Data Management (COMP7060)

Boscombe University Database

Submitted by:
Nazish Khalid
Email

1. Conceptual Design

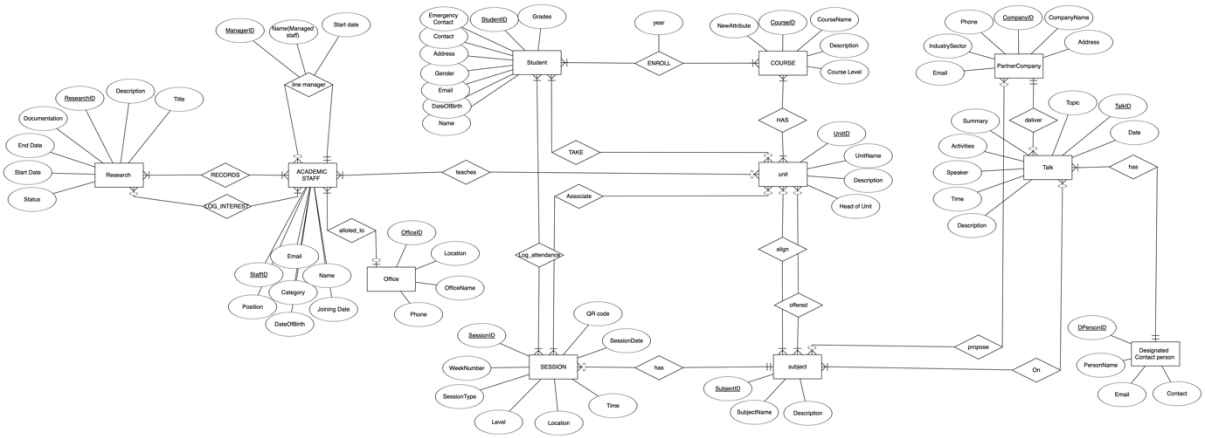

Figure 1: Entity Relationship Diagram (ERD)

Entities:

1. Academic Staff
    - Attributes: StaffID (Primary Key), ManagerID (Foreign Key) Name, DateOfBirth, Email, Position, Category, Joining Date
2. Research
    - Attributes: ResearchID (Primary Key), Description, Title, Status, Documentation, Start Date, End Date
3. Office
    - Attributes: OfficeID (Primary Key), OfficeName, Location, Phone
4. Unit
    - Attributes: UnitID (Primary Key), UnitName, Description, Head of Unit
5. Subject
    - Attributes: SubjectID (Primary Key), SubjectName, Description
6. Student
    - Attributes: StudentID (Primary Key), Name, DateOfBirth, Email, Gender, Address, Contact, Emergency Contact,grades
7. Course
    - Attributes: CourseID (Primary Key), CourseName, Description, Prerequisites, Course Level
8. Session
    - Attributes: SessionID (Primary Key), SubjectID (Foreign Key), SessionDate, Time, Location, SessionType, Level, WeekNumber, QR code
9. PartnerCompany
    - Attributes: CompanyID (Primary Key), CompanyName, Address, Email, IndustrySector, Phone
10. Talk
    - Attributes: TalkID (Primary Key), CompanyID(Foreign Key), DesignatedPersonID(Foreign Key), Topic, Description, Date, Time, Location, Speaker, Activities, Attendance, Audience, Summary
11. Designated Person

- Attributes: DPersonID (Primary Key), PersonName, Email, Contact

12. Record
    - Attributes: StaffID (Composite Primary Key), ResearchID (Composite Primary Key)
13. LogInterest
    - Attributes: StaffID(Composite Primary Key),ResearchID (Composite Primary Key)
14. AllotedTo
    - Attributes: OfficeID (Composite Primary Key), StaffID (Composite Primary Key)
15. Teaches
    - Attributes: StaffID(Composite Primary Key), UnitID (Composite Primary Key)
16. Offered
    - Attributes: UnitID (Composite Primary Key), SubjectID (Composite Primary Key)
17. Enroll
    - Attributes: StudentID (Composite Primary Key), CourseID (Composite Primary Key), Year
18. HasUnit
    - Attributes: UnitID (Composite Primary Key), CourseID (Composite Primary Key)
19. Take
    - Attributes: UnitID (Composite Primary Key), StudentID (Composite Primary Key)
20. LogAttendance
    - Attributes: StudentID (Composite Primary Key), SessionID (Composite Primary Key)
21. Associate
    - Attributes: SessionID (Composite Primary Key), UnitID (Composite Primary Key)
22. Propose
    - Attributes: CompanyID (Composite Primary Key), SubjectID (Composite Primary Key)
23. Align
    - Attributes: SubjectID(Composite Primary Key), UnitID (Composite Primary Key)
24. TalkONSubject
    - Attributes: TalkID(Composite Primary Key), SubjectID (Composite Primary Key)

Relationships

1. Academic staff records Research Activities (Many-to-Many):
    - Connects Academic Staff and Research
    - Justification: Each academic staff member is required to record their research activities, and they have the option to record multiple research activities. Each record of a research activity must be associated with at

least one academic staff member, and it can be recorded by more than one academic staff member.

