

An Experimental Comparison of ER and UML Class Diagrams

Ahmad Al- Shamailh¹

¹ Mutah University, Jordan
Ahmadsham013@gmail.com

Abstract

A data model is a set of concepts that can be used to describe the structure and operations of the database. It represents as a diagram that illustrates all the possible relationships between data elements. Many models are used to represent data. The most famous of these models are ER model and UML class diagrams. Each model has many uses and features different from the other model. The database designers when choosing the right model to represent their data is a hard task for them. In this paper, we have applied an experiment to compare the ER model with UML class diagram. The task of experiment is to determine the strengths and weaknesses points of each model. It is based on matching a given textual specification for a simple application domain against collection of diagrams. The results indicate that ER model is generally better in relationship name and participation representation than the UML class diagram. On the other hand, UML class diagrams provide a better representation for the cardinality relationship compared with ER diagrams.

Keywords: UML class diagram, ER diagram, notation

1. Introduction

Many graphical notations may used to represent data model. The Entity relationship (ER) notations and its extension are the most used notations for database conceptual modeling. In addition, the Unified Modeling Language (UML) class diagram can be used to represent the conceptual schema of a system.

There are differences between ER and UML class Diagrams. Each model has specific uses. On the other hand, the notations that used in ER and UML class diagrams are different from each to other. There are also many advantages and disadvantages for each model. Therefore, the database designers when choosing appropriate model is not easy task for them.

The aim of this research was to present an experimental study for comparing between the ER and UML class diagrams. The purpose of experiment is to know the strength and weakness points of each model. The experiment of subject revolves around matching textual specifications for a particular system with a set of diagrams. The comparison also includes individual notational differences for each model.

The rest of the paper is organized as follows; Section 2 present UML and ER background and including the notations. Section three gives details of the design of the experiment, including the experiment materials, experiment diagrams, and experimental procedure and finally, reports and discusses the achieved results. Related work with our study in Section 5. Section six the limitations, followed by conclusion in section 7.

2. ER and UML Background

The Entity-Relationship(ER) is one of the methodologies for designing relation database. It is illustrate relationship between entities in database. The model developed by Peter Chen and

published in a 1976 [1]. It is containing entities, relationships between them, and attributes of the entities.

The Unified Modeling Language (UML) is a standard language for specifying the design of the system. It is developed by James Rumbaugh, Ivar Jacobson and Gray Booch at Relational Software during 1994-95[2]. It is standard for programming in object oriented programming languages.

2.1. The Notations

The notations that used in ER and UML class diagrams are different from each to other. In ER notation, an entity is represented as rectangle containing the entity name. The attributes are represented as eclipses. In the UML notation, the entity represent as object. The object is an instance of class. The class is represented as rectangle with up three parts. The first part includes the name of the attribute. Part two contains the attributes of class with type and visibility. Part three to describe the method performed by the class.

The cardinality of relationship is representation in UML notation as the symbol used for unlimited cardinality is “*” and or in combination with another low value (like “1…*”) that meaning mandatory relationship. In ER notation, the cardinality is represented as one or N/M and mandatory relationship represent as double link but partial relationship represents as single link. The name of a relationship is specified directly on the relationship line in UML notation. It is displayed inside diamond-shaped in ER notation.

Figure 1 show a simple relationship between two entities is represented using ER and UML class diagram notations .The relationship between the car and its owner. Every car has at least one owner. One person may own several cars or no cars.

