Analysis, Design and Implementation of a Relational Database System WRIT1

Kyle Cuthbert – st20249341

Table of Contents

Introduction

What is this report about?

I have been provided with an example scenario to carry out the steps required to analyse the data provided, design a suitable schema, and implement the design that I have decided is the best for the specification provided.

The main question I am aiming to answer is “what type of database system should I implement?”.

Once I have analysed the information that has been provided, I hope to determine an efficient way to collect and store data.

In addition to efficiency, data security and data integrity will be discussed to identify any risks that can occur with regards to data and database management systems. This will include looking into threats, deleting and amendment anomalies, user authorization, and protecting sensitive data.

As part of deciding how best to store data in a database system, steps will be taken to decide what design would be most relevant to the brief that has been provided.

These steps will include performing Normalization to 3NF, drawing an ERD diagram to explain the relationships between data tables, and creating a database schema and a data dictionary. During the design phase, each of these steps will be explained to understand how each has been applied.

To finish, a conclusion will be provided to reflect on the choice of database that was decided upon and whether it was the right choice. If any improvements have been identified during the process, these will be discussed as part of this conclusion.

Data Security and Data Management Concerns

Information is an asset that an organization must properly look after.

For each of these aspects, considerations will be given to prevent data threats.

A good example of these threats has been included in the following table. (Connolly & Begg, 2014)

THREAT	THEFT AND FRAUD	LOSS OF CONFIDENTIALITY	LOSS OF PRIVACY	LOSS OF INTEGRITY	LOSS OF AVAILABILITY
Using another person's means of access	✓	✓	✓		
Unauthorized amendment or copying of data	✓			✓	
Program alteration	✓			✓	✓
Inadequate policies and procedures that allow a mix of confidential and normal output	✓	✓	✓		
Wire tapping	✓	✓	✓		
Illegal entry by hacker	✓	✓	✓		
Blackmail	✓	✓	✓		
Creating "trapdoor" into system	✓	✓	✓		
Theft of data, programs, and equipment	✓	✓	✓		✓
Failure of security mechanisms, giving greater access than normal		✓	✓	✓	
Staff shortages or strikes				✓	✓
Inadequate staff training		✓	✓	✓	✓
Viewing and disclosing unauthorized data	✓	✓	✓		
Electronic interference and radiation				✓	✓
Data corruption owing to power loss or surge				✓	✓
Fire (electrical fault, lightning strike, arson), flood, bomb				✓	✓
Physical damage to equipment				✓	✓
Breaking cables or disconnection of cables				✓	✓
Introduction of viruses				✓	✓

Figure 1 Data Management Concerns

For an organization to properly look after the information that they maintain in a database the organization must consider all aspects of the database including the software, hardware, users, and data. (Connolly & Begg, 2014)

Hardware security considerations would include who has access to the physical hardware e.g., are premises/rooms locked and only authorized people can access?

For software, considerations such as ensuring updates are installed when available would be advised as these could include fixes to security issues that may have been discovered. Installing anti-virus and/or firewalls would be advised to help avoid any malicious activities taking place. In addition, a strong password policy would be recommended.

Staying up to date with who has access to various systems helps to make sure people don't have access any longer than required. This would include setting various levels of authority so that users only have access to information they require. Staff training on the organization's policies and procedures should also be given to users so that they know what is expected of them about data security. (ico, 2022)

What are the Data Security concerns associated with the development a database?

There are three principles of data security, and these are confidentiality, integrity, and accessibility. These three principles are considered the main security goals that organizations should endeavour to address. Below is a summary of what each of these means to a database management system.

Confidentiality – To protect against the unauthorized disclosure of information that could breach laws such as the data protection act or loss of confidence in the organization. (Elmasri, 2014)

Integrity – To protect against the improper modification of data. If data has been corrupted either intentionally or unintentionally can lead to inaccurate data. This will result in incorrect decision-making and possibly fraud. (Elmasri, 2014)

Accessibility – This is the availability of the data so that it can be accessed when required by the relevant authorized personnel. (Elmasri, 2014)

What are the Data Management concerns associated with the development of a database?

As well as security concerns there are also data management concerns associated with databases.

A concern when thinking of hardware would be loss of information that could be caused by an unforeseen event such as a fire, flood, or physical damage to premises or hardware. To mitigate the risk of data loss by ensuring backups are taken.