2. Academic staff Logs Interest in Research interest (Many-to-Many):
   - Connects Academic Staff and Research
   - Justification: Academic staff have the option to express interest in various research activities, although it is not obligatory for every staff member to do so. However, each research activity must be associated with at least one academic staff member and may involve more than one.
3. Academic staff Line Manages other staff members (One-to-Many):
   - Connects Academic Staff to itself.
   - Justification: There exists a recursive relationship within the academic staff, wherein academic staff members have the option to manage other staff members. Conversely, every staff member must be managed by at least one other staff member.
4. Academic staff allotted to Office (Many-to-One)
   - Connects Academic Staff to Office
   - Justification: Academic staff members share an office space, and not every staff member is assigned an individual office. However, each office is required to be occupied by one or more staff members.
5. Academic staff teaches Unit (Many-to-Many)
   - Connects Academic Staff to Unit
   - Justification: Academic staff members may instruct multiple units, but not all staff members are necessarily involved in teaching. Each unit must be instructed by at least one staff member, and a single unit may be taught by one or more staff members.
6. Student Enrolled in Course (Many-to-Many)
   - Connects Student and Course
   - Justification: Students are required to enrol in a one course and can enrol in multiple courses, but only one course per academic year. Conversely, each course must have at least one student enrolled, and it may have more than one student.
7. Course has Units (Many-to-Many)
   - Connects Course and Units
   - Justification: Each course must consist of multiple units. Each unit belongs to at least one course and can belong to more than one.
8. Student take Units (Many-to-Many)
   - Connects Students and Units.
   - Justification: Students can choose units, with some being compulsory and others optional. Each unit must have at least one enrolled student, and it can have multiple students.
9. Unit Offers Subjects (Many-to-Many)
   - Connects Units and subjects.
   - Justification: Each unit must offer multiple subjects. Subjects can be proposed and may not necessarily be part of any specific unit.
10. Student logs attendance in Session (Many-to-Many)
    - Connects Students and Sessions
    - Justification: Students log attendance for each attended session, participating in multiple sessions. Each session must have at least one student and may have more than one student in attendance.

11. Session associates with Unit (Many-to-Many)
    - Connects Session and Unit
    - Justification: A session may be part of one or more units, but it is not mandatory. Each unit consists of a set of sessions.
12. Session has Subject (Many-to-one)
    - Connects Session and Unit
    - Justification: Each session must be associated with one particular subject. Conversely, a subject may have one or more sessions, but it is not mandatory.
13. Partner Company delivers Talks (One-to-Many)
    - Connects Partner Company and Talks
    - Justification: A partner company can deliver multiple talks, but not all companies deliver talks. Each talk must be delivered by one company.
14. Subject May Align with Unit (Many-to-Many)
    - Connects Talk and Unit
    - Justification: Subjects may align with units, but it is not necessary. However, every unit must have aligned subjects.
15. Talks on Subject (Many-to-Many)
    - Connects Talk and Subject
    - Justification: Every talk must have one or more subjects. Subjects can have one or more talks, but it is not compulsory.
16. Partner Company may propose Subjects (Many-to-Many)
    - Connects Partner Company and Talks
    - Justification: Partner companies can propose one or more subjects, but not all companies do. On the other hand, a subject may be proposed by a company, but it is not necessary.
17. Talks has a Designated Contact Person (Many-to-One)
    - Connects Talk and Designated Contact Person
    - Justification: Every talk has one designated contact person for effective communication. Conversely, one designated contact person must have one or more talks.

