Applied structured database in a small project

To cite this article: F Y Arini et al 2019 J. Phys.: Conf. Ser. 1321 032130

View the article online for updates and enhancements.
Applied structured database in a small project

F Y Arini*, R Arifudin and M Aris
Department of Computer Science, Faculty of Mathematics and Natural Sciences
Universitas Negeri Semarang, Semarang, Indonesia

*Corresponding author: floyuna@yahoo.com

Abstract. The growth of software application today has a lot of variety. The ease for software developers which provide an interest for all is to do self-taught learning. However, not all application developers are able to apply database theory in developed database systems. Database development needs to be normalized. Famous theoretical normalization for the database is known as BCNF Boyce-Codd normal form. This theory is actually very easy to understand but in the execution, application developers for small project application do not use the theory in the implementation of the database in their system. One of the reasons is the lack of understanding of normalization process. The simplest way to verify database normalization in a small project is with the use of ERD (Entity Relationship Diagram). This can be known by the same representation of the number of entities in the ERD by the number of tables in the database schema.

1. Introduction
With the rapid development of information technology [1], various types of software that facilitate users to develop their own applications are also growing. Observer of this application developer is not only from people who have a computer skill background, but also everyone who has the interest and expertise gained from self-taught learning. With the internet technology [2], application development tutorials with various kinds of software make learning easier.

In addition, the development of information technology supported by increasingly good features and inexpensive supporting equipment triggers various institutions [3], government [4], schools [5] and corporations [6] to always improve in the recruitment of information technology experts to integrate their data and systems in order to run more effectively and efficiently. For example, in the banking sector [7], all banks provide online services in doing financial transactions; online stores [8] are able to make transactions to purchase and sell goods with online system without having to meet their customers; and employee recruitment [9] can also be done online.

Therefore, all existing data must be stored properly, for example in a bank, the data of customers who do the saving must be kept secure; online stores must have updated data on the prices of the goods they sell; or agencies that do recruitment must maintain the data validity of prospective employees. These data should be managed by database experts in order to speed up the process and decision making accurately, quickly and precisely.

Therefore, learning of the database technology development starts from institutions or educational institutions through teaching and learning activities, research and dedication to educate and train the resources that they have through teachers, staff and students. Moreover, learning of system or application integration with database technology has been developed through small projects written in scientific work both at national and international levels.
Unfortunately, the standardization of projects that integrate with the database cannot be applied. This situation occurs especially when students carry out the final project by doing simulation project or building application system that is integrated with the database. Their database conditions which integrate with project or application are usually not in normal conditions. This condition can be said as a normal condition when the database does not undergo any repetition and there is certainty of data dependencies (the data is in the right table).

2. Database

Database [10] is a data collection that is interrelated. This relation is indicated by the relationship between tables which is related to the key in the table. Each table has many attributes that are used to describe themselves as unique identities.

2.1. Terms in Database

In the database there are several terms that must be understood by database developers, including (a) Attributes that can be referred to as data elements, data fields, or data item used to explain detailed information of an entity; (b) Data value that is the contents of the actual data stored on the attribute; (c) An entity that is a collection of attributes that are unique; (d) Record/Tuple that is a collection of elements that are interconnected to inform detailed information about an entity. Each record usually represents one data or information; (e) File that is a collection of similar records with the same element length, the same attributes, but with different data values.

2.2. Criteria of Database

Database development should have the following criteria: (a) Data stored does not experience repetition or redundancy to prevent anomalies and save space; (b) Databases can be utilized by many applications without having to change their structure; (c) Database security is prioritized; (d) Relationships between tables must be clear; (e) Database is easy to manage.

2.3. Objective of the Database

Database development has the following objectives: (a) The use of databases makes the speed of data access in processing large amounts of data more efficient; (b) Data accuracy for large amounts of data using a database more structured; (c) The sharing of data for different applications can be sharebility; (d) Eliminating repetition of data; (e) Providing data flexibility.