For secure data, there should be a consideration of whether data encryption is required. Data encryption is the process of masking data into another form that can only be accessed with a decryption key. (Groot, 2022)

Data integrity would be a concern that could be a programmer not following a good database structure whereby the constraints are properly set or a programmer bypassing certain constraints. An example of this could be where a child foreign key attribute is permitted to be deleted when the parent record is still valid.

Relational Database Evaluation

What is a relational database?

A relational database is a concept that was first defined by E.F. Codd in 1970 as part of a paper that he had written called “A Relational Model of Data for Large Shared Data Banks.” as a database system to maintain and hold information. (IBM, n.d.)

Tables, rows, and columns are used to structure information with relationships defined between tables that are separated into relevant entities to enable an efficient way of storing, querying, and maintaining information. (Google Cloud, 2022)

What is the data model employed?

The data model that has been employed for this assessment is the Relationship Model.

For the project scenario that I was provided, other data models were considered and reviewed. These were Non-Relational Systems which include Flat-File, Hierarchical, and Network systems. These early examples of database systems were limited in that they required the developers of systems to know a lot about the data and how it is structured to retrieve the data that they required. These systems were considered inefficient for application performance and maintainability. (Oracle, 2022)

What are the strengths and weaknesses of the relational model?

The Relationship Model is popular because of its many strengths and advantages that it has over Non-Relational Systems that were mentioned previously. It is widely considered an effective way of storing information.

The relational model is easy to use with only little training required. Data can be amended or changed without the need to change the entire record. A record can be structured over multiple tables that can be accessed via SQL queries that can display only information required from both parent and child tables. Weaknesses of the relational model are that it can have costly hardware overhead when the data grows and can have complex mappings which can make understanding difficult for the user. (Nadeem, 2020)

The Hierarchical model works with a parent-to-child relationship but is restricted by only a one-to-many relationship structure. This is where a child can only have one parent node, and it uses pointers to physically connect the physical locations of the data together. This can result in redundant data as records could possibly require the same information as another child whose parent is different.

I applied my understanding of the hierarchical model to the project scenario that I put through the normalization process, and I have created the below diagram to illustrate the parental restriction of the Hierarchical model.

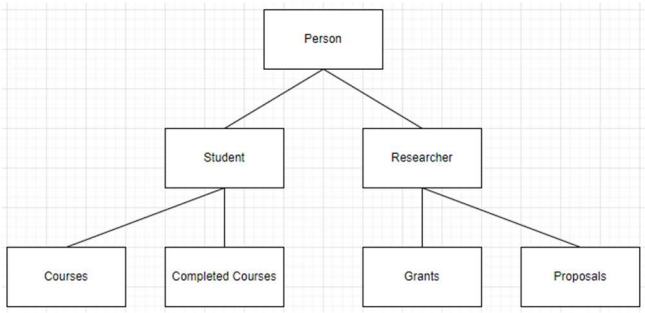

Figure 2 Hierarchical Example1

In this example the 'Researcher' also has the need to provide 'Completed Courses' but it cannot as the child table can only have one parent. For this to be possible in the Hierarchical model something such as the below would be required which in turn creates redundant data.

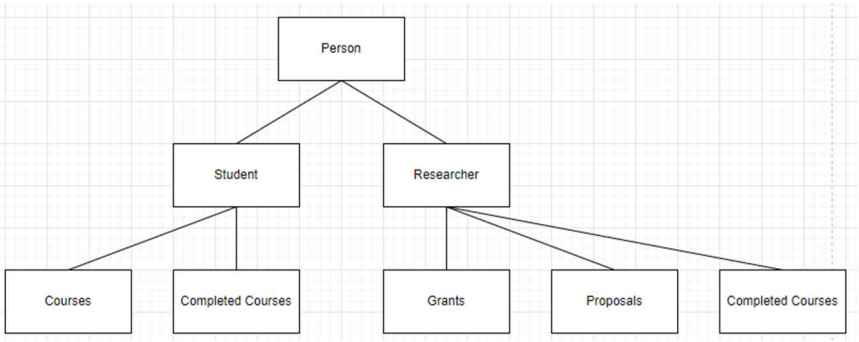

Figure 3 Hierarchical Example2

To query or retrieve data, the developer of the system that uses the data would be required to know the structure of the tree. The relational model does not have the limitation of the one-parent rule, or the issue of relationships being maintained by physical pointers.

