Collabits Journal
Vol. 1 No. 2 May 2024 : 102-104
E-ISSN : 3046-6709, P-ISSN : 3062-8601
https://publikasi.mercubuana.ac.id/index.php/collabits

Entity-Relationship Diagram Technique in Database

Owen Palinggi[1*], Siti Maesaroh[2], Muhamad Bagas Permana[3], Darul Fitahul Huda[4], Kus Andi Priyono[5]

[1,2] Computer Science, Universitas Mercu Buana, Indonesia
[3,4,5] Informatics Engineering, Universitas Raharja. Indonesia
.
*Coressponden Author: owenpalinggi25@gmail.com

Abstract - The Entity Relationship Diagram (ERD) is a crucial data modeling technique in database design. ERDs are used to represent data conceptually before being implemented into the actual database schema. Creating a good and correct ERD becomes critical because errors in designing ERDs can cause serious problems in database implementation. By developing a deep understanding of the concepts and rules in creating ERDs, as well as considering the criteria for a good and correct design, developers can produce accurate ERDs that align with the requirements of the database structure to be built.

Keywords :
Entity Relationship Diagram;
Database;
Attributes;

Article History:
Received: 03-02-2024
Revised: 14-03-2024
Accepted: 16-04-2024

Article DOI : 10.22441/collabits.v1i2.27252

1. INTRODUCTION

The Entity Relationship Diagram (ERD) is one of the most important data modeling techniques in the database design process. ERDs are used to represent data in a conceptual form before being implemented into the actual database schema. By using an ERD, developers can illustrate the relationships between entities, the attributes possessed by each entity, and the types of relationships that occur between these entities.

The use of ERDs in database design has several main benefits. First, ERDs help to accurately and consistently model data requirements before the implementation stage. Second, ERDs provide a clear visual representation of the data structure, facilitating communication and understanding among team members. Third, ERDs serve as the basis for designing an efficient and well-structured database schema.

Despite being an essential technique, many developers still struggle with creating accurate and appropriate ERDs. Errors in designing ERDs can cause serious problems in database implementation, such as data redundancy, anomalies, or even data loss. Therefore, a deep understanding of the concepts and rules in creating ERDs is crucial.

In this journal, an example of a correct ERD will be presented along with a detailed explanation of each entity, attribute, and relationship involved. This discussion is expected to provide guidance for readers in designing accurate ERDs that meet the requirements, thereby producing an efficient database.

2. METHODOLOGY

The Entity Relationship Diagram (ERD) is a conceptual data model used in database design. The ERD depicts the relationships between data objects (entities) and the attributes possessed by each entity. Using an ERD, developers can model data conceptually before implementing it into the actual database schema.

The main components in an ERD consist of:

a. Entities: Entities represent objects or concepts whose data needs to be stored in the database. Entities are depicted with rectangles and named according to the object they represent.
b. Attributes: Attributes are properties or characteristics that describe an entity. Each entity has one or more attributes that serve as descriptors for that entity. Attributes are written inside the entity rectangle.
c. Relationships: Relationships represent the connections between two or more entities. Relationships are depicted with lines connecting the related entities and are named according to the type of relationship they represent.

In modeling ERDs, there are several common types of relationships used, including:
One-to-One (1:1)
One-to-Many (1:N)
Many-to-Many (N:N)

In addition, ERDs also use special notations to represent concepts such as key attributes, weak entities, specialization/generalization, and others. A good

understanding of the notations and rules in ERDs is crucial for producing an accurate and appropriate data model.

3. RESULTS AND DISCUSSION

In creating a good and correct Entity Relationship Diagram (ERD) design, there are several important criteria that must be met. The ERD is the crucial initial step in the database design process, so the accuracy and quality of the ERD design will significantly influence the quality of the resulting database.

Criteria for a Correct ERD Design

The following are the criteria that must be met for an ERD design to be considered good and correct:

The entities in the database are interconnected by relationships. In an ERD, every existing entity must have a relationship or connection with other entities. This relationship illustrates how these entities are related to the database to be built.

Each entity has attributes. Attributes are characteristics or properties that describe an entity. Each entity in the ERD must have at least one attribute that explains the entity.

