INMT5526: BUSINESS INTELLIGENCE

BI SOLUTION OF EYE-CLINIC

Zhulin Lyu (24516605)

Individual Assignment

Contents

Background

The increasing number of patients visiting the eye clinic requires it to figure out a good way to deal with this ever-growing concern of handling volumes of patient appointments. We believe a current system, which is mainly manual or based on out-of-date software, is proving cumbersome to process the increase in the patient visiting status. All of this compounded on the clinic's relationship, which resulted in many conflicts such as scheduling conflicts, missed appointments, and check-in/check-out process hassles situation. This complex process manifests itself through operational disruption of the clinic and ultimately customer dissatisfaction and the overall performance of the clinic.

The heart of the problem is that the clinic cannot process the appointments' database in quickly and able to filter out useful information timely. Due to the increased patient base as well as other factors, the volume of the data generated has surged, and therefore, the need for more work force has arisen. Nevertheless, the clinic does not have a structured scheme aimed at successfully processing this data and ultimately predicting the volume of appointments. This may go on to, in turn, lead to the clinic being unable to arrange enough staff at peak times of work or cause some treatment rooms to get overlapped and report delays. Besides, the physicians' inability to avert the patients' dissatisfaction causes them to be nervous or engaged in doctor-patient conflicts.

To address these challenges, implementing a Business Intelligence (BI) solution is essential. BI solution would provide advanced tools to collect, analyse, and visualize data, to support decision-making. The solution would automate the appointment scheduling data, reducing the likelihood of errors and improving overall efficiency. Additionally, the BI system would allow the clinic to identify trends in patient visits, optimize resource allocation, and predict future appointment demand, ensuring that the clinic can meet patient needs effectively.

The benefit of solving this problem is multifaceted. First, it would enhance the clinic's operational efficiency by automating appointment management and reducing the administrative burden on staff. This would allow the clinic to serve more patients without compromising the quality of care. Second, solving this problem would improve patient satisfaction by minimizing wait times, reducing the risk of clash resources, and providing a more reasonable scheduling process. Finally, the ability to analyse and visualize data would give the clinic valuable insights into patient behaviour, enabling it to tailor services to meet patient needs better and ultimately drive growth. In conclusion, by using BI, the clinic can improve its appointment management process, enhance patient satisfaction, and position itself for sustained growth in a competitive healthcare market.

Methodology

To address the challenges faced by the eye clinic in managing appointments, I will design a comprehensive Business Intelligence (BI) solution that uses the power of databases, data modelling, and SQL. The solution will automate the appointment data monitoring, optimize resource allocation, and provide valuable insights through data analysis. This methodology outlines the steps and considerations involved in designing the system, justifying why these steps are appropriate for solving the identified business problem.

1. Identifying Business Intelligence Solution

The foundation of the project is rooted in BI concepts, which focus on transforming raw data into actionable insights. To ensure the BI system effectively addresses the clinic's needs, I will conduct a thorough analysis of the clinic's current appointment management process. This includes understanding how appointments are booked, how patient data is recorded, and how staff currently manage scheduling. This analysis will help identify inefficiencies.

The first step in designing the whole solution is to consider key performance indicators (KPIs) that will drive decision-making, which means consider about what is my purpose after implementing this BI, which goals I want to achieve. These might include metrics such as appointment booking rates, service booking rate, and resource utilization. By identifying these KPIs, that provide a clear view of the clinic's performance.

Design Justification: The design of the BI solution is tailored to address the specific challenges identified in the eye clinic. By leveraging BI concepts, the system will provide the clinic with the tools needed to make data-driven decisions, improving both operational efficiency and patient satisfaction.

2. Database Design and Data Modelling

The core of the BI solution will be a well-structured database. It is critical to design a database that will store all medical treatment data correctly. In this way, it will support efficient querying and reporting. I suggest the clinic design this database with a relational model, which is the suitable solution to handle structured data and perform complex queries.

Design Justification: By using a relational database ensures that patient and appointment data is stored in a structured, efficient way, and supporting complex queries and reporting. Data modelling and normalization make database is extensible and can handle increasing data of the clinic.