Another problem of the Hierarchical model is data redundancy. Another weakness is that queries can be slow to return as the search will need to go from the top down in the hierarchy/structure to locate the record sequentially. (Nadeem, 2020)

Due to the limitations of the Hierarchical model, the Network model was created to address some of these limitations. The Network model allows tables to have more than one parent table and therefore has many-to-many relationships. This in turn restricts the amount of redundant data.

Although some of the restrictions were addressed, the Network Model is still complicated for programmers to use when accessing data due to its complex structure.

Using the project scenario once again, and applying the Network model, the diagram would now look more like the below and whilst it solves the restrictions of only having one-to-many relationships between parent and child nodes and being easier to access data, it is still quite complex for programmers to access.

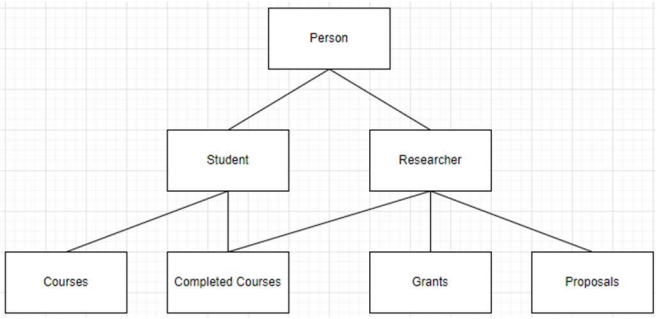

Figure 4 Network Example

Below is a tabular overview of the advantages the Relational database model has over, Hierarchical and Network (Lithmee, 2019)

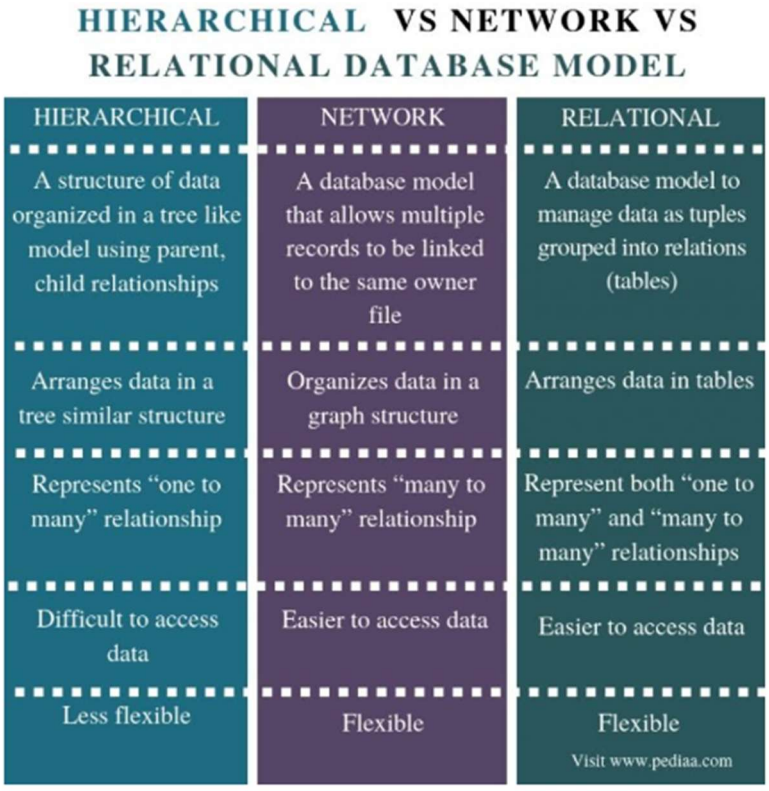

Figure 5 Database Model Advantages and Disadvantages

Rational for Use

Most relational databases have many features that could help to mitigate data security and data management concerns.

For example, Microsoft SQL Server Management Studio includes support to have user accounts whereby the permission of the user can be set to various levels of data access. This will mean that users can only access the data that they are authorized to view or amend.

System backups could also be applied to the database that has been built so that if there is ever an unforeseen event such as a flood, fire, or any other disasters the organisation can restore the data and recover the service that is being provided.

