New Horizons - Journal of Student Research | Volume I Number I 2024 |

Published by the University "St. Kliment Ohridski" - Bitola (www.uklo.edu.mk)

(*) Corresponding author
DOI: 10.20544/NEWHORIZONS.1.1.24.P03

CONCEPTUAL AND LOGICAL DESIGN OF RELATIONAL DATABASES: A CASE STUDY

Tamara Taseska, student*
Faculty of Economics – Prilep,
University St. Kliment Ohridski – Bitola
tamarataseska@gmail.com

Prof. Dr. Ilija Hristoski, mentor
Faculty of Economics – Prilep,
University St. Kliment Ohridski – Bitola
ilija.hristoski@uklo.edu.mk

ABSTRACT

In a time of increasingly accelerated and ubiquitous digitalization, and in conditions where technology, especially information and communication is available to everyone, business entities are becoming the subject of intensive digital transformation that allows them to fundamentally change the way they work to deliver more value to the customers. Today, technologies are the main driving force of businesses, and the information obtained through data processing and analysis processes is the lifeblood of knowledge-based economies, as well as of the entire digitized society. The wave of digital transformation also swept through the segment of driving schools for training future drivers at the global and local levels, a large number of driving schools previously carried out automation of manual processes using ICT-based software solutions, some introduced management information systems, and some completely migrated their operations to a cloud environment. Regardless of the form and degree of digitization, the common denominator of all these technical-technological advances is the use of relational databases. Based on the 'database first' principle, the main goal of this research is to offer a conceptual and logical design of a relational database (RDB) that can later be physically implemented in any relational database management system (RDBMS) to meet the needs of accurate, timely, and complete information about any driving school. Throughout the development of a generic RDB, the power of conceptual design has been demonstrated through the use of E-R diagrams, as one of the leading methodologies for modeling. Through the proposed user questionnaires, the power of relational algebra has been demonstrated as the main formal, theoretical, and procedural query language and mathematical notation of SQL data manipulation operations (SQL DML) performed on the RDB.

KEYWORDS: relational database, driving school, E-R diagram, relational schema, relational algebra.

1. INTRODUCTION

The modern dynamics of living and working have imposed on a wide range of people the need to have a driver license of a different category. A direct consequence of this trend is the increasing number of driving schools all over the world, but also in our country, whose main purpose is to provide training for professional and non-professional drivers and organize exams for obtaining a driver license. In the era of ubiquitous digitization in all social pores, the numerous driving schools present in our country still keep records of their operations manually, using paper/printed forms, which are often non-standardized, as well as informal notes and records. In certain cases, the entire record is kept in the form of Microsoft Excel tables/worksheets, which are mostly isolated from each other, inaccessible to a wider range of users, unsafe, and subject to numerous errors and inconsistencies.

The numerous shortcomings and problems arising from the traditional way of organizing business activities contribute to inefficient, ineffective, and uneconomical operations. The source of all problems is the absence of a central repository of organized data, ie. the non-existence or non-use of a database. This modern approach allows all relevant data that are of interest to the functioning of a given business entity to be organized in one logical unit and to be contained in a single secondary memory space.

Databases (DBs) represent the highest hierarchical level of data organization in computer systems; they are a shared collection of logically interconnected files designed to meet organizational needs for timely, accurate, and complete information.

The relational data model is the basis of relational databases (RDBs), which today are still the most dominant and popular form of organizing data in global frameworks. As a second-generation data model, the relational data model, among other things, aims to: (a) provide a higher degree of data independence from applications; (b) to give a theoretical and scientific basis to the semantics of the data and their consistency; (c) to address problems arising from data redundancy; (d) to allow expansion of data manipulation languages based on set operations.

The process of designing relational databases is a complex process that consists of three stages: (1) conceptual design, represented through the Entity-Relationship model, E-R model, that is, through the corresponding E-R diagram; (2) logical design, which constitutes the set of relational schemes that make up the RDB, which are derived directly from the E-R diagram; (3) physical design and implementation, which constitutes the physically implemented RDB using a specific relational database management system (RDBMS).