2. Logical Design

1. AcademicStaff(StaffID, ManagerID* Name, DateOfBirth, Email, Position, Category, joining date)
2. Research(ResearchID, Description,Title,Status,Documentation,Start Date, End Date)
3. Office (OfficeID , OfficeName, Location, Phone)
4. Unit:(UnitID, UnitName, Description, Head of Unit)
5. Subject(SubjectID, SubjectName, Description)
6. Student(StudentID , Name,DateOfBirth, Email, Gender, Address, Contact, Emergency Contact, Grades)
7. Course:(CourseID , CourseName, CourseCode, Description, Prerequisites, Course Level)
8. Session(SessionID , SubjectID*, SessionDate, Time, Location, SessionType, Level, WeekNumber,QR code)
9. PartnerCompany(CompanyID,CompanyName,Address,Email,IndustrySector,Phone)

10. Talk(TalkID, CompanyID*, DesignatedPersonID*,Topic, Description, Date, Time, Location,Speaker ,Activities, Attendance,Audience,Summary,)
11. Designated Person(DPersonID, PersonName, Email, Contact)
12. Record(StaffID*,ResearchID*,)
13. LogInterest(StaffID*,ResearchID*)
14. AllotedTo(OfficeID*, StaffID*)
15. Teaches(StaffID*,UnitID*)
16. Offered(UnitID*,SubjectID*)
17. Enroll(StudentID*, CourseID*, Year)
18. HasUnit(UnitID*, CourseID*)
19. TAKE(UnitID*, StudentID*)
20. LogAttendance(StudentID*,SessionID*)
21. Associate(SessionID*, UnitID*)
22. Propose(CompanyID*,SubjectID*)
23. Align(SubjectID*, UnitID*)
24. TalkONSubject(TalkID*, SubjectID*)

3. Report Creation

Report 1: The sessions most students attended out of all subjects delivered by an external company.

- Session(SessionID , SubjectID*, SessionDate, Time, Location, SessionType, Level, WeekNumber,QR code)
  is joined with
  LogAttendance(StudentID*,SessionID*)
  Joining field attribute is SessionID
- LogAttendance(StudentID*,SessionID*)
  is joined with
  Student(StudentID , Name,DateOfBirth, Email, Gender, Address, Contact, Emergency Contact, Grades)
  Joining field attribute is StudentID
- Session(SessionID , SubjectID*, SessionDate, Time, Location, SessionType, Level, WeekNumber,QR code)
  is joined with
  Subject(SubjectID, SubjectName, Description)
  Joining field attribute is SubjectID
- Subject(SubjectID, SubjectName, Description)
  is joined with
  Propose(CompanyID*,SubjectID*)
  Joining field attribute is SubjectID
- Propose(CompanyID*,SubjectID*)
  is joined with
  PartnerCompany(CompanyID,CompanyName,Address,Email,IndustrySector,Phone)
  Joining field attribute is CompanyID

Report 2: The unit with the most sessions delivered by a company on any subject.

- Unit(UnitID, UnitName, Description, HeadofUnit)
  is joined with

Associate(SessionID*, UnitID*)
Joining field attribute is UnitID

- Associate(SessionID*, UnitID*)
  is joined with
  Session(SessionID ,SubjectID*, SessionDate, Time, Location, SessionType, Level, WeekNumber)
  Joining field attribute is SessionID
- Session(SessionID ,SubjectID*, SessionDate, Time, Location, SessionType, Level, WeekNumber)
  is joined with
  Subject(SubjectID, SubjectName, Description)
  Joining field attribute is SubjectID
- Subject(SubjectID, SubjectName, Description)
  is joined with
  Propose(CompanyID*,SubjectID*)
  Joining field attribute is SubjectID
- Propose(CompanyID*,SubjectID*)
  is joined with
  PartnerCompany(CompanyID,CompanyName,Address,Email, IndustrySector,Phone)
  Joining field attribute is CompanyID

Report 3: The academic staff with the most Research Interests.

- AcademicStaff(StaffID, Name, DateOfBirth, Email)
  is joined with
  LogInterest(StaffID*,ResearchID*)
  Joining field attribute is StaffID
- Research(ResearchID, Description,Title,Status,Documentation,Start Date,End Date)
  is joined with
  LogInterest(StaffID*,ResearchID*)
  Joining field attribute is ResearchID