2.1 Data Modelling

The first step in database design is to create a data model that accurately represents the clinic's operations. The data model will include the key participators as entities. Each of these entities will be linked through relationships, make storing and retrieving data efficiently. For example, the relationship between Patients and Appointments will be a one-

to-many relationship, as each patient can have multiple appointments. The complete entities are included in the **Appendix 1.**

2.2 Normalization

To ensure the database is efficient and avoids redundancy, normalization techniques will be applied. This involves organizing the data into tables and defining relationships between them to minimize duplication. For example, patient contact information will be stored in a separate table and linked to the appointment table through a foreign key, ensuring that updates to contact information only need to be made in one place.

2.3 Entity relationship diagrams

After identified the relationships between entities, Entity relationship diagrams (ERD) can be drawn to represent this relationship and the different entities in the system. The ERD visually represents how these entities interact, helping to ensure that the database structure supports the clinic's operational needs. An ERD is included in the **Appendix 2.**

3. SQL for Data Management:

Once the database design is complete, SQL (Structured Query Language) will be used to manage the data. SQL will be used for inserting new appointments, updating patient records, and querying the database for specific information. For example, SQL queries can be used to find on the number of appointments booked in each period, identify peak times for appointments, and track popular services. A full SQL is included in the **Appendix 3.**

Design Justification: SQL is a powerful tool for managing and querying the database. It will be easy to find specifical information for the clinic. By integrating BI tools, the clinic will be able to visualize data in a meaningful way.

4. Linking Design to the Business Problem

The BI solution is directly linked to the business problem identified earlier. By automatedly analysing the appointment data and providing visual Entity-Relationship Diagram, the system addresses the inefficiencies and errors in the current clinic's operations. The BI system will help the clinic make informed decisions, leading to improved patient satisfaction and better resource management.

Overall, this whole design is appropriate because it addresses the clinic's current challenges while providing a scalable solution that can grow with the clinic. The use of BI concepts, databases, and SQL ensures that the system is both efficient and capable of delivering the insights needed to optimize the clinic's operations.

Solution

To address the eye clinic's appointment management challenges, I implemented a comprehensive solution based on Business Intelligence (BI) concepts, database design, and SQL. The process involved several key steps, integrating various BI components to provide a robust, data-driven system.

According to the results of business research, the operation of the eye clinic is summarized as follows：

For each actual patient, their name, address, and their Medical History are recorded. The clinic has multiple treatment rooms. Each treatment room has a unique room number and large-scale treatment facility. Similarly, each doctor in clinic has a unique identifier ID recorded along with their name, contact phone number, and can provide one to more services.

There are various services available in the clinic. For each service available, the service code is recorded along with a description and the current service fee. Also identified are those doctors who give this service. Most doctors give many services.

When a patient requires some treatment, one or more appointments are scheduled; each appointment is for a particular date and starting time with a specific doctor. At any one time there is only one appointment for a particular room and a particular doctor.

1. Data Modelling

Base on this situation, the data model will include the following entities:

- **Patient**: This entity will store patient information such as name, address details, medical history, and unique patient ID.
- **Appointment**: This entity will record appointment details, including appointment ID, date and time, patient ID, and the doctor assigned.
- **Doctor**: This entity will contain information about doctors, including doctors' name, their specialty service and contact phone number.
- **Room**: This entity will track the availability of treatment rooms.
- **Service**: This entity will record the specific service that clinic provided, such as eye-sight text or special eye treatments.

And there are following relationships:

- Patient and Appointment --- One to Many (1:M) relationship
- Appointment and Room --- One to One (1:1) relationship
- Appointment and Service --- One to One (1:1) relationship
- Appointment and Doctor --- One to One (1:1) relationship
- Doctor and Service --- Many to Many (M:N) relationship

The complete entities are included in the **Appendix 1.**

2. Entity Relationship Diagrams

As part of the design work, I created an Entity-Relationship Diagram (ERD) that maps out the relationships between the different entities in the system—Patients, Appointments, Staff, and Resources. The ERD visually represents how these entities interact, helping to ensure that the database structure supports the clinic's operational needs.

In particular, because of the many-to-many relationship between Doctor and Service that existed before, I break up to produce a set of 1:M relationships with a composite entity (Service_Doctor). The composite entity table contains the primary keys of the original Doctor and Service tables and is also the foreign key for the Service_Doctor entity (Watt, p. 2014). The ERD is included in the **Appendix 2** of this report for reference.