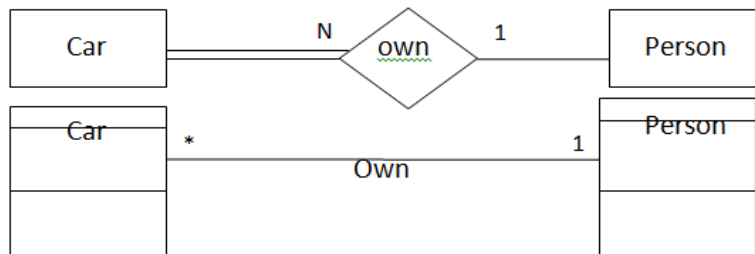


Figure 1. The Example using The ER and UML Class Diagrams Notations

3. Experimental Method

The experiment is based on matching a given textual specification for a simple system against collection of diagrams, indicating whether each diagram correctly matches specification. The collection of diagrams includes correct and incorrect diagrams that represented using the ER and UML class notations.

3.1. Experimental Materials

3.1.1. The Application Domain: The experimental diagrams are based on a simple system, which models a small college. A college includes many departments. Each department offer courses teaching by instructors. The students can enroll number of courses. The example includes four entities and five relationships (see Figure 2).

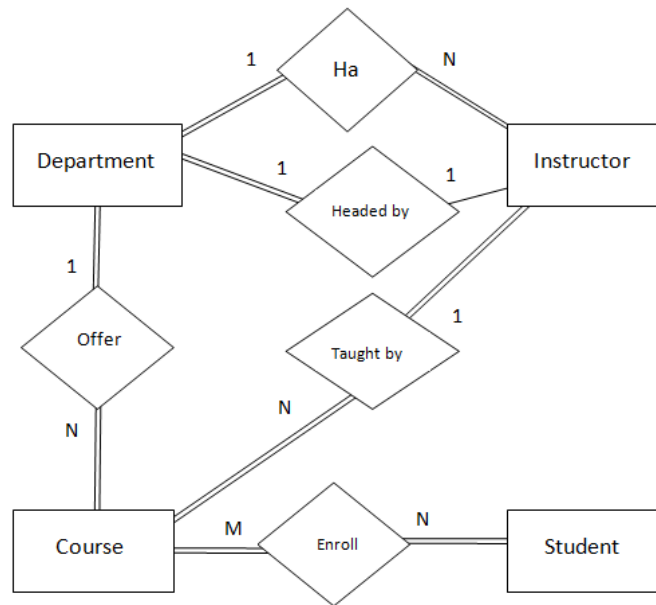


Figure 2. The Experimental Application Domain in ER Notation

3.1.2. Worked Example: Figure 3 show a small textual specification of the college system. It is illustrate a general description for entities and the relationships between them. The attributes for the entities are not mentioned to reduce the complexity of the diagrams and focusing on the content of the experience.

The college system

A college includes many departments. Each department offer courses teaching by instructors. The students can enroll number of courses.

A Department:

- Must offer at least one course.
- Must be headed by one instructor.
- Must be worked for by at least one instructor.

A Student:

- Must enroll at least one course.

A Course:

- Must be taken by only one instructor.
- Must have number of students.

An Instructor:

- Must work for one department.
- May manage one department.
- must take at least one course.

Figure 3.The Specification for College System

3.1.3. The Experimental Diagrams: The experiment was required different diagrams in performing comparison task. The ER and UML notations were used to draw the experimental application domain. The diagrams include correct and incorrect diagrams. Three errors were applied on the correct diagrams. The total of diagrams is eight diagrams that using in the experiment. The errors affected on the cardinalities and relationships names and the constraints. The errors are applied on the correct diagrams, giving six incorrect diagrams.

Figure 4 show the experimental application domain is represented using UML notations in correct representation that used in the creation of the experimental diagrams.

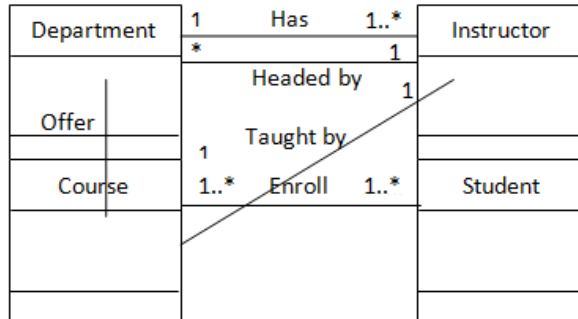


Figure 4. The Application Domain in UML Notation

Figure five show the error operation (a) is applied on the correct diagram for the experimental application domain using UML notations. The error affected on the cardinalities of relationships. The cardinality of relationship between the course and instructor entities are changed from one-to-many to many-to-many. In addition, the cardinality of relationship between the course and department entities are changed from one-to-many to many-to-many. The figure shows this change and a simple description of the error operation.

