

Assignment 1

DBAS6211



March 20, 2024

ST10393280

Daniel Luke James

Table of Contents

[Question 1 2](#_Toc161864298)

[Question 2 4](#_Toc161864299)

[Q.2.1 4](#_Toc161864300)

[Q.2.2 4](#_Toc161864301)

[Question 3 5](#_Toc161864302)

[Question 4 6](#_Toc161864303)

[Reference List 9](#_Toc161864304)

# Question 1

(appdynamics, 2024).

(shine, 2023).

(interviewbit, 2024).

# Question 2

## Q.2.1

Choosing a relational database is what I would advise as relational databases are the best option for storing course-related data, like the title of the course, the creator's name, the category, and other comparable features because they are made to hold organized data with predetermined relationships between entitys. The relational databases use constraints like unique constraints, foreign keys, and primary keys to maintain data consistency and integrity. These measures help guarantee that data is reliably stored and quickly retrievable (googlecloud, 2024).

## Q.2.2

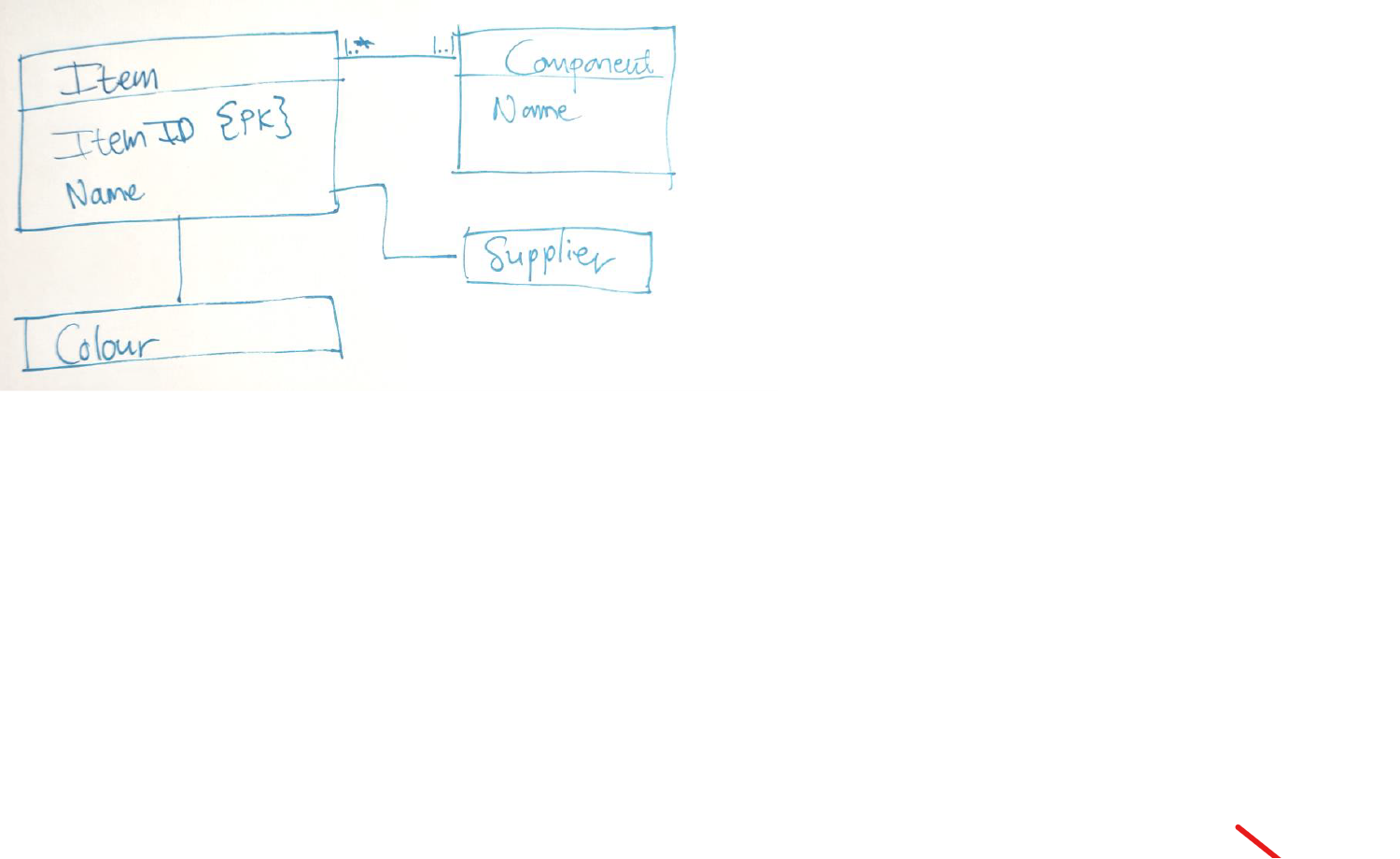
A NoSQL database, particularly one that is document-oriented, is what I would advise using. Files, images, and other data that is neither structured nor unstructured can be stored in NoSQL databases. This kind of data is especially well suited for document-oriented databases, which store data in an adapentity and schema-less fashion. Complex and hierarchical data structures can be stored in document-oriented databases thanks to their use of a document format that resembles JSON. This facilitates the storing and retrieval of documents, images, and videos associated with a particular course. Furthermore, document-oriented databases are extremely scalable and capable of managing massive amounts of data, a crucial feature when working with multimedia files, which can often be enormous (mongodb, 2023).