3. Database Implementation and Data Preparation

The foundation of the solution was the creation of a relational database that structured the clinic's data efficiently. Using SQL, I developed tables for Patient, Appointment, Doctor, Room, and Service, ensuring proper normalization to avoid redundancy and maintain data integrity. This database served as the centre for all data, enabling accurate and efficient data retrieval.

To prepare the data for analysis, I focused on ensuring that the database could handle large volumes of appointment and patient data. This involved setting up basic data tables and adding cleaned data, up-to-date for subsequent analysis. I created some basic KPI query as mentioned before, including monthly appointment booking rates, service booking rate, and Room utilization. These queries will give you an overview of how appointments, services, and rooms are utilized on a monthly basis. A full SQL is included in the **Appendix 3.**

Conclusion

This solution for the eye clinic's appointment management can address the key issues we identified in previous section. By implementing a Business Intelligence (BI) solution, a well-structured database and data analysis tools can be established in their business, helping the clinic is better to manage its growing patient volume and make operational efficiency.

The solution met the business problem in several critical ways:

- **Improved Appointment Source Management:** The BI system automated the appointment scheduling sources, reducing errors and minimizing the risk of scheduling conflicts. The integration of predictive analytics allowed the clinic to anticipate busy periods and allocate resources more effectively, ensuring that patient care remained prompt and efficient.
- **Enhanced Data-Driven Decision Making:** The important data query by the BI tools provided the clinic's management with real-time insights into key performance metrics. This can support decision-making, allow the clinic to optimize its operations and improve patient satisfaction. Filtering the data by various needs made it easier to identify trends and areas for improvement.
- **Increased Patient Satisfaction:** By reducing wait times, minimizing missed appointments, and improving the overall scheduling experience, the solution directly contributed to higher levels of patient satisfaction.

The system successfully addresses the clinic's operational challenges while also providing a stable foundation for future growth. The use of BI tools and concepts enables clinics to not only manage their appointments more efficiently, but also gain a deeper understanding of their operations, enabling continuous improvement.

However, there were several challenges encountered during the development of the solution:

- **Data modelling:** Due to the manual intervention of multiple scenarios in the past business, the systematic business is complicated, and modelling is difficult. By mining the real requirements, considering the underlying logic, and assisting the use of ERD, I successfully built a new data model to meet the business。
- **Data Integration and Cleansing:** One of the main difficulties is integrating the clinic's existing data into the new system and ensuring its accuracy. There are old data that need to be resolved before they can be used a BI system. This is addressed by implementing a data model, cleaning, and standardizing it before importing a new database.
- **User Adoption:** Ensuring that clinic staff were comfortable using the new system was another challenge. I place emphasis on change management and user training to make them gradually adapt to the new processes.

In summary, the solution not only meets the current needs of the eye clinic, but also provides a platform for continuous improvement and scalability. While there were challenges along the way, they were successfully overcome, resulting in a system that I believe will significantly enhance clinic operations.

APPENDIX

Appendix 1 - Data Modelling

Entity
Patient (patientID, name, address, medicalHistory**)** **Primary Key** patientID
Reason
Patient entity has simple attributes including name, address and medical history

Entity
Room (roomNumber, facility) **Primary Key** roomNumber
Reason
Room entity has a simple attribute roomNumber and facility.

Entity
Dorctor (doctorID, name, phoneNumber) **Primary Key** doctorID
Reason
Doctor entity has simple doctorID, name and phoneNumber.

Entity
Service (serviceCode, description, serviceFee) **Primary Key** serviceCode
Reason
Service entity has simple attributes serviceCode, description and serviceFee.

Entity
Service_Doctor (doctorID, serviceCode) **Primary Key** doctorID**,** serviceCode **Foreign Key** doctorID **references** Doctor (doctorID) ON UPDATE CASCADE ON DELETE NO ACTION **Foreign Key** serviceCode **references** Service (serviceCode) ON UPDATE CASCADE ON DELETE NO ACTION
Reason
Creating an entity named Service_Doctor represent a *:* relationship between Doctor and Service. Both doctorID and serviceCode as foreign keys and together form the primary key.