2.4. Component of Database

Components as supporting databases include (a) database support hardware that is usually in the form of computer devices that have the specification standards of minimum storage media required for the database; (b) Operating System that is a system software that regulates the performance of hardware and basic syntax operations including programs integrated with the computer; (c) Database: a collection of structured data from various types of facts stored in a storage medium that is represented in the form of a schema; (d) DBMS stands for DataBase Management System. DBMS is a system used to manage large amounts of data supported by data performance and integrity that can operate independently so as to produce data that is in accordance with the user's request; (e) Users that are people who use database facilities so they can insert, update or delete data; and (f) Database Developer Software that is a software that allows database developers to create databases.

3. Normalization

Database normalization [11] is a systematic approach used in building logic database relation design by applying a number of standard rules and criteria to minimize data redundancy in a database so that the database is able to work optimally. This design logic database can be defined by using functional dependencies to state normalization rules. The objective of normalization is to produce a normal structure. The table structure can be said to be in the normal form if the rules imposed on relations or
tables in the database meet the stages at normalization levels. The stages of normality of a database include the form of a database that is not normal (still according to pure data from field data), the 1st Normalize Form or called 1NF, the 2nd Normalize Form or called 2NF, the 3rd Normalize Form or called 3NF, and the Boyce-Codd BCNF form.

3.1. **Unnormal Form**

Unnormal Form is a collection of data that will be processed without having to follow a certain format. Data in this form is obtained from a variety of formats so that there is still duplication of data because the data is in accordance with field facts that are sometimes imperfect or incomplete. In relational databases there should be no duplication of data because it can affect anomalies. This anomaly occurs if in one line/tuple there is data to be deleted so the consequences are other data may lost.

3.2. **1st Normalize Form**

The first normal level or also called 1 NF if each attribute only has a single value of record. In other words, data in an abnormal condition must be changed to the first normal form by creating a row containing the same number of columns with each column containing only one value. This is to confirm that each field contains only one meaning and is not a data set that has a double meaning.

3.3. **2nd Normalize Form**

In the design of relational databases there should be no partial functional dependency to the primary key, because it can have an anomaly. Therefore the first normalization stage will produce a second normal form or what is called as 2 NF. It is considered as 2 NF form if and only if the relation in the database meets the criteria of the 1st Normalize Form. Moreover, each of non-key attribute in 2 NF must functionally dependent on with the Primary key. So that when the 2nd Normalize Form has been formed then the key in the defined field has also been determined.

3.4. **3rd Normalize Form**

Anomalies can occur if in a relational database design there is a transitive functional dependency. The functional dependency transitive occurs because there is a functional dependence of an attribute on another attribute through another attribute as well. So that the normalization of the third form or often referred to as 3 NF is a database relation that meets the rules of 2 NF form and all of the attributes are not primary keys that do not have a transitive functional dependency relationship to the primary key. This means that each non-key attribute must depend only on the primary key as a whole, and the third form of normalization already has an optimal table.

The 3NF form is a database normalization technique that has a very close relationship with the normal form of BCNF Boyce-Codd form. At this stage of normalization, BCNF or 3NF is considered sufficient to minimize problems in database design (redundancy, lossless, dependency, and preservation).

4. **ERD**

ERD [12] is a diagram technique used to model data during system development. This technique provides a basic description for carrying out the relational database design specifications that underlie the information system developed.

4.1. **Components of Entity Relationship Diagram**

ERD has 4 main components: (a) Entity, an object that is a representation of a table, which is usually referred to as a single noun and is represented as a rectangle; (b) Relation that comes in the form of a rhombus to express the relationship of two entities connected by a straight line; (c) Attributes, that provide more detailed information about the details (characters) of the entity and can be optional; (d) Outline that serves as a link between relations and entities or relations and entities with attributes.
4.2. Cardinality/Degree of Relations
Cardinality or degree of relation is used to describe the maximum entity relationship in a set that can relate to entities in other sets and vice versa. Cardinality relations that occur between two sets of entities can be: (a) One to One relation: each entity in entity X is only related to one entity in entity Y and vice versa. (B) One to Many relations: each entity in entity X can have a relationship with many entities in entity Y, but not vice versa. (c) Many to One relation: each entity in entity X can have a relationship with at most one entity in entity Y, but not vice versa. (D) Many to many relations: each entity in entity X can relate to many entities in entity Y and vice versa.