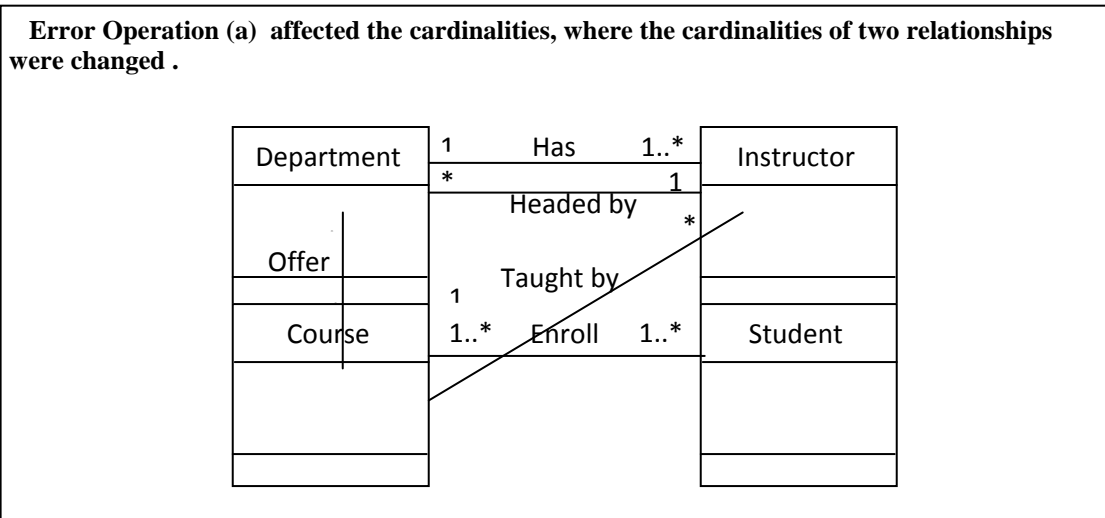


Figure 5. The Application Domain in UML Notation with Error

Figure 6 show the error operation (b) is applied on the correct diagram for the experimental application domain in UML notations. The error affected on the relationships names. The relationship name between the course and instructor entities (enroll) is swapped with the relationship name between the course and department (offer). The figure shows this change and a simple description of the error operation.

