AN APPROACH TO AUTOMATE THE RELATIONAL DATABASE DESIGN PROCESS

Wasana C. Uduwela[1], Gamini Wijayarathna[2],

[1]Department of Mathematics and Computer Science, The Open University of Sri Lanka
[2]Department of Industrial Management, University of Kelaniya

ABSTRACT

Information and Communication Technology improves the business competitiveness in both large scale enterprises as well as small and medium scale enterprises. Lack of technical knowledge in Information Communication Technology and the cost have been identified as challenges forsmall and medium enterprises to adopt ICT for their businesses. They can overcome this problem by using freely available tools/systems which aid to generate information systems automatically. However,they require the database structure; therefore, it is desirable to have a tool to automate the relational database design process. In the proposed approach, business forms were considered as the database requirement input sources among: learning from examples, natural language, structured input/output definition and schema definition and forms. The approach uses a functional dependency algorithm on the un-normalized data which is fed through the business form and then apply a normalization algorithm onthe discovered functional dependencies to have the normalized database structure. User intervention is needed to have the domain knowledgeof this approach. Finally, it develops the normalized database with all the keys and relationships; the accuracy of the out-come totally depend on the data fed by end users.

KEYWORDS

Information Communication Technology, Natural Language Processing, Business Forms, Relational Database, Normalization, Functional Dependencies

1. INTRODUCTION

Information and Communication Technology (ICT) plays a vital role in the modern business world; it improves business competitiveness [1] of large enterprises as well as Small and Medium Enterprises (SMEs) [2]. Although, SMEs interest to adopt ICT for their businesses, it is a challenge for them, because SMEs are lack of technical skills to develop ICT application in-house [3] and system cost to outsource the development of ICT applications or to purchase the off the shelf products [3]. It is a cost for them to maintain a separate team for this purpose as some countries have a big demand for the ICT expertise (ICT workforce has risen by 50% since 2010), [4]. However, it won't be a matter for SMEs if there is a system or tool to develop their ICT applications by themselves without expert knowledge.

Information systems are major application ofInformation and Communication Technology that can be used by businesses to ease their task. Technical knowledge is needed to develop information systems, but we found some freely available tools that support to develop information systems without much technical knowledge. These tools support the basic functions of the system - add, edit, delete, and view - for a given database structure [5]. Additionally, we found some researches on designing relevant forms automatically with basic functions for any given database structure [6].We can conclude that for all these approaches of information systems development need the database structure which should be provided by ourselves. The database is one of the most critical factors of all information systems, but database designers need technical

knowledge to develop databases based on the requirement specification of the system [7], which is difficult for non-technical people in small and medium sized enterprises.

In literature, we were able to find researches/studies on automating the database design process.Some of them have used the approaches on natural language processing, form based analysis, structured language, etc. We analyzedthe approaches based on natural language processing and form based analysis to identify the suitability of them to our target user groups as the others are not feasible for non-technical people in SMEs. A research papers on these tool/system analysis says that the approaches based on natural language processing as well as the form based analysis are not suited for our target user groups [8, 9], but the research paper [9] says that business form can be considered as a vital input source to automate the database design process than the others.The primary goal of thisresearch is propose an approach to automate the relational database design process based on business formsby eliminating the user involvements in the tasks which need technical skill.

This paper is organized as follows. Section 2 describes the background of the database design process and existing automation approaches. Section 3 gives a description on the new approach and section 4 discusses and concludes the paper.