5. Discussion
The steps in implementing this normalization are indeed used as a first step to understanding the development of a database so that data repetition does not occur. However, when this normalization process is applied to a project that has case studies, its application becomes more difficult. To that end, understanding is to make it easier to normalize by implementing Entity Relationship Diagrams (ERD).

To find out more clearly how the application of ERD in the database concept, it can be seen in the sample of the small project that was done in the final project in case study basis. The samples presented were ERD and Database Schema applied in Schedule simulation for Department of Computer Science at the Universitas Negeri Semarang, Indonesia.

5.1. ERD for Schedule Simulation
The detail relation description of ERD for Schedule Simulation in the Department of Computer Science can be seen in Figure 1.

![Figure 1. ERD](image)

The business rules that can be seen in Figure 1 are:
1. Each lecturer is able to teach more than one course and each course can be taught by many lecturers so that many to many relationships arise. If there is many to many relation between those two entities there must be a new entity whose attributes consist of foreign keys originating from these two entities (Figure 2).
2. Each lecturer in a particular subject has 1 status that is as Chairperson of the Lecturer or becomes Assistant of Lecturer. If there is only one lecturer, it means that the lecturer is the Chairperson of the lecturer who is considered as a single lecturer.
3. The lecture_course entity can have many teaching schedules influenced by semester entity and room entity.
5.2. Database Schema for Schedule Simulation

Database Schema of the Schedule Simulation in Department of Computer Science at the Universitas Negeri Semarang in Figure 2 that is a detailed description of the entity in the ERD for the Schedule Simulation that can be seen in Figure 1. By using the database scheme, relations between tables can be figured out by the linking the keys between the tables. There are seven tables for the detail description of Database Schema of the schedule simulation and each table have their relation. Every lec

Lecturer table represents detailed information of lecturer's data including lecturer_id, lecture_ein (ein: employer id number) and employer_name. Course table consists of course_id, course_name, ucu (university credit unit), course_type (theory or practice), course_practice_code. Therefore, these two tables have many to many relationship. Due to that condition, there must be a new table between those two tables called lecturer_course table consisting of keys from lecturer table and lecturer course as the foreign key. Lecturer_course table also includes status_id as a foreign key from the table status which represent status of lecturer as a chairperson or assistant and schedule_id from schedule table. Furthermore, schedule data are stored in schedule table containing foreign key from semester, room and course tables.

6. Conclusion

Comparison of the two examples presented in this paper illustrates how ERD is the most important part from the final results of the normalization process applied in the formation of database structures. Explanations and examples in this paper can facilitate software application developers in understanding the database normalization process even though software-based application developers do not have specific computer background like informatics, computer science or information systems.
so they should have model databases following the rules applied in the database structure. The easiest way after normalization is to see relations between tables by matching entity relationships in ERD, especially for small projects that are usually developed by students in a thesis or dissertation to stimulate their small project. The term of normal condition occurs if database entities and schema have the same number of expectations.

References

[1] Fichman R G and Carroll W E 2004 JAIS 5 314
[2] Charland A and Leroux B 2011 Mobile application development: web vs. native (New York: Queue)
[3] Thompson L, Jeffries M and Topping K 2010 J Innov Educ Teach Int 47 305
[4] Bertot J C, Jaeger P T and Grimes J M 2010 J Gov Inf Quart 27 264
[5] Wills H P and Mason B A 2014 J Behav Edu 23 421
[6] Baumann-Pauly D and Scherer A G 2013 J Bus Ethics 117 1
[7] Martins C, Oliveira T and Popović A 2014 Int J Manag Soc Sci 2 21
[8] Kourmpouroulis D N 2013 Int J Econ Manag Soc Sci 2 21
[9] Parry E and Tyson S 2011 HRMJ 21 335
[10] Elmasri R and Navathe S B 2010 Fundamentals of Database System (Massachusetts: Addison-Wesley)
[11] Verma S 2012 IJREAS 2 59
[12] Thalheim B 2000 Entity-Relationship Modeling: Foundations of Database Technology (New York: Springer)