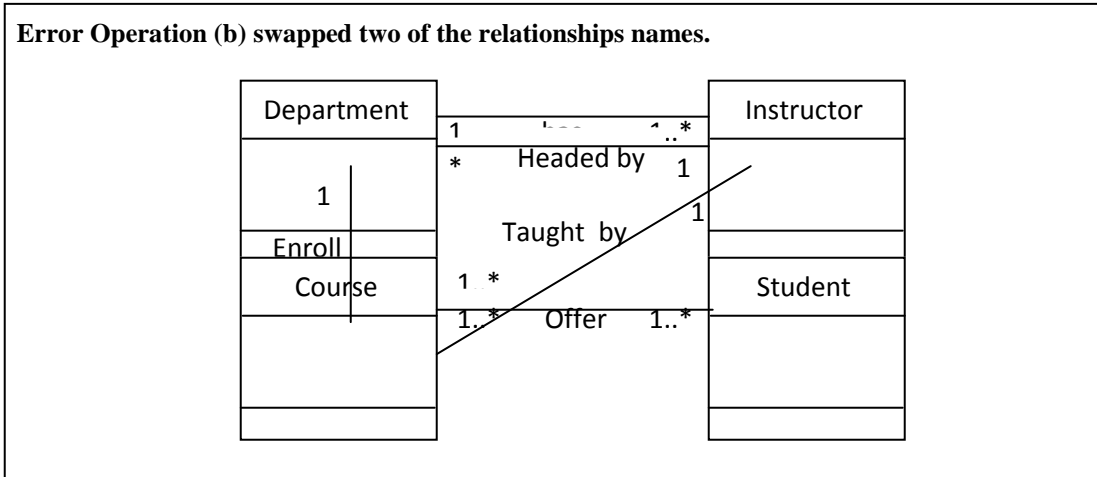


Figure 6. The Application Domain in UML Notation with Error

Figure seven show the error operation (c) is applied on the correct diagram for the experimental application domain in UML notations. The error affected on the constraints of two relationships. The participation of the department entity, which is associated with the instructor entity, is changed from total to partial. In addition, the participation of the course entity that is associated with the student entity is changed from partial to total. The figure shows this change and a simple description of the error operation.

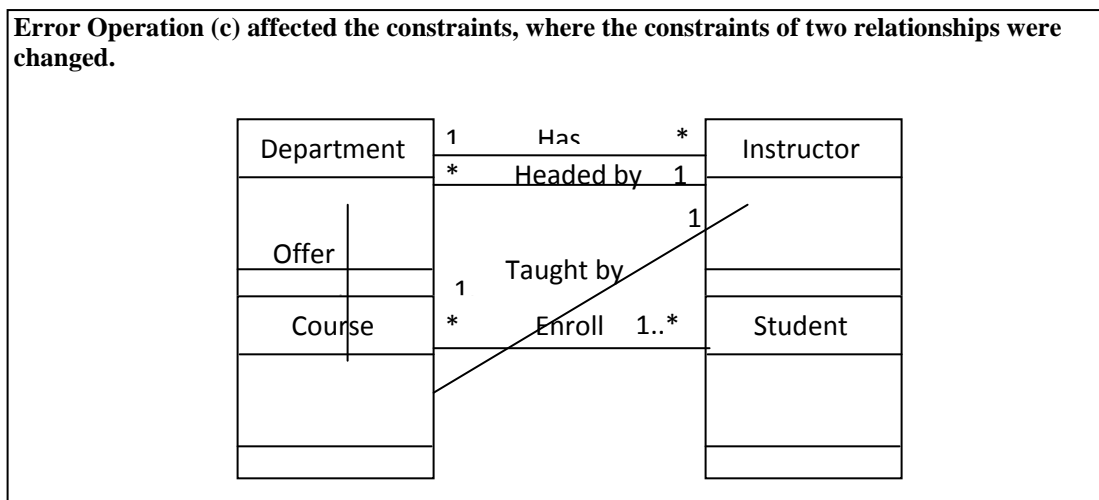


Figure 7. The Application Domain in UML Notation with Error

Errors operations have been applied separately in the experiment and studied independently. It is enabled to understand the national differences. It may be that one notation

is better for representing participation, while another may be better for representing relationship name.

4. Experimental Procedure

4.1. Preparation

The performed experiments involved students of the Utah University having backgrounds in ER and UML diagram. The number of students who participated in the experiment is 10 students. The students were given a copy of text specification and the experimental diagrams. They are answered correct or incorrect based on the textual specification. The students were also asked which of notations they preferred for representing the relationships name, cardinality and participation. They were also asked about the reasons of their preferences.

4.2. Collection Data

After the experiment is conducted, the data of experiment are collected and analyzed.

4.2.1. Performance Data: We were measured the number of errors from the answers of each student's responses. Analysis was performed on the student's average percentage performance in identifying the correct diagrams and their average percentage performance in identifying each set of incorrect diagrams.

