCHAR(30) | NOT NULL |  |  |
| dlePWD | VARCHAR(30) | NOT NULL |  |  |

1. WiFi Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Data Type | Constraints | Constraint Name | Reference Table |
| AccessID | INTEGER | PRIMARY | pk\_AccessID |  |
|  |  | FOREIGN | fk\_AccessID | Access |
| WiFiID | VARCHAR(30) | NOT NULL |  |  |
| WiFiPWD | VARCHAR(30) | NOT NULL |  |  |

1. Gate Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Data Type | Constraints | Constraint Name | Reference Table |
| AccessID | INTEGER | PRIMARY | pk\_AccessID |  |
|  |  | FOREIGN | fk\_AccessID | Access |
| Gate Number | INTEGER | NOT NULL |  |  |

Following Data Formats are considered at the Data Entry.

1. Date: YY-MM-DD

2. Gender: ‘M’ or ‘F’

**Section 02**

Create Table Statements

1. Student Table

Graphical user interface, text, application

Description automatically generated

1. Guardian Table

Graphical user interface, text, application

Description automatically generated

1. Library Member Table

Text

Description automatically generated

1. Book Transaction Table

Graphical user interface, text, application, email

Description automatically generated

1. Subject Table

Graphical user interface, text, application, email

Description automatically generated

1. Result Table

Text

Description automatically generated

1. Access Table

Text, letter

Description automatically generated

1. Learn Table

Text

Description automatically generated

1. DLE Table

Text, letter

Description automatically generated

1. WiFi Table

Text, letter

Description automatically generated

1. Gate Access ID

Text, letter

Description automatically generated

Database Diagram



Database Sample Records

1. Student Table

Graphical user interface

Description automatically generated with low confidence

1. Guardian Table

Table

Description automatically generated

1. Library Member

Table

Description automatically generated with medium confidence

1. Book Transaction

Table

Description automatically generated with medium confidence

1. Subject

Table

Description automatically generated

1. Result

Table

Description automatically generated

1. Access

Table

Description automatically generated

1. Learn

Chart

Description automatically generated with medium confidence

1. DLE

Table, calendar

Description automatically generated with medium confidence

1. Wi-Fi

Table

Description automatically generated

1. Gate

A picture containing chart

Description automatically generated

**Section 03**

Create Trigger Statements

1. CheckResultAvailability Trigger

Graphical user interface, text, application

Description automatically generated

1. BookInsert Trigger

Graphical user interface, application

Description automatically generated

1. LearnInsert Trigger

Graphical user interface, text, application

Description automatically generated

Create Function Statements

1. StudentDetailsByName Function

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

1. BookBorrowDetailsByMemberID Function

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text

Description automatically generated

Graphical user interface, text

Description automatically generated

1. GuardianDetailsByStudentName Function

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Create View Statements

1. Borrower Details View

Graphical user interface, application, table

Description automatically generated

Graphical user interface, application, table

Description automatically generated

Graphical user interface, application, table

Description automatically generated

1. Subject Leader Details View

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

1. Student Access Details View

Graphical user interface, text, application

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

Create Procedure Statements

1. GetGender Procedure

Graphical user interface, text, application

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

1. GetStudentDetailsByBorrowedDateandDueDate Procedure

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface

Description automatically generated

Graphical user interface

Description automatically generated

**Section 04**

Critical Evaluation

Our relational database management system is planned, designed, and implemented

as a solution to create a Student Management System with a fully functional database application which is located in several areas with Student registration department, examination department, the library and IT department where the student administration is assigned to maintain each and every aspect of the system. The libraries can locate books which are borrowed by students. In

addition, the below functionalities will help the student DLE to easily administrate all

the daily activities regarding subject and exam details with ease and reliability.

• Database will validate all the entered data into the system using the Stored

Procedures and User-Defined Functions so that the Student administration cannot enter invalid data to the system. It comprises the stored procedures for the Student Table,

to get the student details by Gender, to get student details by entering Borrowed

Date and Due date accordingly.

• The database also checks the availability of the results when the

student has completed his/her exams and is waiting to check. Therefore, the triggered table would roll back the execution process.

• The student management system database has its drawbacks such as.

* The Result table is connected with the Subject table with an identifying

relationship, therefore the Result table is a weak entity and is fully dependent on the Subject table, which is also a week entity of student table.

That could create a conflict to the Result and Subject table and might deliver errors in future implements.

* The IT department is bound to maintain the student IT Access division, but since there are two types of access as DLE and Wi-Fi and Gate, the system requires a proper mechanism to allocate the department, respectively.

• To overcome the weaknesses of the database, we might have to come up with

another solution which would consider the result table only as a weak entity of

the subject Table. To allocate the IT staff members accordingly, the system should

be able to provide the DLE, Wi-Fi and Gate access details to the staff separately.

Future Implementations

The current implementation of the Student Management System primarily serves the examination, registration, library and IT related activities and management. As steps and phases of further implementation, we would like to make the following implementations to the current database application.

1. Implementation of the database to handle students borrowing books from library and book return Management functionalities more accordingly which would notify the book return details with the borrowed date and due date to the student accordingly when a due date is close, warning notifications when the due date is passed, and update the Book Transaction table according with the fee information for each delayed book returns, and book return notification which would update the book availability.
2. Implementation of the database application to serve Human Resource

management functionalities required by Human Resource department of

Student Management System, so that the staff can be divided further into academic and non-academic staff which would issue and manage payrolls of the staff and

maintain staff allocation with work hours. Hence, payroll reports and all other

functionalities of the department will also be handled.

III. Implementation of the database to have registered and non-registered

student library members so that only registered members can borrow books.

IV. Implementation of the database to with trigger functions to each table along

with UPDATE and DELETE statements and using Stored Procedures to

improve the efficiency. Using functions with INPUT and OUTPUT parameters

to view and improve the database tables in a meaningful way. Usage of views

to reduce the complexity of the database schema and implement row and

column level security to the tables.

V. Updating the system into better a software with a good interface, making the

current version of the database to keep up with the latest SQL software in the

future, and using built-in encryption to protect sensitive database information.

Workload Matrix

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Index Number | EER Diagram | Relational Schema, Data Normalization | Data Dictionary, Table Constraints | Triggers, Functions, Views, Stored Procedures | Evaluation, Future Implementation |
| 10749130 |  |  |  |  |  |
| 10749170 |  |  |  |  |  |
| 10750062 |  |  |  |  |  |
| 10749135 |  |  |  |  |  |
| 10749140 |  |  |  |  |  |
| 10749131 |  |  |  |  |  |