Data encryption can also be included in most relational databases so that even if a database is somehow accessed by and unauthorized user, through poor maintenance of user accounts or as the result of a security breach, the sensitive data would not be able to be accessed.

As well as these features there is the ability to add logging and auditing functionality into most relational databases which can help to keep track of which users have accessed or modified data as well as keep information such as date and times that data was accessed. (IBM, 2019)

Conclusion

Was the relational database the right choice?

Having been provided with the scenario for this assignment I needed to answer, “what type of database model and system should Implement?”.

To be able to answer this with a level of confidence I have looked at both relational and non-relational databases, compared the strengths and weakness of both types and decided on a model to follow.

After deciding that there were more reasons to follow the relational model system, I started to follow the process of designing a database and could quickly see that there were clear boundaries when following the process.

I have carried out normalization to 3NF and quickly discovered that I could do it better by having an entity for a person rather than having a separate student and researcher.

This allowed for linking both entities with commonality and duplicate info in each. Having a separate positions table has enabled me to append further positions as well as students and researchers should there ever be a need. The database could also cater to a researcher that is also a student so is somewhat future proof.

During the design phase, I created various diagrams and design specifications that I found really assisted me when implementing the database as well as being a good reference for other stakeholders in the database i.e., project managers, business analysts, or other developers.

Having implemented the database I found I was easily able to access and interrogate data from all the tables via a range of SQL queries that updated, deleted, and selected various records whilst also keeping their integrity.

With all of this considered I believe the relational model was the right choice to implement as it offers flexibility to grow the database, removes duplication and redundant data in addition to offering features to assist with security goals of the organization.

How could the existing solution be improved?

Some of the improvements that I would investigate to implement this database in the real world would be to research the higher forms of normalization, implement security features and to review the constraints.

For normalization I would look at what can be offered to improve the database with Boyce-Codd normal form and 4NF.

Security that I would like to implement would be to research encryption for various fields within the 'Person' table so that if data were illegally accessed it would not be able to be read. In addition, I would like to put user access and authority levels into place.

I realise that with my tables I have used only a limited number of datatypes. For a real implementation I would like to do more research into what the max lengths would be for columns so that the database isn't using more than it requires.

Finally, I would review and refactor the constraints to ensure they are behaving as they should. When completing the data dictionary, I noticed that I had not considered whether each field should be unique or nullable so I would review this.

References

Connolly, T. & Begg, C., 2014. *Database Systems: a Practical Approach to Design, Implementation, and Management, Global Edition.* 6 ed. s.l.:Pearson Education, Limited.

Elmasri, R., 2014. *Fundamentals of database systems.* Seventh edition, Global edition ed. Boston: Pearson.

Google Cloud, 2022. *what-is-a-relational-database.* [Online]
Available at: https://cloud.google.com/learn/what-is-a-relational-database
[Accessed 20 November 2022].

Groot, J. D., 2022. *what-data-encryption.* [Online]
Available at: https://digitalguardian.com/blog/what-data-encryption
[Accessed 03 12 2022].

IBM, 2019. *database-security.* [Online]
Available at: https://www.ibm.com/uk-en/cloud/learn/database-security
[Accessed 08 12 2022].

IBM, n.d. *Icons of Progress.* [Online]
Available at: https://www.ibm.com/ibm/history/ibm100/us/en/icons/reldb/
[Accessed 20 11 2022].

ico, 2022. *11-practical-ways-to-keep-your-it-systems-safe-and-secure.* [Online]
Available at: https://ico.org.uk/for-organisations/sme-web-hub/whats-new/blogs/11-practical-ways-to-keep-your-it-systems-safe-and-secure/
[Accessed 03 12 2022].

Lithmee, 2019. *pediaa.com.* [Online]
Available at: https://pediaa.com/what-is-the-difference-between-hierarchical-network-and-relational-database-model/#:~:text=The%20main%20difference%20between%20hierarchical,model%20organizes%20data%20in%20tables.
[Accessed 04 12 2022].

Nadeem, N., 2020. *hierarchical-vs-relational-database.* [Online]
Available at: https://dataintegrationinfo.com/hierarchical-vs-relational-database/
[Accessed 04 12 2022].

Oracle, 2022. *what-is-a-relational-database.* [Online]
Available at: https://www.oracle.com/uk/database/what-is-a-relational-database/
[Accessed 20 November 2022].