Table 1 shows the errors average was calculated for each diagram. Four independent sets of data were collected; one for correct diagrams, and three for the three different error operations. Error percentages was measured percentages of the total depend on the answers of each student's responses.

Table 1. Errors Averages for All Experimental Diagrams

Experimental diagrams	Errors average
ER (correct)	0.125
UML (correct)	0.2
ER (cardinality error)	0.7
UML (cardinality error)	0.3
ER (participation error)	0.25
UML (participation error)	0.625
ER (relationship naming error)	0.3
UML (relationship naming error)	0.5

Figure eight show graph representing the values of the errors averages from Table 1.

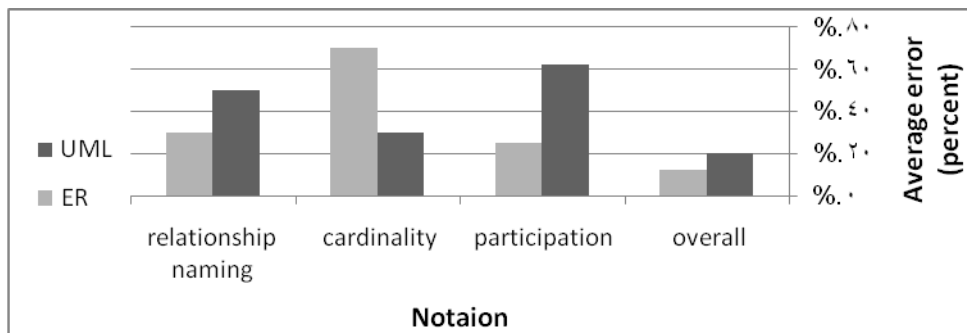


Figure 8. Graphic Representation for Errors Averages

4.2.2. Preference Data: We collected preference data from student. It was as follows:

Cardinality: The student's reasons for their choice of ER notation for the cardinality notation were that it was a simpler notation and clearer than UML notation. The student's reasons for their choice of UML notation for the cardinality notation were that it was an easily understood and reading.

Relationship Naming: The reasons the students gave for the UML relationship-naming notation being more easily understood and simpler than the ER notation. The student's reasons for their prefer of ER relationship naming notation is writing inside the shape diamond leads to highlighted the relationship.

Participation: The student's reasons for their choice of UML notation for the participation notations were that it was more easily understood and reading than the ER class diagram notation. The reason for the preferred the ER notation is the single/double lines were more pronounced than the symbol "*" or (one...*). The double lines are more indicative of mandatory participation.

Tables 2 show the number of students who prefer each model in the representation of the cardinality, participation and relationship name.

Table 2. Preferences (Number of Students) for each Notation

Notations	ER	UML
Cardinality preference	3	7
Participation preference	8	2
Relationship naming preference	6	4

Figure nine show graph representing the data in Table 2:

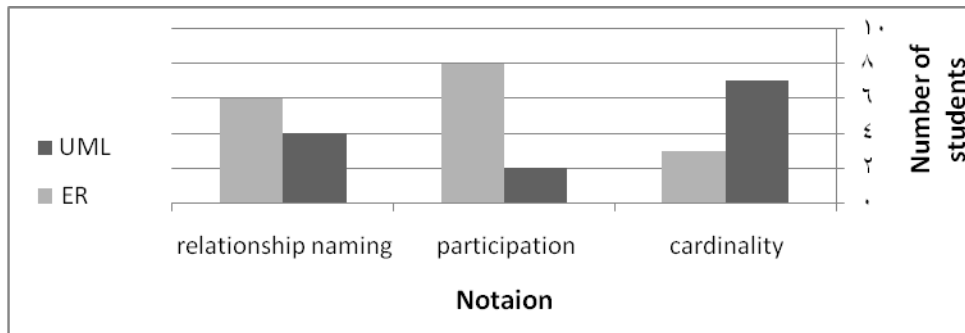


Figure 9. Graphic Representations for Number of Students

4.3. Analysis Results

Error average in ER diagram after changing the relationship name is less than error average in UML class diagrams. The number of students who prefer to represent the relationship name in ER notation greater than the number of students who prefer to represent the relationship name in UML notation.