2. LITERATURE ON RELATIONAL DATABASE DESIGN PROCESS AUTOMATION APPROACHES

There are few types of database models: Object Oriented Database model, Relational Database model, Network model and etc. Relational database model which is proposed by Dr.Codd widely used in almost all information systems to manage their transaction data, in past few decades [10]. Relational database consists with a set of tables (relations) and relationships among them [11]. Each table in a relational database has data fields, rows, primary key, and foreign key. A table is a collection of records; it has a two dimensional structure with column (data fields) and rows (tuples). The table represents the entities of the semantic model/conceptual model of a database (e.g. The “Item details” table represent the “Item” entity in the inventory control system). Each column of the table represents attributes of the entity [11] (e.g. “item_name” and “unit_price” are the column names of the “Item” table that represents the attributes of “Item” entity). The primary key is an attribute or combination of attributes of the table. It is used to identify each row uniquely [11]. The tables in a relational database model have relationships among each table to represent some relevant fact of the system [11] (e.g. The relationship between “Item” and “Invoice” tables of “Inventory control system, ” says “Invoice” has “Item”). Foreign key helps to maintain the relationship between the tables, by keeping references [11].

2.1RelationalDatabase Design Process

To create a relational database, few steps need to be followed: gather system requirement for the database, design the conceptual model (Entity Relation Diagram is the most popular model), map the conceptual data model to logical data model, and translate the logical data model (normalized database) into the target database- physical data model- (Figure 1 describes the relational database design process).

The logical data model should be a normalized one. Database normalization also introduced by Dr Codd and it helps to minimize the data redundancy in relational databases by organizing their attributes and tables.Normalization process confirms whether the relational database satisfies a certain normal form in a relation schema through a series of tests. In order to achieve each normal form, database designers have to analyze the give relation schema based on their functional

dependencies and primary keys. There are three types of normal forms commonly use in the database design process such as: First Normal form (1NF), Second Normal form (2NF) and Third Normal form (3NF).Database designers/developers need technical knowledge for each step of the process as well as for the normalization process.

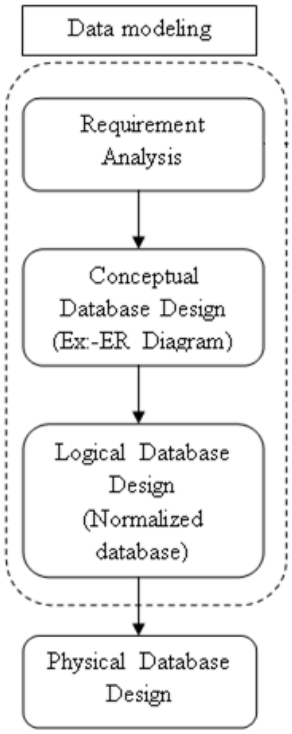

Figure 1. Relational Database Design Process

2.2 Database Design Process Automation Approaches

2.2.1 Converting user requirementsinto a conceptual model/database structure

Converting user requirements into conceptual model is the most critical task in therelational database design process processandothers are straightforward [12, 13]. The research paper [14] categorized existing approaches on gathering database requirements – the first step of the database design process - into four categories: learning from examples, natural language, structured input/output definition, and schema definition. This research does not consider the approaches: learning from examples, structured input/output definition, and schema definition given by the user, because non-technical people do not have the technical knowledge to state the requirements in structured input/output definition, schema definition and to learn from examples.
Usually, database designers use system requirement specifications which are in natural language to obtain the database requirements. In the literature survey,we extracted systems/tools, which can automate the database design process from the requirements given in natural language. Some tools generate conceptual model (ER model) from the requirement while some tools generate the normalized relational database [7, 13, 15] from the requirement. In order to improve the accuracy of the out-comes, these tools apply few techniques: pre-process the natural language input (effective way to remove noisy data and achieve normalization) using controlled language (putting constraints on the input), and sublanguage; use dialogues (effective in requirements elicitation and acquisition with having a dialogue with the user) [13]; and use knowledgebase –

domain knowledge [16], but the accuracy of the out-come had reduced for complex requirements [8]. These tools need user involvement to generate their out-comes or to improve the accuracy of out-comes further [8]; therefore, approaches in natural language also ignored as it difficult to communicate the requirements easily and difficult to illustrate the correct requirement at the beginning speciallynon-technical people and novel database designers.