In this paper, emphasis will be placed only on the first two stages of the Relational Database Design process, as they are independent of any particular RDBMS. The paper aims to offer a universal and general RDB design solution that will be able to meet the needs of any driving school for timely, accurate, and complete information, and that can later be implemented in any RDBMS. The demonstration of the capabilities offered by such an RDB will be done through the specification of a certain number of representative user questionnaires, defined with the help of relational algebra operations.

The paper is structured as follows: in the next section, some of the modern research in the world on the topic "development of a database system for the needs of driving schools" is presented. Section 3 is entirely devoted to the elaboration of the research organization, the collection of relevant data and information, as well as to the explanation of the methodology on which the development of the RDB design proposed here is based. Within section 4, through three subsections, the development of the relational database is presented: problem description, conceptual design, and logical design. The presentation of the application of RDB was made in Section 5, through two subpoints: a brief overview of relational algebra and the specification of a certain number of representative user questionnaires, defined using the original notation of relational algebra. The last section, section 6, summarizes the concluding observations, traces the directions of further development, and provides recommendations regarding the treated topic.

2. RELATED RESEARCH

Modeling relational databases using E-R diagrams (a methodology proposed by Peter Chen) has long been a standardized approach used both in the development of small RDBs and in the conceptual design of large RDBs. Due to the current situation and the emergence of an increasing number of driving schools and candidates for obtaining driver licenses not only here but also in other countries, in the past period, there has been a noticeable increase in the number of published seminar works and diploma theses, master theses, research papers, articles, and blogs in which the authors propose different solutions in terms of designing an appropriate RDB. The following is a chronological review of some of the research done in the last ten years.

Ishak (2012) built his diploma thesis around the development of DriSIS (Driving School Information System) - an information system intended for driving schools in Malaysia. He applied the "waterfall" methodology in the system development, and MySQL as the back-end RDBMS.

In 2013, PACE University in New York, USA, conducted a study on the development of a management information system for the needs of Manhattan School of Driving LLC. This study was based on the use of a relational database, within which an E-R diagram was presented. The diagram reflects the conceptual design of the complete RDB implemented within the system (MSD, 2013).

In his blog, Kher (2015a; 2015b) elaborates on the conceptual design of a relational database for a reservation system catering to the needs of a generic driving school.

The focus of the research of Zhou, Cai, & Liu (2016) is the design and implementation of an online information management system (Information Management System) for the needs of driving schools, based on the Dreamweaver development environment, which is also built on a suitable E-R diagram.

Tamilarasu (2019) describes in his dissertation the development of an online monitoring system for the needs of driving schools, based on the use of PHP (front-end) and MySQL (back-end).

In her master thesis, Mithrasena (2020) describes the development of TrainMe - an integrated, web-based management information system (MIS), within which she proposes a conceptual design of relational BP in the form of an E-R diagram.

Also worth noting is the fact that increasingly certain web portals are offering online lessons and tutorials, such as iNetTutor.com (2021; 2022) and 101Computing.net (2022), which elaborate on the gradual development of the E-R diagram of a software system with a database for the needs of driving schools.

3. DATA AND METHODOLOGY

The data and information necessary for the construction of the conceptual and logical data model are provided with the help of unstructured interviews with driving instructors, employed in several private local driving schools in Prilep and with their owners and/or managers, in the period from 07.11.2022 to 27.11.2022. The acquired data and information served as the foundation for identifying the fundamental components of the E-R diagram: entity types, their attributes, and the connections/relationships between individual entity types while establishing all relevant properties of the connections: their degree, cardinality, and modality at their ends.