Error average in ER diagram after changing the constraint is less than error average in UML class diagrams. The number of students who prefer to represent the constraint in ER

notation greater than the number of students who prefer to represent the constraint in UML notation.

Error average in UML diagram after changing the cardinality is less than error average in ER class diagrams. The number of students who prefer to represent the cardinality in UML notation greater than the number of students who prefer to represent the cardinality in ER notation.

In interpreting these results, we conclude that ER model is generally better in relationship name and participation representation than the UML class diagram. On the other hand, UML class diagrams provide a better representation for the cardinality relationship compared with ER diagrams.

5. Related Work

There are various empirical comparisons of data models between UML and ER models have been conducted in the past. In work [3] perform an empirical study for compare ER and UML diagrams from a maintainer perspective. Their results indicated that UML diagrams provide better supports during comprehension activities on data models. Biota and Gabriele [4] present comparing between ER and UML class diagrams during comprehension activities to identify weakness in a notation and/or justify the need of preferring ER or UML for data modeling. The result of study indicates that UML class diagrams are generally more comprehensible than ER diagrams.

There are also several empirical comparisons of other Design Models. Purchase et al. [5] compare the comprehension of the syntax of the two types of ER notations, named the Chen model and the SSDM notation the result of their empirical analysis the SSDM is better understood than the Chen notation. The SSDM notation is more concise than Chen notation with fewer shapes and text on the page. In [6] compare Extended ER (EER) and OO data models from the point of view of design quality. Their results indicated that the EER model surpasses the OO model for designing unary and ternary relationships.

There are others work about UML and ER models. Such as, in [7] focus on deriving ER diagram from database schema with little information for the attribute in its table and no information for keys. Through analysis of data, instances in database and the query and view statements in program code related to the database. In work [8] focus on extracting an UML conceptual schema and set of OCL, integrity constraints and derivation rules inferred from the database schema to facilitate the comprehension of the enforced rules and their evaluation in database.

There are many tool used to draw ER diagrams. First tool [9] present tool for converting the traditional diagram to new editor diagram to make the assessment process appropriate for semi-automated. The tool enables students to obtain information on each component in a conceptual database diagram. The paper solving the problem increased the number of sub-diagrams marked by the examiner. Other tool [10] presents an XML-based ER diagram drawing and translation tool. Named ER Draw. The tool aims to drawing ER and provide automatic translation to relational models automatically. The tool developed for educational purpose and available in public. Also in [11] present tool for automate the process of relational database normalization, named normalize. The tool is very useful an easy to used for database designer.

6. Limitations

Every the empirical study has limitations. In our study, the application of the comparison is constraint in a simple framework, only two notations and three differences. We have also

applied the experiment on the small number of students. In the future work, we will intend to apply the experiment on larger segment of students having different academic backgrounds and experience in ER and UML diagram. In addition, we will use domain that more complexity.

7. Conclusion

The choice between two diagram notations is not easy job with regard to database designers. Each diagram has characteristics different from the other model. On the other hand, each model is represented in different notations. Therefore, in this paper, an experiment is applied to compare two diagrammatic notations; the models are the ER and UML class diagrams. The comparison includes comprehension and individual notational variations. The achieved results demonstrated that that ER model is generally better in relationship name and participation representation than the UML class diagram. In addition, UML class diagrams provide a better representation for the cardinality relationship than ER diagrams. This experiment can be used to compare other diagram notations that would benefit from the empirical studies realization their comprehension.