Except the approaches categorized in the research paper [14], business forms also help to gather requirements to database design: forms represent the database layer of the system [14]. Researchers have been already carried out to develop tools/systems in order to automate the database design process using the forms [9]. Most of them give the correct out-come; therefore, the research paper [9] suggests that the form is one of the best approaches to gather comprehensive database requirement accurately. However, existing tools/systems are unfit tonon technical and novel database designers they need user involvement - with technical knowledge - for some steps of the process [9].

2.2.2 Normalization approaches

The concept of functional dependencies (FDs) is the foundation of the normalization process. Database designers practice their domain knowledge and experiences to identify the functional dependencies of databases [11]. The functional dependency is a property of the semantics or meaning of the attributes and it describes a constraint between two sets of attributes from the database [17]. It is denoted by X→Y between two sets of attributes X and Y. X and Y are subsets of R specifies a constraint on the possible tuples that can form a relation R. The constraint is that for any two tuples t1 and t2 in r that have t1 [X] =t2 [X] they must also have t1 [y] = t2 [y] [17].
We found few algorithms whichhave been developed to obtain the functional dependencies in the given set of data. Some of them are FD_Mine [18], FastFD [19], DepMiner [20], and TANE [21]. We can use these algorithms to have an idea about the functional dependencies in the existing databases.Among them, FD_Mine is better for data tables with a large number of records and a lesser number of attributes. If the number of attributes large then we can use FastFD algorithm, but it has poor performance for a large number of data records [22].

Database normalization can be donewith relevant functional dependencies. We found few normalization algorithms to get the normalized database based on the given set of functional dependencies. Most of them were developed to learn database normalization process [23, 24], while one of them is a semi automated one [25]. There are two algorithms that can be used for normalization process and both gives the normalized tables with primary keys and foreign keys [26, 27].Wetested the algorithm [26] among them and it works well for any given set of functional dependencies.

3. FORM BASED APPROACH TO AUTOMATE THE DATABASE DESIGNPROCESS

A form based approach was used for the proposed tools, because manual forms as well as the digital forms are widely used to gather and report the business data in a structured way. The forms are more familiar, easy to read, and understandable to end-user to communicate many requirements of the system [28]; it provides common vocabulary, and set of goals among end users and data processing professionals, rather than providing exhaustive requirements collection by end-users [29]. Thus, forms can be considered as a vital source for database design process [9].

System users use forms to communicate with the systems while they are created by one source of the business function and sometimes modified or used in some other function [30]. As an example invoice is a commercial instrument issued by the seller to the customer; it describes all

transaction details that happened between customer and seller. Business forms are categorized into two groups; update forms (ex: - invoice, purchase order) - keep track of real system state changing to update, and report – for reporting [30]. Update forms are used for data entry purposes; it consists with the all the underlying data that necessary to consider for database design. Therefore, update forms were considered for the research. An update form consists with a title, set of fields, and blank spaces to provide or display value of form fields. When a form fields are filled with values then it becomes an instance of a particular form [29].

3.1 Architectural Design of the Approach

The architectural design of the proposed approach is illustrated in Figure 2. First, it gathers datainto an un-normalized database, which is received from forms. This research categorized the form fields into two groups: repetitive fields and non-repetitive fields. Repetitive fieldsare repeated with data in a form instance (typically these fields are in a table format) while non-repetitive fields are not repeated with the data (only one value) in a form instance. Figure 3 shows the field categorization can be happened in a customer invoice. It shows non-repetitive fields: CusName- Customer Name, CusAdd –Customer Address, CusTP – Customer Telephone Number, InvoiceNo, Date, Total, Sub Total, and Tax; and repetitive fields: IntemNo, ItemName, UnitPrice, Quantity, and Amount. This makes the hierarchical format for the form like root-node describes the non-repetitive fields while leaf-node describes the repetitive fields. The entered data are gathered into two tables, one table to keep repetitive data and another one for non-repetitive data (these tables are un-normalized tables).