The research is entirely based on the application of two methodologies: (a) the methodology for conceptually designing relational databases using E-R diagrams, originally proposed by Peter Chen (Chen, 1976; Elmasri & Navathe, 2016, pp. 59–105), which are based on the principles and postulates of the relational data model, developed by E. F. Codd (Codd, 1970), and (b) the methodology for deriving a logical RDBs design by mapping the E-R diagram into a relational database schema (Elmasri & Navathe, 2016, pp. 289–306). The design of RDB is based on the principles of systems analysis and design. User questionnaires, on the other hand, are defined following the definitions of relational algebra operations (Elmasri & Navathe, 2016, pp. 239–288).

4. DEVELOPMENT OF A RELATIONAL DATABASE

4.1 Description of the problem

Based on the conducted interviews with driving instructors, owners, and managers of several private driving schools, relevant information was obtained based on which the problem can be described, as follows:

Each driving school is identified by its name and the city in which it is located. The year of establishment, telephone number, email address, address, VAT (Value Added Tax) number, and website are also recorded. Many candidates are enrolled in each driving school. One candidate is enrolled in only one driving school.

Each candidate is identified by his EMBG (unique citizen identification number), and in addition to it, the name, surname, gender, date of birth, place of birth, telephone number, and e-mail address should be recorded. The date of enrollment in the driving school and the category of vehicles for which the training will be carried out should also be recorded.

Each driving instructor in the system should be identified by his code. Apart from that, his EMBG, first name, last name, gender, phone number, and e-mail address should also be recorded. Each instructor works for one driving school. For each driving school, multiple instructors are working. The start and end dates of employment for each instructor need to be recorded.

Each training is identified by its unique code. Also, the number of scheduled hours, the category of vehicles, and the price per hour should be recorded. Each training is conducted by one instructor, and each instructor conducts multiple trainings. Each training applies to one and only one candidate. Each candidate corresponds to only one training.

Each vehicle is identified by its unique code. Apart from that, the manufacturer, model, type of vehicle, chassis number, year of production, registration plate number, engine power

[kW], and engine volume [cm^3] should also be recorded in the RDB. Each training is carried out with one vehicle, and with each vehicle, several trainings are carried out.

Each term within a particular training is identified by its serial number. The date of the session, the time from which the training starts, the time until which the training lasts, as well as information on whether the training was held in that term are also recorded. If the training is not held, information about the reason should be saved. Each training is realized in several terms. Each term, on the other hand, is realized within one training session. Terms are identifying and existentially dependent on the training to which they belong.

Each exam is identified by a unique code, and the name of the exam (theory, driving range, city driving) and the price of the exam should be recorded in the system. Each candidate takes multiple exams, and each exam is taken by multiple candidates. It is necessary to record the date of the exam, the time of the exam, the outcome of the exam (passed, failed), the number of negative points, and the serial number of the attempt (the number of times the candidate takes the exam).

Each examination committee is identified by its unique identification number. Each examination committee participates in the administration of multiple exams, and one examination committee is responsible for each exam.

4.2 Conceptual design

The conceptual design of the RDB is generated based on the problem description and is realized and documented through the process of creating an appropriate E-R diagram. The basic building blocks used in the creation of any E-R diagram include entity types (represented by rectangles), attributes of entity types (represented by ellipses/ovals), and relationships between individual entity types (represented by diamonds). Entity types represent collections of abstract objects of the same kind (persons, objects, places, events, concepts) that can be uniquely identified in the analyzed system (based on the description of the problem) and for which the organization wants to keep data. Attributes are individual properties of entity types – common characteristics of all individual instances (examples) of each entity type. The relationships represent named associations/connections between two or more entity types, which differ from each other according to their degree, cardinality, and modality. The degree of a relationship indicates how many entity types participate in its construction. Cardinality and modality are entered at each separate end of the relationship. Cardinality indicates the maximum number of instances of the corresponding entity type participating in the relationship (1 or M), and modality indicates the minimum number of instances of the corresponding entity type participating in the relationship (0 = optional/optional participation in the relationship; 1 or more = mandatory participation in the relationship).