References

- [1] Chen and P. Pin-Shan, "The entity-relationship model—toward a unified view of data", *ACM Transactions on Database Systems (TODS)*, vol. 1, no. 1, (1976), pp. 9-36.
- [2] J. Rumbaugh, I. Jacobson and G. Booch, "Unified Modeling Language Reference Manual", The Pearson Higher Education (2004).
- [3] A. De Lucia, C. Gravino, R. Oliveto and G. Tortora, "An experimental comparison of ER and UML class diagrams for data modeling", *Empirical Software Engineering*, vol. 15, no. 5, (2010), pp. 455-492.
- [4] G. Bavota, C. Gravino, R. Oliveto, A. De Lucia, G. Tortora, M. Genero, and J. A. Cruz-Lemus, "Identifying the weaknesses of UML class diagrams during data model comprehension", In *Model Driven Engineering Languages and Systems*, pp. 168-182. Springer Berlin Heidelberg (2011).
- [5] Purchase, Helen C., Ray Well and, M. McGill, and L. Colpoys. "Comprehension of diagram syntax: an empirical study of entity relationship notations." *International Journal of Human-Computer Studies* 61, no. 2, pp. 187-203(2004).
- [6] [6] Shoval, Peretz, and Sagit Shiran. "Entity-relationship and object-oriented data modeling—an experimental comparison of design quality." *Data & Knowledge Engineering* 21.3 ,pp. 297-315(1997).
- [7] [7] Yeh, Dowming, Yuwen Li, and William Chu. "Extracting entity-relationship diagram from a table-based legacy database." *Journal of Systems and Software* 81.5 ,pp.764-771(2008).
- [8] [8] Cosentino, Valerio, and Salvador Martinez. "Extracting UML/OCL Integrity Constraints and Derived Types from Relational Databases." *13th International Workshop on OCL, Model Constraint and Query Languages*, October (2013).
- [9] [9] Batmaz, Firat, and Chris J. Hinde. "A diagram drawing tool for semi-automatic assessment of conceptual database diagrams." *Loughborough University* pp.71-48(2006).
- [10] [10] Xu, Shuyun, Yu Li, and Shiyong Lu. "ERDraw: An XML-based ER-diagram Drawing and Translation Tool." *Computers and Their Applications*,pp.143-146(2003).
- [11] [11] Arman, Nabil. "Normalizer: A case tool to normalize relational database schemas." *Information Technology Journal* 5, no. 2 ,pp. 329-331(2005).
- [12] [12] Cruz-Lemus, José A., Marcela Genero, M. Esperanza Manso, and Mario Piattini. "Evaluating the effect of composite states on the understandability of UML statechart diagrams." In *Model Driven Engineering Languages and Systems*, pp. 113-125. Springer Berlin Heidelberg, (2005).
- [13] [13] Ricca, Filippo, Massimiliano Di Penta, Marco Torchiano, Paolo Tonella, and Mariano Ceccato. "The role of experience and ability in comprehension tasks supported by UML stereotypes." In *ICSE*, vol. 7, pp. 375-384(2007).
- [14] [14] Agarwal, Ritu, Prabuddha De, and Atish P. Sinha. "Comprehending object and process models: An empirical study." *Software Engineering, IEEE Transactions on* 25, no. 4 ,pp.541-556(1999).
- [15] [15] Mfourga, N. "Extracting entity-relationship schemas from relational databases: a form-driven approach." In *Reverse Engineering, 1997. Proceedings of the Fourth Working Conference on*, pp. 184-193,IEEE(1997).
- [16] [16] Agarwal, Ritu, Prabuddha De, and Atish P. Sinha. "Comprehending object and process models: An empirical study." *Software Engineering, IEEE Transactions on* 25, no. 4, Agarwal, Ritu, Prabuddha De, and

Atish P. Sinha. "Comprehending object and process models: An empirical study." Software Engineering, IEEE Transactions on 25, no. 4 ,pp. 541-556 (1999).

Author



Ahmad Al- Shamailh, he took Bachelor's degree in Mutah University studied in the month of September 2010 to the date of the month of September 2013, a researcher in the field of data storage rules, residents of the city of Karak in Jordan.