Each entity has a primary key and descriptive attributes. The primary key is an attribute that functions as a unique identifier for an entity, while descriptive attributes are other attributes that describe the entity in more detail. Each entity in the ERD must have an appropriate primary key and descriptive attributes.

Relationships between entities are clearly depicted using the correct notation. Relationships in an ERD must be depicted with a diamond shape and filled with a verb that describes the relationship between the related entities. Additionally, the degree of cardinality (one-to-one, one-to-many, or many-to-many) must also be clearly shown for each relationship.

Use consistent notations and symbols. When depicting ERDs, the use of notations and symbols must be consistent with the applicable standards or rules. This facilitates the readability and understanding of the diagram.

By meeting the above criteria, the resulting ERD design will be conceptually good and correct, and can accurately represent the data requirements and structure of the database to be built.

Example of a Correct ERD Design

The use of the entity-relationship diagram (ERD) technique is no longer a foreign concept for database designers. In an ERD, all entities, attributes, and relationships must be designed completely and in detail. An ERD design can be considered good and correct if:

1. The entities in the database are interconnected by relationships.
2. Each entity has attributes.
3. Each entity has a primary key and descriptive attributes.

An example of a good and correct ERD design is as follows:

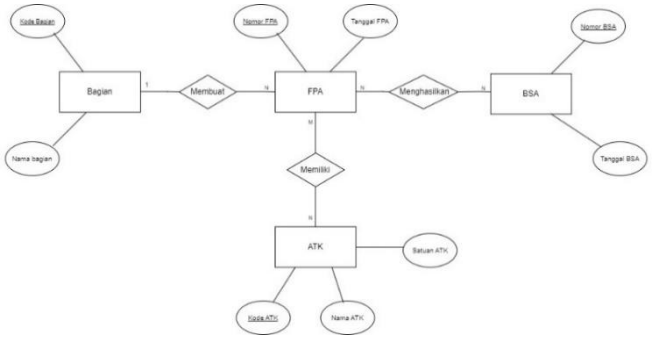

Figure 1. Example of a Correct ERD Design

In the image above, each related entity is connected by a relationship and there are lines connecting the entities and relationships. Each entity in the image has attributes that are outside the diagram, consisting of a primary key (denoted by an asterisk) and descriptive attributes.

For example, the "Creates" relationship connects the "Department" entity with the "FPA" and "ATK" entities through the use of lines. The "Department" entity has the "Department Code" attribute as the primary key, and "Department Name" as a descriptive attribute. The "FPA" entity has the "FPA Number" attribute as the primary key, and "FPA Date" as a descriptive attribute. Meanwhile, the "ATK" entity has the "ATK Code" attribute as the primary key, and "ATK Name" and "ATK Unit" as descriptive attributes.

Furthermore, the "Creates" relationship also connects the "FPA" and "ATK" entities with the "BSA" entity. The "BSA" entity has the "BSA Number" attribute as the primary key, and "BSA Date" as a descriptive attribute.

Thus, this ERD design has fulfilled the three criteria mentioned, namely that entities are interconnected by relationships, each entity has attributes, and each entity has a primary key and descriptive attributes. This indicates that this ERD design is good and correct according to the concepts and rules in creating ERDs.

Example of an Incorrect ERD Design

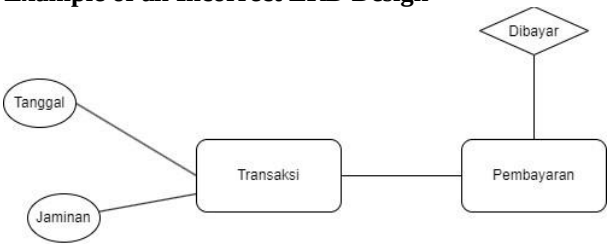

Figure 2. Example of an Incorrect ERD Design

In the given ERD image, there are several conceptual errors that reflect the creator's lack of understanding of the components in an ERD. One obvious mistake is the absence of relationships, which should be depicted with a diamond shape and contain a verb describing the relationship between entities. However, in this image, the relationships are not depicted correctly and do not contain any verbs.

Another error is the incorrect placement of attributes. Attributes should be inside the entities, but in this image, "Paid" which should be an attribute of the "Payment" entity, is instead placed on the relationship. This shows that the ERD creator cannot distinguish between attributes and

relationships.