A special type of entity type is the so-called "weak" entity, which is existentially and identificationally dependent on another, conditionally, "strong" entity type (represented by a rectangle drawn with a double line). A relationship between a "strong" and a "weak" entity type always has a cardinality of 1:N (represented by a diamond drawn with a double line).

The key of each entity type consists of one or more attributes that uniquely identify all of its instances (represented as underlined attributes).

The resulting E-R diagram, which fully corresponds to the description of the problem given in the previous point, is presented in Figure 4.1. The diagram was drawn using the EDraw Max v6.8 software tool, using the original notation proposed by Peter Chen.

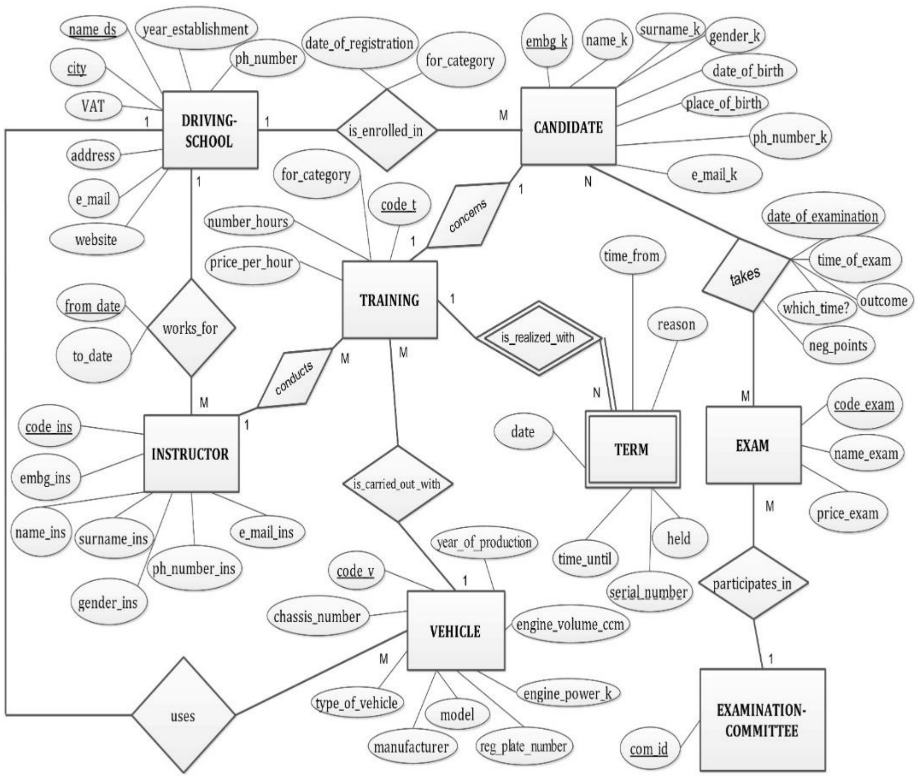

Figure 4.1 – The proposed E-R diagram, rotated (Source: The author)

4.3 Logical design

The logical design of an RDB represents the relational schema of the RDB, obtained by directly mapping/converting the E-R diagram into a set of relation schemas that will constitute the database. In the physical implementation later, each schema will be materialized as a source table within the RDB, in which specific data will be stored. The schema of each relation, according to the relational data model, consists of the relation name and specifications of all its attributes, making a clear distinction between the key attributes (attributes that constitute the primary key) of the non-key attributes, and attributes of any foreign keys, if they exist.

The logical design is obtained in two stages. In the first phase, the relations that directly derive from the entity types are identified, and in the second phase, the connections/relationships between the entity types are analyzed and, based on their properties (degree, cardinality, and modality), or the previously identified relations are updated. by adding a foreign key, or creating a new relation.