Entity
Appointment (appointmentID, patientID, roomNumber, doctorID, serviceCode, dateAndStartTime) **Primary Key** appointmentID **Foreign Key** patientID **references** Client (patientID) ON UPDATE CASCADE ON DELETE NO ACTION **Foreign Key** roomNumber **references** Room (roomNumber) ON UPDATE CASCADE ON DELETE NO ACTION **Foreign Key** doctorID **references** Doctor (doctorID) ON UPDATE CASCADE ON DELETE NO ACTION **Foreign Key** serviceCode **references** Service (serviceCode) ON UPDATE CASCADE ON DELETE NO ACTION
Reason
Appointment entity has simple attributes date, startTime. patientID has been posted as a foreign key into Appointment from a 1:* relationship between Patient and Appointment UPDATE CASCADE has been chosen to update patientID whenever the attribute changes in the Client table. DELETE NO ACTION has been selected to prevent Patient deletion without first updating Appointment. roomNumber has been posted as a foreign key into Appointment from a 1:1 relationship between Room and Appointment UPDATE CASCADE has been chosen to update roomNumber whenever the attribute changes in the Room table. DELETE NO ACTION has been selected to prevent Room deletion without first updating Appointment. doctorID has been posted as a foreign key into Appointment from a 1:1 relationship between Doctor and Appointment UPDATE CASCADE has been chosen to update doctorID whenever the attribute changes in the Doctor table. DELETE NO ACTION has been selected to prevent Doctor deletion without first updating Appointment. serviceCode has been posted as a foreign key into Appointment from a 1:* relationship between Service and Appointment UPDATE CASCADE has been chosen to update serviceCode whenever the attribute changes in the Doctor table. DELETE NO ACTION has been selected to prevent Service deletion without first updating Appointment.

Appendix 2 - Entity Relationship Diagrams (ERD)