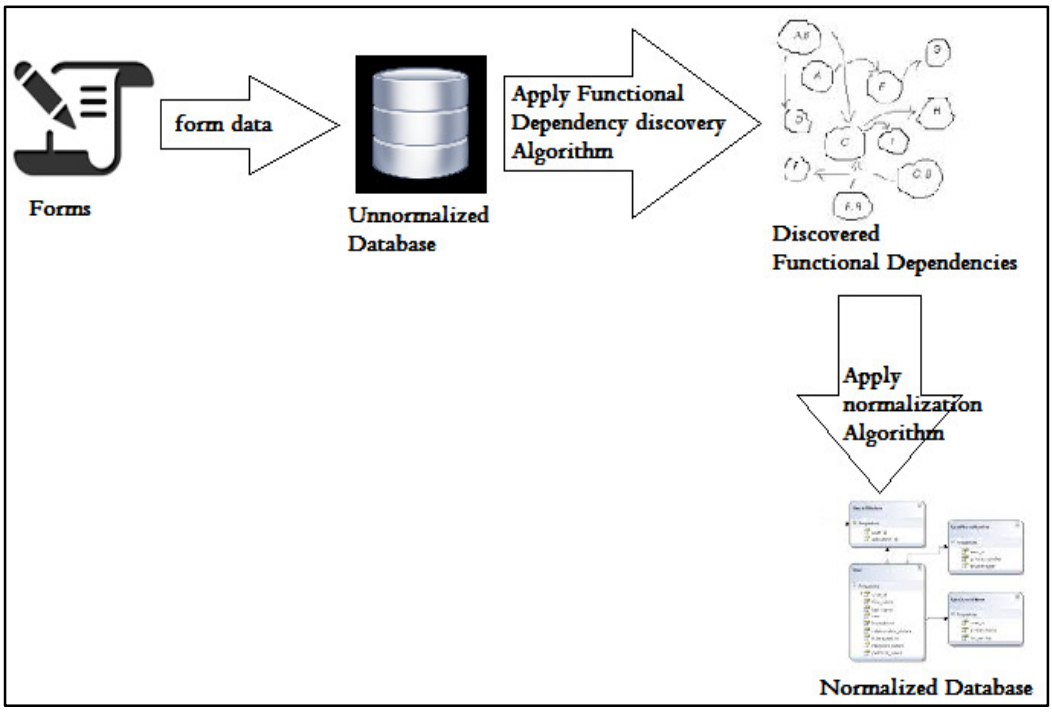

Figure 2: Architectural Design

The functional dependency algorithm is applied onthese two unnormalizedtables. If there are all the varieties of data exist, then we can expect the accurate outcome from the FD algorithm.If the number of attributes is greater than fifteenFastFoD algorithm should applyto keep the better performance of the system, if not FD_Mine algorithm shows better performance [22]. End users' domain knowledge is needed to apply improve the accuracy of the discovered functional

dependencies, because sometimes it gives functional dependencies as "ItemNo→ItemName" and "ItemName→ItemNo" which cannot be sorted out by the system.In such situations system needs end users' domain knowledge to correct the functional dependencies. Further system conducts a conversation with the user, based on the functional dependencies to make sure the gathered data have all varieties of data. If there are wrong functional dependencies, then end user need to make the correction of data in the data tables accordingly. User inputs and their decisions on functional dependencies keep in a log file for future use. Since it is not related to technical knowledge non-technical people can give their view point to have the set ofaccurate functional dependencies.