1) Relations that derive directly from entity types:

DRIVING_SCHOOL (**<u>name_ds</u>**, **<u>city,</u>** VAT, year_establishment, ph_number, address, e_mail, website)

INSTRUCTOR (**code_ins**, embg_ins, name_ins, surname_ins, gender_ins, ph_number_ins, e_mail_ins)

CANDIDATE (**embg_k**, name_k, surname_k, gender_k, date_of_birth, place_of_birth, ph_number_k, e_mail_k)

TRAINING (**code_t**, for_category, number_hours, price_per_hour)

VEHICLE (**code_v**, chassis_number, type_of_vehicle, manufacturer, model, reg_plate_number, engine_power_kw, engine_volume_ccm, year_of_production)

TERM (**serial_number**, date, time_from, time_until, held, reason)

EXAM (**code_exam**, name_exam, price_exam)

EXAMINATION_COMMITTEE (**com_id**)

2) Analysis of relationships between entities; updating existing foreign key relations and/or introducing new relations

- Relation DRIVING_SCHOOL – CANDIDATE; '*is_enrolled_in*'; 1:M

IS_ENROLLED_IN (**embg_k**, date_of_registration, for_category, *name_ds**, *city**)

- Relation DRIVING_SCHOOL – INSTRUCTOR; '*works_for*'; 1:M

WORKS_FOR (**code_ins**, **from_date**, to_date, *name_ds**, *city**)

- Relation INSTRUCTOR – TRAINING; '*conducts*'; 1:M

TRAINING (**code_t**, for_category, number_hours, price_per_hour, *code_ins**)

- Relation CANDIDATE – TRAINING; '*concerns*'; 1:1

TRAINING (**code_t**, for_category, number_hours, price_per_hour, *code_ins**, *embg_k**)

- Relation TRAINING – VEHICLE; '*is_carried_out _with*'; M:1

TRAINING (**code_t**, for_category, number_hours, price_per_hour, *code_ins**, *embg_k**, *code_v**)

- Relation TRAINING – TERM; '*is_realized_with*'; 1:N

TERM (**code_t**, **serial_number**, date, time_from, time_until, held, reason)

- Relation CANDIDATE – EXAM; ‘*takes*’; N:M

TAKES (**embg_k**, **code_exam**, **date_of_examination**, time_of_exam, which_time?, outcome, neg_points)

- Relation EXAM – EXAMINATION_COMMITTEE; ‘*participates_in’*; M:1

EXAM (**code_exam**, name_exam, price_exam, *com_id**)

- Relation DRIVING_SCHOOL – VEHICLE; ‘*uses*’; 1:M

VEHICLE (**code_v**, chassis_number, type_of_vehicle, manufacturer, model, reg_plate_number, engine_power_kw, engine_volume_ccm, year_of_production, *name_ds**)

The final relational schema is as follows:

DRIVING_SCHOOL (**name_ds**, **city**, VAT, year_establishment, ph_number, address, e_mail, website)

INSTRUCTOR (**code_ins**, embg_ins, name_ins, surname_ins, gender_ins, ph_number_ins, e_mail_ins)

CANDIDATE (**embg_k**, name_k, surname_k, gender_k, date_of_birth, place_of_birth, ph_number_k, e_mail_k)

TRAINING (**code_t**, for_category, number_hours, price_per_hour, *code_ins**, *embg_k**, *code_v**)

VEHICLE (**code_v**, chassis_number, type_of_vehicle, manufacturer, model, reg_plate_number, engine_power_kw, engine_volume_ccm, year_of_production, *name_ds**)

TERM (**code_t***, **serial_number**, date, time_from, time_until, held, reason)

EXAM (**code_exam**, name_exam, price_exam, *com_id**)

EXAMINATION_COMMITTEE (**com_id**)

TAKES (**embg_k***, **code_exam***, **date_of_examination**, time_of_exam, which_time?, outcome, neg_points)

WORKS_FOR (**code_ins***, **from_date**, to_date, *name_ds**, *city**)

IS_ENROLLED_IN (**embg_k***, date_of_registration, for_category, *name_ds**, *city**)

The final relational schema consists of a total of 11 relations, each of which is in the Third Normal Form (3NF). This means that the non-key attributes in each relation depend only on the primary key and on the entire primary key (in cases where it is composite/complex).