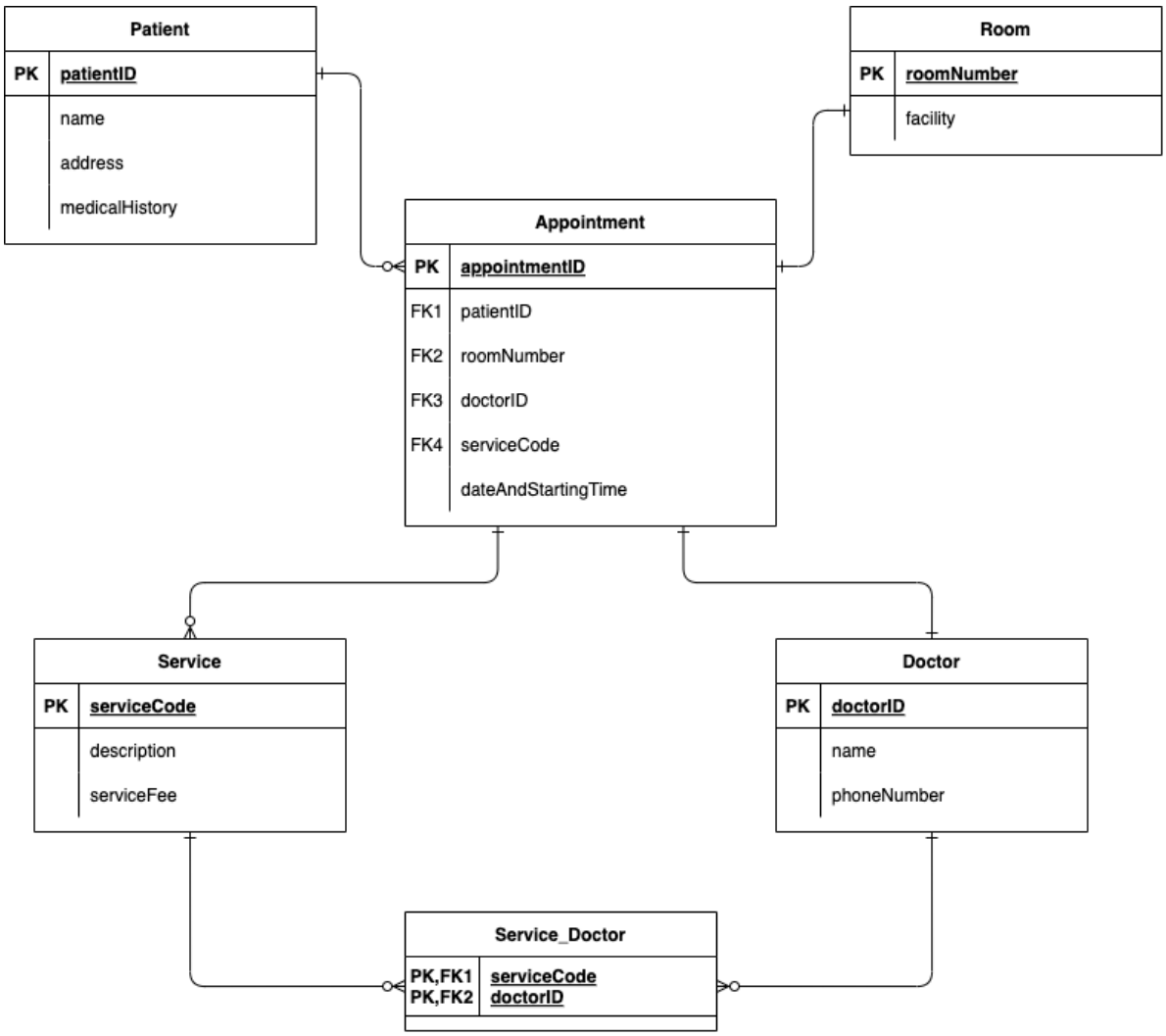

Figure 1: ERD of Eye-Clinic Appointment

Appendix 3 - SQL for Data Management

```
/* Drop tables if it exists*/
DROP TABLE IF EXISTS Patient;
DROP TABLE IF EXISTS Room;
DROP TABLE IF EXISTS Doctor;
DROP TABLE IF EXISTS Service;
DROP TABLE IF EXISTS Appointment;


/* Create entities*/
CREATE TABLE Patient (
   patientID            INT              PRIMARY KEY,
   name                 VARCHAR(20)      NOT NULL,
   address              VARCHAR(45)      NOT NULL
   medicalHistory       VARCHAR(12)      NOT NULL
);

CREATE TABLE Room (
   uniqueNumber         INT              PRIMARY KEY;
   facility             VARCHAR(12)
);

CREATE TABLE Doctor (
   doctorID             CHAR(2)          PRIMARY KEY,
   name                 VARCHAR(20)      NOT NULL,
   phoneNumber          CHAR(14)         NOT NULL
);

CREATE TABLE Service (
   serviceCode          CHAR(4)          PRIMARY KEY,
   description          VARCHAR(20),
   serviceFee           DECIMAL(3, 2)    NOT NULL
);

CREATE TABLE Service_Doctor (
   doctorID             CHAR(2)          NOT NULL,
   serviceCode          CHAR(4)          NOT NULL,
   PRIMARY KEY (doctorID, serviceCode),
   FOREIGN KEY (doctorID) REFERENCES Doctor(doctorID) ON UPDATE CASCADE ON DELETE
NO ACTION,
   FOREIGN KEY (serviceCode) REFERENCES Service(serviceCode) ON UPDATE CASCADE ON
DELETE NO ACTION
);

CREATE TABLE Appointment(
  appointmentID         CHAR(5)          PRIMARY KEY,
  patientID             INT              NOT NULL,
```

```
   roomNumber          INT              NOT NULL,
   doctorID            CHAR(2)          NOT NULL,
   serviceCode         CHAR(4)          NOT NULL,
   dateAndStartTime    DATETIME,
   PRIMARY KEY (appointmentID),
   Foreign Key patientID references Client (patientID) ON UPDATE CASCADE ON DELETE NO
ACTION;
   Foreign Key roomNumber references Room (roomNumber) ON UPDATE CASCADE ON
DELETE NO ACTION;
  Foreign Key doctorID references Doctor (doctorID) ON UPDATE CASCADE ON DELETE NO
ACTION;
  Foreign Key serviceCode references Service (serviceCode) ON UPDATE CASCADE ON
DELETE NO ACTION
);



/* Insert sample data*/
insert into Patient values ('10000001', 'Liang', '11 Towning St', '6062', 'B');
insert into Patient values ('10000002', 'Tian','14/12 Hay St', '6002', 'P');
insert into Patient values ('10000003', 'Ku', '76 Harbour St', '1023', 'C');
insert into Patient values ('10000004', 'Spencer', '99 Edith St', '3020', 'P');
insert into Patient values ('10000005', 'Chen', '99 Edith St', '3020', 'C');
insert into Patient values ('10000006', 'Embleton', '22 akon St', '4900', 'B');

insert into Room values ('101', 'Ophthalmic ultrasonic diagnostic instrument');
insert into Room values ('102', NULL);
insert into Room values ('201', 'OTC');
insert into Room values ('202', 'ophthalmic operating microscope');

insert into Doctor values ('A1', 'Joe', '01224-861212');
insert into Doctor values ('A2', 'Carol', '0141-357-7419');
insert into Doctor values ('B1', 'Tina', '0141-943-1728');