# Question 3

# Question 4

**10 recommended changes that align with the business rules:**

1. Replace the entity name from "Item" to "Product", this adjustment reflects the business rules that states that different components are required at different stages of the production process for each product.
2. To ensure that each product in the database has a unique identity, add the attribute "ProductID" to the "Product" object.
3. Establish a many-to-many relationship between the "Component" and "Product" entities, this adjustment reflects the business rules that a component can be used in multiple different goods and that each product requires its own part during the production process.
4. The database will always include the quantity of each component used for a certain product due to the addition of the attribute "Quantity", which is related to the "Component" and "Product" entities.
5. Add the attribute “ComponentID” to the “Component” entity, with this adjustment, the database will guarantee that every component has a distinct identifier.
6. Establish a relationship between the "Component" and "Supplier" entities. This adjustment considers the requirement that each component must be purchased from a specific supplier.
7. To ensure every supplier's name and address are retained in the database, add the attributes "SupplierID", "SupplierName" and "Address" to the "Supplier" entity.
8. Establish a many-to-many relationship between the "Supplier" and "Component" entities, this adjustment reflects the business rules that a supplier can provide a large variety of components.
9. Establish a one-to-many relationship between the "ProductionLine" and "Product" entities, this adjustment reflects the organization's policy that each production line produces exactly one type of product at a time.
10. To make separation easier, assign a colour to each production line according to business rules, the adjustment can be made by adding the attribute "Colour" to the "ProductionLine" entity.

**Improvements to implement the design in a relational database:**

**E**

**D**

**C**

**B**

**A**

1. Add a primary key to the "Product" entity, each entity entry has a unique identifier known as a primary key, which may be used to link similar entities together and make sure that no record is duplicated. I suggest making the "ProductID" field the entities primary key.
2. Make sure the "Component" entity has a primary key, I suggest that a "ComponentID" field must be included in this database as the primary key.
3. The "Supplier" entity needs a primary key added which is absent from the "Supplier" entity, which can compromise data performance and integrity. I suggest making the "SupplierID" field the entity's primary key.
4. For the relationship between the "Product" and the "Component", create a new entity “Product\_Component”. It is not optimal for a relational database since the ERD only displays a many-to-many relationship between "Product" and "Component" currently. I suggest creating a new entity to show the relationship between the product and component to fix this by including "ProductID”, "ComponentID" and "Quantity" may be present in this entity.
5. Rename the entity “Colour” to "ProductionLine" and add the “Colour" attribute to the "ProductionLine" entity and the primary key “ProductID”.

**A diagram of a product

Description automatically generatedThato’s final Entity Relationship Diagram after the following improvements and changes mentioned above:**

# Reference List

* appdynamics. 2024. What is Database Management Systems (DBMS)?, 20 February 2024.

[Online] Available at:

<https://www.appdynamics.com/topics/database-management-systems>

[Accessed on 12 March 2024].

* shine. 2023. What Are the Advantages of Database Management System?, 30 November 2023.

[Online] Available at:

<https://www.shine.com/blog/advantages-of-database-management-system>

[Accessed on 12 March 2024].

interviewbit. 2024. Components of DBMS (Database Management System), 8 January 2024.

[Online] Available at:

<https://www.interviewbit.com/blog/components-of-dbms/>

[Accessed on 12 March 2024].

* googlecloud. 2024. What is a relational database?

[Online] Available at:

<https://cloud.google.com/learn/what-is-a-relational-database#:~:text=The%20main%20benefit%20of%20the,access%20to%20related%20data%20points>.

[Accessed on 13 March 2024].

* mongodb. 2023. NoSQL vs. SQL Databases, 17 March 2023.

[Online] Available at:

<https://www.mongodb.com/nosql-explained/nosql-vs-sql#:~:text=popular%20programming%20languages.-,What%20are%20the%20benefits%20of%20NoSQL%20databases%3F,for%20developers%20to%20work%20with.&text=NoSQL%20databases%20typically%20have%20very%20flexible%20schemas>.

[Accessed on 13 March 2024].