Additionally, there are no connecting lines in the image indicating the relationships between the existing entities. Each entity that has a relationship should be connected by a line that meets at the relationship diamond. However, in this image, no connecting lines are visible.

Finally, there is no explanation of the degree of cardinality (one-to-one, one-to-many, or many-to-many) in the relationships between entities. The degree of cardinality is crucial for indicating the type of relationship and constraints between entities in the ERD.

These errors reflect the ERD creator's lack of understanding of the basic concepts in creating ERDs, such as entities, attributes, relationships, and degree of cardinality. This can lead to an inaccurate ERD that cannot properly represent the data structure.

Solving Difficulties in Creating ERD

In designing an ERD, there are several important steps that must be considered to ensure accuracy and alignment with the requirements of the database to be built. The first step that must be taken is to identify the entities involved in the system or application to be developed. Entity identification can be done by finding nouns that indicate objects, people, places, processes, organizations, or relevant concepts.

After the entities have been identified, the next step is to determine the attributes possessed by each of these entities. Attributes are characteristics or properties that describe an entity. In designing an ERD, attributes are depicted with ellipses connected to the corresponding entity.

One of the main challenges in designing an ERD is determining the relationships or connections between the existing entities. These relationships are crucial for illustrating how entities are interconnected within the system or application being built. Relationships are depicted with a diamond shape and typically use a verb to describe the relationship.

Additionally, determining the degree of cardinality or type of relationship between entities (one-to-one, one-to-many, or many-to-many) must be done carefully. Errors in determining the degree of cardinality can lead to an inaccurate data structure.

In depicting an ERD, the use of appropriate colors should also be considered. Bright and contrasting colors can help improve the readability of the diagram and facilitate the identification of its components. Furthermore, the size of the diagram must be adjusted so that all components can be clearly and proportionally depicted.

After designing the ERD, the final step is to analyze the diagram again to ensure there are no errors or omissions. A well-designed ERD will result in a detailed, well-specified database that can support the development of the desired application or system.

4. CONCLUSION

The Entity Relationship Diagram (ERD) is a crucial data modeling technique in database design. ERDs are used to represent data conceptually before being implemented into the actual database schema. Creating a good and correct ERD becomes critical because errors in designing ERDs can cause serious problems in database implementation.

For an ERD design to be considered good and correct, there are several criteria that must be met:

1. Each entity must be interconnected with other entities through relationships.
2. Each entity must have attributes, including a primary key and descriptive attributes.
3. Relationships between entities are clearly depicted using diamond notation and appropriate verbs.
4. The degree of cardinality (one-to-one, one-to-many, many-to-many) is clearly shown for each relationship.
5. The use of notations and symbols must be consistent, following applicable standards.

By developing a deep understanding of the concepts and rules in creating ERDs, as well as considering the criteria for a good and correct design, developers can produce accurate ERDs that align with the requirements of the database structure to be built.

REFERENCE

[1] ZF Azzahra, AD Anggoro, "Analysis of Entity-Relationship Diagram Technique in Database Design: A Literature Review" JURNAL INTECH, VOL.3, NO.1, MAY 2022.

[2] SM Pulungan, R Febrianti, T Lestari, "Analysis of Entity-Relationship Diagram Technique in Database Design" December 2022 Journal of Economics, Management and Business (JEMB) Vol.1 No.2 July-December Edition Pages 143-147.

[3] M Larassati, A Latukolan, A Arwan, MT Ananta, "Automatic Mapping System Development of Entity-Relationship Diagram Into Database" Journal of Information Technology and Computer Science Development, Vol. 3, No. 4, April 2019, pp. 4058-4065.

[4] IS Akbar, "Development of Entity Relationship Diagram Database for Ira Online Store Surabaya".

[5] IS Akbar, T Haryanti, "DEVELOPMENT OF ENTITY RELATIONSHIP DIAGRAM DATABASE FOR IRA ONLINE STORE SURABAYA" Scientific Journal of Computing Insight Vol.3 No.2 | 2021,.

[6] R Nurmasari, S Pinem, U Nurkhalifah, "Design of Nusantara Fishery Port (PPN) Data Management at Pelabuhan Ratu Using Entity Relationship Diagram (ERD)" Journal of Information System Engineering and Management, Vol. 9, No. 1, February 2023, pp. 52-57,.