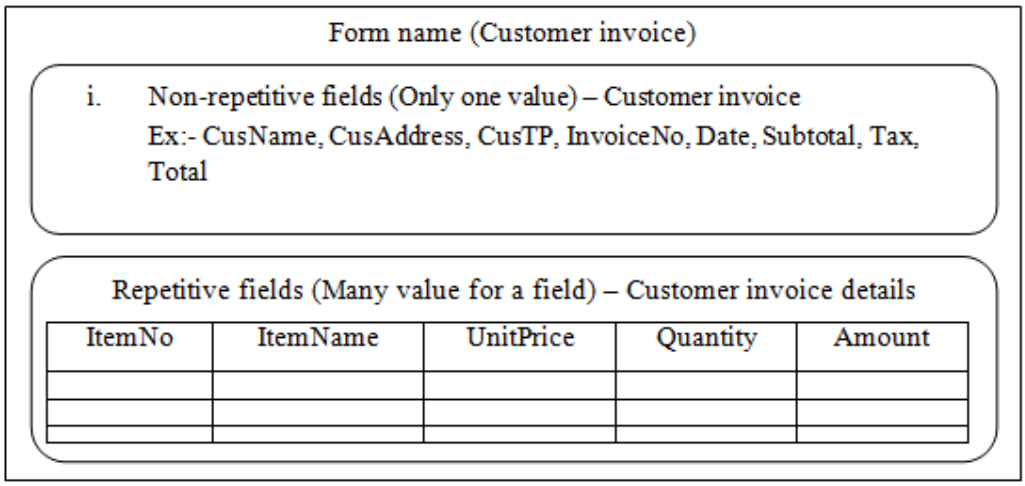

Figure 3.Sample Form Structure

Afterward, it applies the normalization algorithm [26] on the discovered functional dependencies to have normalized tables. This algorithm automatically identified the primary keys. Based on that we can identify the foreign keys as well.To have accurate normalized tables, first check whether there is data fieldsrepetition between the data tables other than the foreign keys. If there are repetitive data fields means tables have not normalized accurately.Then user asked to enter another 50 data records and repeat the steps described above. The knowledge in the log file is applied to check the accuracy of the newly created functional dependencies. If there are new functional dependencies than the previous, end user need to check their accuracy. Log file also needs to update according to that. Though there are repetitive fields have or not, end user needs to add another 50 records and repeat the steps described above. If it generates same set of normalized data tables in both times prove that system has generated the accurate normalized data tables. Then forms can directly keep their data in normalized database.

4. DISCUSSION AND CONCLUSION

We found that the approach gives us the normalized database for business forms by eliminating the middle steps of the database design process,when we give the form details in an organized manner as described above.Please refer Figure 4 for its illustration.

This approach also requires the user involvement, but it is only to extract domain knowledge which is familiar, even for non-technical end-users. The proposed approach poses simple questions for this purpose; therefore, end-users do not have to interact with technical stuffs like other tools required. They do not need to extract the business entities in the form by themselves as other tools/systems do (identify the primary keys, foreign keys and the relationships among tables) [30]. The accuracy of the out come totally depend on the data fed by end user. Finally, we

can conclude that the approach helps to create the database dynamically without user intervention for technical tasks and it will help a lot for our target user group – non-technical peoples in small and medium sized enterprises.