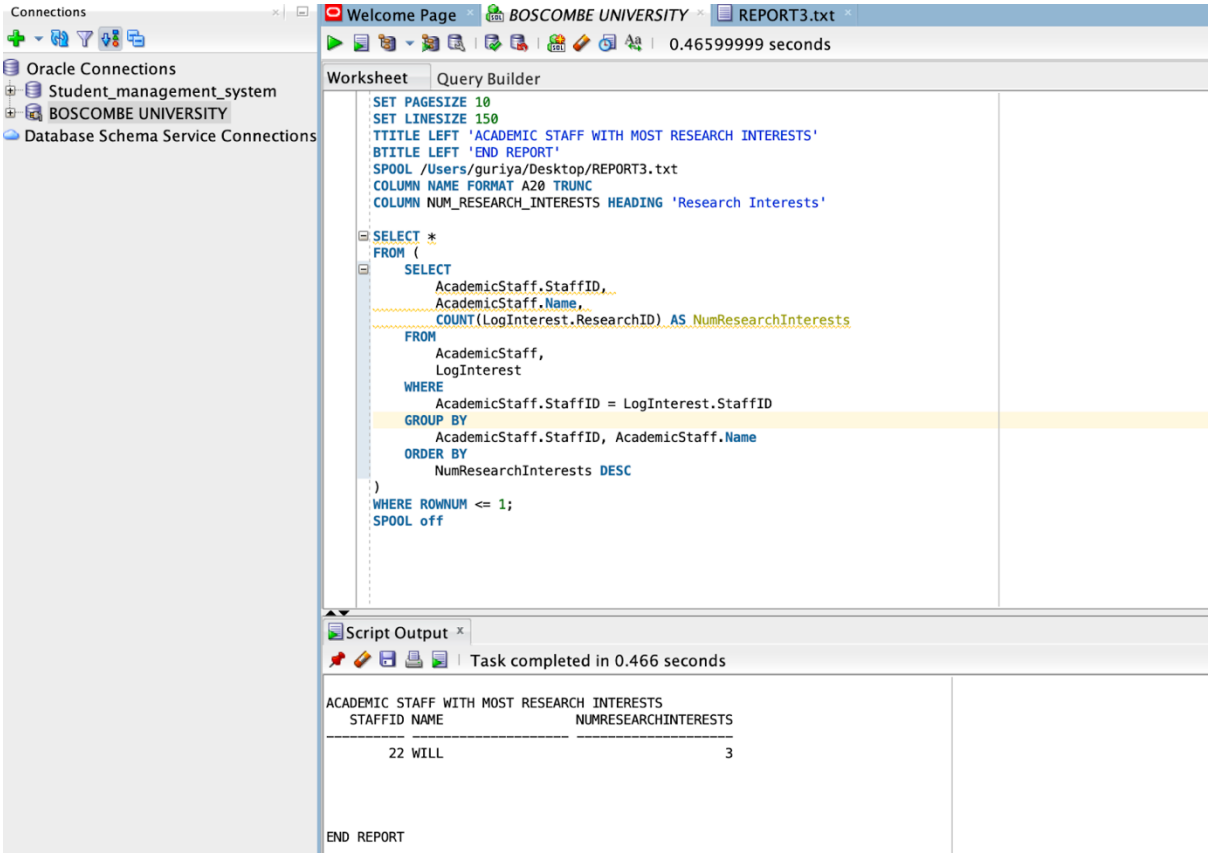

Figure 2: Creating Report query in SQLDeveloper

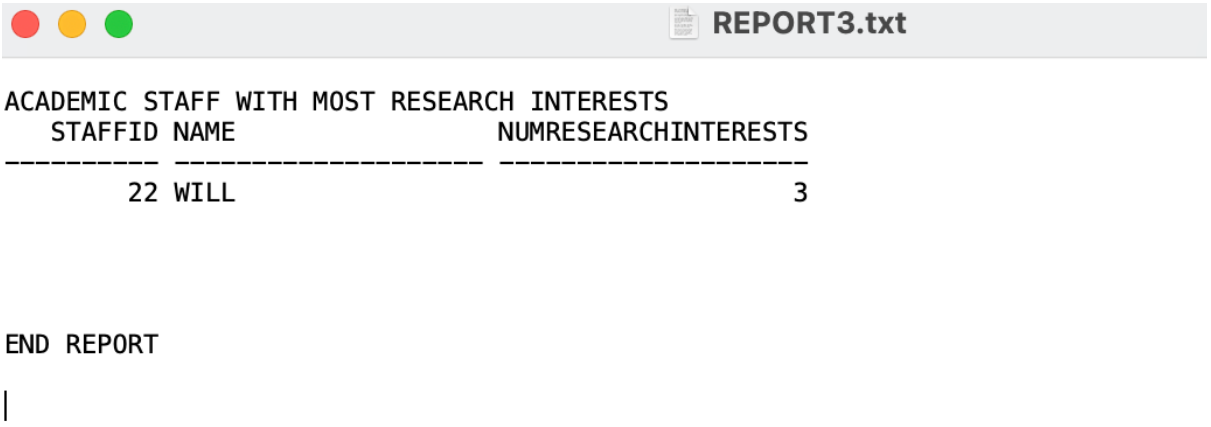

Figure 3: Preview of report created in SQLDeveloper in a file

4. Data Accuracy Strategy

Accurate Data is the most crucial dimension of information quality– the primary and essential aspect (Olson, 2003). Strategies for ensuring data accuracy involve precise data types, atomic entries, clear entry guidelines, and validation checks. Constraints in conceptual design contribute to meaningful relationships, while predictive models and graphical representations enhance data analysis. Accurate data is not merely a dimension but a foundational principle for the reliability and success of the database.