insert into Doctor values ('B2', 'Tony', '0141-225-7025');
insert into Doctor values ('C2', 'Mary', '0141-225-7025');


insert into Service values ('CR01','Comprehensive Eye Exams',425.00);
insert into Service values ('CR02','Prescription Eyeglasses and Contact Lenses',350.00);
insert into Service values ('CR03','Refractive Surgery Consultations',750.00);
insert into Service values ('CR04','Pediatric Eye Care',600.00);
insert into Service values ('CR05','Low Vision Services',300.00);
insert into Service values ('CR06','Emergency Eye Care',90.00);


insert into Service_Doctor values ('A1', 'CR01');
insert into Service_Doctor values ('A1', 'CR02');
```

```
insert into Service_Doctor values ('A1', 'CR03');
insert into Service_Doctor values ('A1', 'CR04');
insert into Service_Doctor values ('A1', 'CR05');
insert into Service_Doctor values ('A2', 'CR02');
insert into Service_Doctor values ('A2', 'CR03');
insert into Service_Doctor values ('B1', 'CR03');
insert into Service_Doctor values ('B1', 'CR04');
insert into Service_Doctor values ('B1', 'CR05');
insert into Service_Doctor values ('B2', 'CR02');
insert into Service_Doctor values ('B2', 'CR03');
insert into Service_Doctor values ('C1', 'CR04');
insert into Service_Doctor values ('C1', 'CR06');


insert into Appointment values ('10001', '10000001', '101', 'A1', 'CR01', '2024-01-02
09:00:00');
insert into Appointment values ('10002', '10000002', '102', 'A1', 'CR02', '2024-04-11
09:00:00');
insert into Appointment values ('10003', '10000002', '102', 'A2', 'CR03', '2024-04-11
13:00:00');
insert into Appointment values ('10004', '10000004', '201', 'B1', 'CR04', '2024-02-16
09:00:00');
insert into Appointment values ('10005', '10000006', '101', 'C1', 'CR04', '2024-03-07
09:00:00');


/* Data Query*/
/* Monthly Appointment Booking Rates */
SELECT
   DATE_FORMAT(dateAndStartTime, '%Y-%m') AS Month,
   COUNT(appointmentID) AS TotalAppointments
FROM
   Appointment
GROUP BY
   Month
ORDER BY
   Month;

/* Monthly Service Booking Rates */
SELECT
   DATE_FORMAT(A.dateAndStartTime, '%Y-%m') AS Month,
   S.serviceCode,
   S.description,
   COUNT(A.appointmentID) AS ServiceBookingCount
FROM
   Appointment A
JOIN
```

```
   Service S ON A.serviceCode = S.serviceCode
GROUP BY
   Month, S.serviceCode
ORDER BY
   Month, S.serviceCode;


/* Monthly Room Utilization */
SELECT
   DATE_FORMAT(A.dateAndStartTime, '%Y-%m') AS Month,
   R.uniqueNumber AS RoomNumber,
   COUNT(A.appointmentID) AS RoomUsageCount
FROM
   Appointment A
JOIN
   Room R ON A.roomNumber = R.uniqueNumber
GROUP BY
   Month, R.uniqueNumber
ORDER BY
   Month, R.uniqueNumber;
```

Bibliography

Oracle, n.d. In: *MySQL Reference Manual.* s.l.:s.n.
Watt, A., 2014. Entity Relationship Data Model. In: *Database Design.* s.l.:s.n.
Watt, A., 2014. Structured Query Language. In: *Database Design.* s.l.:s.n.
Watt, A., 2014. Data Manipulation Language. In: *Database Design.* s.l.:s.n.
Watt, A., 2014. Integrity Rules and Constraints. In: *Database Design.* s.l.:s.n.