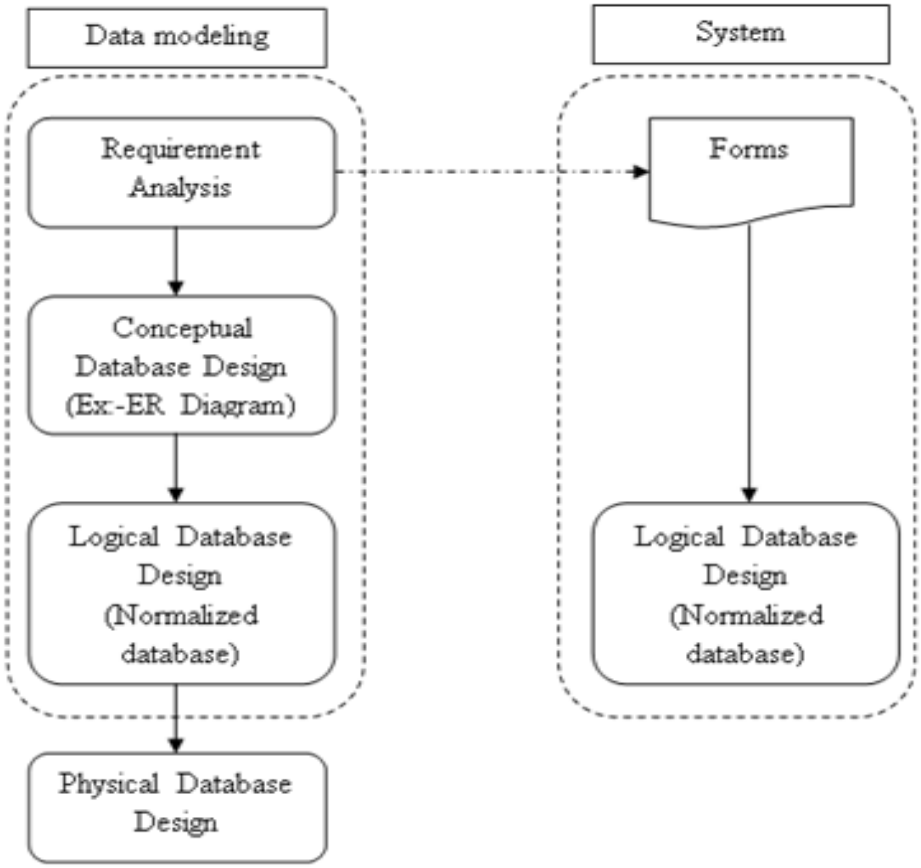

Figure 04: System in data modeling levels ofabstractions

REFERENCES

[1] Skoko, H., Ceric, A., & Huang, C. (2008). ICT adoption model of Chinese SMEs. Int. J. Bus. Research , Vol. 8, No. 4, pp.161-165.

[2] Apulu, I., & Latham, A. (2009). Information and Communication Technology Adoption: Challenges for Nigerian SMEs. TMC Academic , vol. 4, pp.64-80.

[3] Harindranath, G., Dyerson, R., & Barnes, D. (2008). ICT in Small Firms: Factors Affecting the Adoption and Use of ICT in Southeast England SMEs. European Conf. Inform. Syst., (pp. pp.889-900).

[4] "National ICT workforce survey, ". I. (n.d.). Retrieved from http://www.icta.lk/attachments

[5] BigProf software. . (2014, June). Retrieved from appgini.: http://www.bigprof.com/appgini

[6] Jayapandian, M., &Jagadish, H. (August 2008). Automated Creation of a Forms-based Database Query. VLDB Endowment , vol. 1, no. 1, pp. 695-709 .

[7] Buchholz, E., Cyriaks, A., Düsterhöft, Mehlan , H., &Thalheim, B. (1995). Applying a Natural Language Dialogue Tool for Designing Databases.1st Int. Workshop on Applicat.of Natural Language to Databases.

[8] Uduwela, W., &Wijayarathna, G. (2014). Survey on Tools and Systems to Generate ER Diagram from System Requirement Specification.IEEE International Conference on Industrial Engineering and Engineering Management. Selangor, Malaysia.

[9] Uduwela, W., &Wijayarathna, G. (2013). A Survey on tools/systems to generate database from form analysis. Annual Academic Sessions of OUSL, (pp. 377-381).

[10] Sugumaran, V., & Storey, V. C. (2006). The role of domain ontologies in database design: An ontology management and conceptual modelling environment . ACM Transactions on Database Systems (TODS)

[11] Elmasri, R.., &Navathe, S. B. "The relational data model and relational database constraints", in Fundamentals of Database Systems, New York, Addison-Wesley, 6th edition, pp. 59-75.

[12] L. A. Al-Safadi, (Sep. 2009), "Natural language processing for conceptual modeling," Int. J. Digital Content Technology and its Applicat., vol. 3, pp. 47–59,.