In the resulting final relational schema, relation names are in capital letters, key attributes (primary key attributes) are bold (bold) and underlined with a solid line, and foreign keys are written in italics, underlined with a dashed line and marked with an asterisk (*). If a particular primary key attribute is also part of a foreign key, it is also marked with an asterisk (*). All other attributes that are not in bold or italics are non-key attributes.

5. APPLICATION OF THE RELATIONAL DATABASE

5.1 Relational algebra

Relational algebra is one of the main formal theoretical query languages of relational databases, with operations that are defined on one or more source relations, to obtain another, derived relation (view) without changing the structure and content of the original relations. It represents an interface between the user of the relational database and the data stored in it, but also a mathematical record of the SQL (Structured Query Language) operations that are performed over the relational database. The set of operations on relations in relational algebra consists of the following five basic operations: selection (Selection), projection (Projection), Cartesian product (Cartesian product), union (Union), and difference of sets (Set difference). Apart from the basic operations, other additional operations that can be expressed through the five basic operations are the operations of joining (Join), intersection (Intersection), and division (Division).

A brief elaboration of the most important operations of relational algebra follows.

The SELECTION operation (σ) is a unary operation that defines another relation containing a horizontal subset of tuples - only those tuples that satisfy the specified logical condition.

The PROJECTION operation (Π) is a unary operation that defines another relation containing a vertical subset of attributes of the relation, eliminating duplicate tuples.

The CARTESIAN PRODUCT operation ($\times$) is a binary operation that defines another relation representing the combination of each tuple from the first relation with each tuple from the second relation.

The UNION operation ($\cup$) is a binary operation that defines another relation containing all tuples from either the first or the second relation, or both while eliminating duplicate tuples.

The SET DIFFERENCE operation (\ ) is a binary operation that defines another relation containing all tuples from the first relation that are not in the second relation.

The JOIN operation ($\bowtie$) is a binary operation that defines another relation whose tuples represent combinations of related tuples from the first and second relations, which also satisfy a specific condition. The join operation is based on the Cartesian product of the two relations, followed by the application of the selection operation. Generally, there are two main forms of the join operation: (a) inner joins - Theta join, Equijoin, and Natural join; and (b) outer joins - Left outer join, Right outer join, and Full outer join.

The INTERSECTION operation ($\cap$) is a binary operation that defines another relation containing all the common tuples that simultaneously belong to both relations.

The DIVISION operation (÷), applied to two relations, defines a new relation that contains the tuples from the first relation (dividend) that match the combination of each tuple in the second relation (divisor).

In the following subsection, the operations of relational algebra will be used to define a series of representative *ad hoc* user queries with an aim to illustrate the efficacy of query languages in extracting high-quality information from a relational database, showcasing their capability to address complex data retrieval needs efficiently.

5.2 Specification of user questionnaires

User queries can specify requests for obtaining specific information from the database, requests for performing certain operations on the data in the database, or both at once. The query can provide an answer to a simple or complex question, perform calculations, or combine data from two or more different relations.

1. *Which driving schools were founded after 2020?*

$$\Pi_{\text{name\_ds, city}} (\sigma_{\text{year\_establishment} > 2020} (\textbf{DRIVING\_SCHOOL}))$$

2. *Find the names and surnames of female candidates who enrolled in driving school on May 31, 2021.*

$$\Pi_{\text{name\_k, surname\_k}} (\sigma_{\text{date\_of\_registration} = \text{'31.05.2021'} \wedge \text{gender\_k} = \text{'F'}} (\textbf{IS\_ENROLLED\_IN} \bowtie \textbf{CANDIDATE}))$$