Data Types
Choosing appropriate data types is essential as it guarantees data accuracy. It involves selecting specific data types for each column to optimize storage efficiency and ensure accurate representation of the data. In computer programming, data is stored in variables, and each variable must be assigned a distinct data type. Different data types serve various purposes. For instance, VARCHAR2 is commonly used for string data, INTEGER for numerical values, DATE for dates, and BOOLEAN for true/false values. Matching the nature of the data with the appropriate data type helps maintain consistency and precision in data storage and processing.

Atomicity
Each entry in the database needs to be atomic, meaning it should contain unique and non-repeating information. For instance, in the student table, it's required to have a minimum of two contact numbers. To achieve this, contact details are kept separate for regular and emergency contacts, ensuring a clear distinction instead of combining both numbers into a single field.

Guidelines
To maintain system integrity in a database with numerous users, it's crucial to ensure accurate data entry. Establishing clear input guidelines helps in achieving this goal. For instance, specifying a consistent date format, such as dd-mm-yyyy, helps standardize how individuals input dates. This consistency not only enhances data accuracy but also makes it easier to manage and retrieve information from the database.

Input validation
Validating input is essential to ensure data integrity. Various validation types can be employed to guarantee the entered data is valid.

- Presence Check:
  - Confirm that the input is not an empty string.
- Format Check:
  - Ensure the email follows the basic format, such as user@example.com.
  - Verify the presence of the '@' symbol and at least one '.' after it.
- Length Check:
  - Validate that the data length is within a reasonable range.
  - Check the validity of the country code for contact information.
- Character Set Check:
  - Confirm that the input contains only valid characters; for instance, email addresses should allow letters, numbers, dots (.), hyphens (-), and underscores (_).
- Uniqueness Check:
  - Verify that data for unique attributes (Primary keys) is not repeated.

These validation checks collectively contribute to maintaining data accuracy and conformity to specified standards (Jena, 2022).

Constraints
In conceptual design, constraints play a crucial role in defining the semantics of data by placing restrictions on acceptable data values for entities and their attributes (Morris, 1994).

- Uniqueness: Certain attributes (e.g., student ID) may serve as unique identifiers for students.
- Existence Dependence: Some student information may depend on other entities, ensuring meaningful data relationships (e.g., enrollment in a course depends on student existence).
- Domains: Constraints on attribute domains, specifying valid values (e.g., restricting birthdates to plausible ranges).
- Optionality: Optional information associated withTalks (e.g., hands-on activities) can be captured without affecting data validity.
- Cardinality: Constraints on how Academic Staff members teach units (e.g., a staff member may be part of multiple units).

Efficiency
Predictive Model: Subjects with higher attendance session will have better grades than with low attendance percentage ?
Utilizing QR codes for attendance automates the process, minimizing errors and ensuring data accuracy with unique identifiers. Integrating attendance data into a predictive model enables historical analysis, revealing patterns linking attendance rates to academic performance. This approach involves data collection, feature engineering, model training, and continuous analysis, providing valuable insights for refining the model based on factors influencing academic success (OpenAI ChatGPT 3.5 ,2023).

Graphical Data Representation
Utilize charts and graphs to visually communicate data quality metrics, trends, and anomalies. Graphical representations enhance comprehension, enabling stakeholders to quickly identify areas needing attention and make informed decisions during the data auditing process (Soukup et al. 2002)

References

1. Morris, K., 1994. *Database management systems in engineering*. DIANE Publishing. Accessed: 22 December 2023).
2. Jena, M., 2022. *What is Data Validation?: It's Working and Importance Simplified 101 - Learn*, *Learn | Hevo*. Hevo Data. Available at: https://hevodata.com/learn/data-validation/ (Accessed: 23 December 2023).
3. OPENAI CHATGPT 3.5 , 2023. ChatGPT response to Nazish Khalid , (personal communication, 25th December)
4. Soukup, T. and Davidson, I., 2002. *Visual data mining: Techniques and tools for data visualization and mining*. John Wiley & Sons. (Accessed: 25 December 2023).
5. Olson, J. E., 2003. *In*: *Data quality: The accuracy dimension*. San Francisco: Morgan Kaufmann. (Accessed: 5 January 2024).