[13] S. Du, (2008) "On the use of natural language processing for automated conceptual data modeling," Ph.D. dissertation, Pittsburgh Univ.

[14] Veronica, P., Tseng , Michael, V., &Mannino. (1989). A method for database requirements collection. Journal of Management Information Systems , vol. 6, no. 2, pp.51-75.

[15] Gomez, F., Segami, C., &Delaune, C. (1999). A system for the semi-automatic generation of E-R models from natural language specifications. In Data & Knowledge Engineering (pp. ELSEVIER (pp. 57-80))

[16] Thonggoom, O. (2011). Semi-Automatic Conceptual Data Modeling Using Entity and Relationship Instance Repositories.Ph.D Dissertation, University of Drexel

[17] Jalal A., D. Bader & A. Awajan, (2008). Mining Functional Dependency from Relational Databases Using Equivalent Classes and Minimal Cover , Journal of Computer Science 4 (6): 421-426, 2008 ISSN 1549-3636 © 2008 Science Publications, pp.421-426

[18] H. Yao, H. J Hamilton, and C. J. Butz., (2002) "Fd_Mine: discovering functional dependencies in a database using equivalences. In Data Mining", ICDM2003. Proceedings. 2002 IEEE International Conference on, pages729–732.

[19] W. Catharine, C. Giannella, and E. Robertson, (2001) "Fastfds: A heuristic-driven, depth-first Algorithm for mining functional dependencies from relation instances extended abstract.", In Data Warehousing and Knowledge Discovery, pages 101–110.

[20] L. St´ephane, J.-MarcPetit, and L. Lakhal., (2000), "Efficient discoveryof functional dependencies and Armstrong relations. In Advances in Database Technology EDBT 2000, Springer, pages 350–364.

[21] H. Yk¨a, J. K¨arkk¨ainen, P. Porkka, and H. Toivonen, (1999), "Tane: An efficient algorithm for discovering functional and approximate dependencies," The computer journal, 42(2),pp.100–111.

[22] W.C.Uduwela and P.G.Wijerathna, (in press),"Comparative Study of Functional Dependency Generation Algorithms", International Journal of Advanced Information Science and Technology

[23] Georgiev, N. (2008, September). A Web-Based Environment for Learning Normalization of Relational Database Schemata.Master's Thesis in Computing Science. Department of Computer Science, Ume˚a University.

[24] Ali Yazici, ZiyaKarakaya. (2007). JMathNorm: A Database Normalization Tool Using Mathematica. ICCS '07 Proceedings of the 7th international conference on Computational Science, Part II (pp. 186 - 193). Berlin: Springer.

[25] Dhabe, P., Deshmukh , S., &Dongare, Y. (February 2011). RDBNorma: - A semi-automated tool for relational database schema normalization up to third normal form. Journal of Database Management Systems (IJDMS) , vol. 3, no. 1, pp. 133-154.

[26] Demba, M. (2013). An Algorithmic Approach to Database Normalization. International Journal of Digital Information and Wireless Communications (IJDIWC) , 57-65.

[27] Bahmani, A.H. ,Naghibzadeh, M. &Bahmani, B. (2008). Automatic database normalization and primary key generation.Electrical and Computer Engineering, 2008.CCECE 2008. Canadian Conference (pp. 11-16). Niagara Falls, ON: IEEE.

[28] Choobineh, J., Mannino, M., Nunamaker , J., J.R., &Konsynski, B. (1988). An Expert Database Design System Based on Analysis of Forms. IEEE Transactions on Software Engineering ,vol 14, no. 2, pp.242-253.

[29] Choobineh, J., Mannino, M., & Tseng, V. (1992). A form-based approach for database analysis and design. Communications of the ACM , Vol. 35, no. 2, pp.108-120.

[30] Mogin, P., &Luovic, I. (n.d.). A Prototyping CASE tool.