3. *To display the details of the terms when training was carried out for the candidate with EMBG = '2203989440050'!*

$$\Pi_{\text{serial\_number, date, time\_from, time\_until, held}} (\sigma_{\text{EMBG} = \text{'2203989440050'}} (\textbf{TRAINING} \bowtie \textbf{CANDIDATE}) \bowtie \textbf{TERM})$$

4. *What training for a category 'B' driver's license did the instructor with the code 'I007' carry out?*

$$\Pi_{\text{code\_t, number\_hours, price\_per\_hour}} (\sigma_{\text{code\_ins} = \text{'I007'} \wedge \text{for\_category} = \text{'B'}} (\textbf{TRAINING} \bowtie \textbf{INSTRUCTOR}))$$

5. *With how many negative points did the candidate with EMBG = '1212999440048' pass the 'polygon' exam?*

$$\Pi_{\text{neg\_points}} (\sigma_{\text{name\_exam} = \text{'polygon'}} (\sigma_{\text{embg\_k} = \text{'1212999440048'} \wedge \text{outcome} = \text{'passed'}} (\textbf{TAKES} \bowtie \textbf{CANDIDATE})) \bowtie \textbf{EXAM})$$

6. *Which instructors worked at the "SEMAFOR" driving school from Prilep in the period from 2000 to 2010?*

$\Pi_{\text{name\_ins, surname\_ins}} (\sigma_{\text{name\_ds = 'SEMAFOR'} \wedge \text{city = 'Prilep'}} ($

$\sigma_{\text{from\_date} \geq \text{'01.01.2000'} \wedge \text{to\_date} \leq \text{'31.12.2010'}} (\textbf{WORKS\_FOR}) \bowtie \textbf{DRIVING\_SCHOOL}) \bowtie \textbf{INSTRUCTOR})$

7. *What total amount should the candidate with EMBG = '1810997440004' pay for the training?*

$\rho_{\text{total\_sum}} (\Pi_{\text{number\_hours} \times \text{price\_per\_hour}} ($

$\sigma_{\text{embg\_k = '1810997440004'}} (\textbf{TRAINING} \bowtie \textbf{CANDIDATE})))$

8. *What training has been carried out with the 'truck' type vehicles?*

$\Pi_{\text{code\_t}} (\textbf{TRAINING} \bowtie \sigma_{\text{type\_of\_vehicle = 'truck'}} (\textbf{VEHICLE}))$

6. CONCLUSION AND RECOMMENDATIONS

The proposed conceptual and logical model of the relational database, whose primary goal is to satisfy the needs of employees and customers of driving schools for timely, accurate, and complete information, are generic and are applicable, in their entirety or with minor design changes, for every driving school. The provided solution is independent of the specific RDBMS in which the RDB will be physically implemented, as well as the chosen hardware platform and operating system. The solution offers a large number of functionalities, which can be easily supplemented, but also it can be improved in terms of the quantity and quality of the data to be stored, kept, and processed in the system. The proposed data model can be implemented as a relational database within a standalone software application or can become an integral part of a specific information system, in a networked or online environment. User-defined queries, formally specified through relational algebra syntax, allow obtaining a wide variety of information based on the stored data. In addition, relational algebra allows for composing the most complex queries. However, the syntax of relational algebra does not allow specifying aggregate functions (SUM, AVG, MIN, MAX, COUNT), sorting tuples in the resultant set, or grouping them. All these limitations of relational algebra are overcome in the SQL language. Among other important constraints, which are a direct consequence of the flexibility and openness of such an approach, are the potential increase in the complexity of the database structure when adding additional tables and possible issues regarding the maintenance and administration of the database over time. Additionally, the increased volume of data in the relational database can result in decreased system performance. In such cases, the denormalization process can be applied to one or more derived relations that are currently normalized and are in the Third Normal Form (3NF) as part of the relational schema. Future research activities will focus on further refining the existing design of the RDB and enriching it with new entity types, attributes, and relationships. Additionally, defining additional user queries will be part of the efforts aimed at improving and enhancing the functionalities offered by the RDB. In the context of the previous, and taking into account the numerous benefits of using relational databases in terms of a high level of data integration, simultaneous and shared access, minimal data isolation and inconsistency, high processing performance, increased economy, reduced operational costs, etc., which are incomparably more essential than all their limitations and shortcomings, the recommendation is self-imposed, that all business entities that have not yet stepped more seriously into the world of digitization should do so as soon as possible so that their entire business operation will base it on the use of information systems with relational databases. For this reorientation of business activities to be successful for any business entity, the basic prerequisites are changing previously adopted habits and ways of thinking, acquiring basic information literacy, education, and technical culture, but also the desire and persistence of the latest technical-technological achievements, especially those in the field of ICT, to be put to the function of business operations. Only in this way, through a comprehensive digital transformation of business entities and the entire society, the work of organizations will be significantly facilitated and improved, which will result in maximum organizational effectiveness, efficiency, and performance, while the costs and use of vital resources (time, people, money, energy, repair materials, raw materials, ...) will be reduced to a minimum.

REFERENCES

[1] 101Computing.net (2022). Driving School – Entity Relationship Diagram (ERD). URL: https://www.101computing.net/driving-school-erd/

[2] Chen, P. P.-S. (1976). The Entity-Relationship Model – Toward a Unified View of Data. *ACM Transactions on Database Systems*, Vol. 1, No. 1, pp. 9–36.

[3] Codd, E. F. (1970). A Relational Model of Data for Large Shared Data Banks. *Communications of the ACM*, Vol. 13, No. 6, pp. 377–387.

[4] Elmasri, R. & Navathe, S. B. (2016). *Fundamentals of Database Systems*, Seventh Edition. Pearson Higher Education, Hoboken, NJ, USA. ISBN-13: 978-0-13-397977-7

[5] iNetTutor.com (2021). How to Create Database for Driving School Management System. URL: https://www.inettutor.com/mysql/how-to-create-database-for-driving-school-management-system/

[6] iNetTutor.com (2022). Driving School System ER Diagram. URL: https://www.inettutor.com/diagrams/driving-school-system-er-diagram/

[7] Ishak, M. H. B. (2012). Driving School Information System (DriSIS). Bachelor thesis. Faculty of Systems Computer & Software Engineering, University of Malaysia Pahang, Pahang, Malaysia. URL: http://umpir.ump.edu.my/id/eprint/7023/1/DRIVING_SCHOOL_INFORMATION_SYSTEM_(DriSIS).pdf

[8] Kher, S. (2015a). Database Model for a Driving School's Reservation System. Part 1. URL: https://vertabelo.com/blog/database-model-for-a-driving-school-reservation-system/

[9] Kher, S. (2015b). Database Model for a Driving School's Reservation System. Part 2. URL: https://vertabelo.com/blog/database-model-for-a-driving-school-reservation-system-part-2/

[10] Mithrasena, A. H. R. P. (2020). TrainMe - A Web-based Management System for Driving School. Master thesis. University of Colombo School of Computing, Colombo, Sri Lanka. URL: https://dl.ucsc.cmb.ac.lk/jspui/bitstream/123456789/4499/1/2017%20MIT%20046.pdf

[11] MSD (2013). Manhattan School of Driving LLC, Management System. URL: http://csis.pace.edu/~marchese/CS389/L10/ElaborationDocument_rev_389.pdf

[12] Tamilarasu, B. (2019). Driving School Monitoring System. Bachelor thesis. Department of Computer Technology, Bharathiar University, Coimbatore, India. URL: https://www.academia.edu/38538776/DRIVING_SCHOOL_MONITORING_SYSTEM_

[13] Zhou, J., Cai, N., & Liu, X. (2016). Design and Implementation of Driving School Information Management System Based on Dreamweaver. *Proceedings of the 6th International Conference on Sensor Network and Computer Engineering* (ICSNCE 2016), Xi'an, China, pp. 510–514. DOI: 10.2991/ICSNCE-16.2016.