UNIVERSITY OF ZAGREB
FACULTY OF ORGANIZATION AND INFORMATICS
VARAŽDIN

Andrija Krmpotić
Mislav Vetma
Sara Sušac
Theric Pheron

APPLICATION FOR PREGNANCY TRACKING

PROJECT

THEORY OF DATABASES

Varaždin, 2023

UNIVERSITY OF ZAGREB

FACULTY OF ORGANIZATION AND INFORMATICS

V A R A Ž D I N

Andrija Krmpotić
Mislav Vetma
Sara Sušac
Theric Pheron

Student ID:

Programme: Baze podataka i baze znanja

APPLICATION FOR PREGNANCY TRACKING

PROJECT

Mentor:

Full Prof. Markus Schatten, PhD

Varaždin, january 2023

Andrija Krmpotić
Mislav Vetma
Sara Sušac
Theric Pheron

Statement of Authenticity

Hereby I state that this document, my Project, is authentic, authored by me, and that, for the purposes of writing it, I have not used any sources other than those stated in this thesis. Ethically adequate and acceptable methods and techniques were used while preparing and writing this thesis.

The author acknowledges the above by accepting the statement in FOI Radovi online system.

Abstract

This project revolves around designing and implementing a database along with its interface toward users. The topic of the project is tracking pregnancy with all its checkups and things related to measuring the fetus's health along with its mother's health. The theoretical part of the project will be the design of the whole database and the practical part will revolve around the implementation of that same database and interface for controlling it with CRUD operations. The focus of the database will be on the active and temporal components. The active component is expressed throughout the implementation of triggers in the database and the temporal component is expressed with the date and time components inside the relational tables.

Keywords: active database; temporal database; pregnancy; trigger; tracking; doctor; hospital; application;

Table of Contents

1. Introduction

The topic of this project assignment is the development of an application for monitoring the development of human pregnancies. At the same time, active and temporary databases will be used for the functioning of this task in PostgreSQL, and a desktop graphical interface developed in the programming language C#. Alone the application was developed in the Visual Studio environment, i.e. in the form of a Windows Forms Application. The application is based on recording pregnant women who come to the hospital/clinic and monitoring the process of pregnancy through ultrasound, CTG, and other checkups with partial automation triggers. The application is designed to be used in a hospital by a doctor or someone else responsible person had access to the records of pregnant women, ultrasound, and other entities used in this domain. The pregnant woman will also have permission to see her data in the application but she will not be able to change or delete it.

2. Domain description

Here we need to describe what happens during the pregnancy period for a better understanding of it. Also we need to list all the checkups that are being done during pregnancy. and also we need to list all of the tables that we have here in the description and describe them accordingly

2.1. Person

In this case, the person can be the woman that is carrying the baby, the father, and the baby itself when it is born. This person is the foundation upon which this application is built. The attributes describing the person, in this case, are the following: id, name, surname, gender, date_of_birth, password, pregnancy_id (which refers to the table pregnancy_history), numberofbornchildren and the foreign keys to this table that represent the mother and the father of the person. These foreign keys can be null in case the parents are not known or if it goes too deep into the history of that family tree.

2.2. Pregnancy history

This table is the main temporal aspect of multiple pregnancies of the same person. All pregnancies are recorded inside this table by the ID of a person. This table is represented by the following attributes: id, beginningOfPregnancy, outcome, numberOfBornChildren, numberOfUnbornChildren, person_id, and doctor_id.

2.3. Hospital

This entity represents the location of the hospital and all the medical staff that is working inside it. The attributes describing the hospital are the following: id, name, address, city, and zip_code. This entity is made for the purpose of connecting the doctors with the physical location in the real world.

2.4. Doctor

The entity that goes by the name "Doctor" is the one that is performing and supervising all the checkups on the pregnant person. It is a very important entity because it is connected to all the other checkups and ultimately to the pregnant person itself. The "Doctor" is represented by the following attributes: id, name, surname, password, and the hospital in which the doctor works represented by a foreign key on the table Hospital.

2.5. PAP Test

A Pap smear, also called a Pap test, is a procedure to test for cervical cancer in women. A Pap smear involves collecting cells from the cervix — the lower, narrow end of the uterus that's at the top of the vagina. This test should be done by women every three years. In pregnancy, it is not necessarily needed if the woman has done it in three years period. The table for the PAP test is represented with the following attributes: id, outcome, date, time, person_id, doctor_id, and infirmary. The outcome attribute is described as positive or negative.

2.6. Gynecological tumors

There are 6 types of gynecological tumors that are known by today's date. These are cervical tumors, ovarian tumors, uterine tumors, vaginal tumors, vulvar tumors, and very rare fallopian tube tumors. In this case, they are separated in a separate table and represented by the following attributes: id, name, tumorDescription, and lethal. The lethal attribute is a bool type that shows whether the tumor can be lethal or not.
In the table DetectionOfGynecologicalTumors, there are the following attributes: id, tumor, gynecologicaltumors_id, date, time, infirmary, person_id, and doctor_id. These attributes represent the checkup that determines whether the pregnant woman has a tumor or not. It is crucial to do this checkup to determine the safety of the baby and the safety of the pregnant woman.

2.7. Urinary incontinences

There are 5 types of urinary incontinence known today. These are stress incontinence, urge incontinence, overflow incontinence, functional incontinence, and mixed incontinence. They are separated in a table for themselves that goes by the name "Incontinences" and are described by the following attributes: incontinence_id, incontinence_name, incontinence_description.
The table UrinaryIncontinence represents the checkup done by the doctor on an ordinary person (not just a pregnant person). The test is done by inserting a tube into the urethra and then releasing the urine and measuring the pressure that it produces. Then from that pressure, the doctor can determine whether the person has incontinence or not and what type it is if it's present. This table is represented by the following attributes: urinaryIncot_id, incontinence, incontinence_id, date, time, infirmary, doctor_id, and person_id.

2.8. Blood Tests

Blood tests are done regularly to see all types of parameters of all people, not just pregnant women, and they also show all kinds of things such as sugar levels or hemoglobin levels in the blood. All these tests are not just to determine the health of a pregnant woman but also to determine the health of the fetus inside the womb. They are connected to parameter

types via many-to-many relation which results in a new table called parameters. The blood test table is represented by the following attributes: id, analysis, time, date, infirmary, person_id, and doctor_id.

2.8.1. Parameter types

This table is considered a list of all the parameters that are checked with a blood test on a person. This table represents all the types of parameters that are possible to check and it is represented by the following attributes: id, name, description, and unit.

2.8.2. Parameters

Table parameters is here to enable the search of specific blood tests done by a specific person in a specific period of time. In this table are records of all the parameters and their result in a blood test of one person. This table is represented by the following attributes: BloodTest_id, Type_id, result, and result_description.

2.9. Color Doppler test

This test is a specific type of ultrasound test that converts the waves into pictures. It is mostly used to check for issues with blood flow (to search for blood cloths inside the vein or artery system. The table is represented with the following attributes: id, bloodFlow, umbilicalArtery, centralCerebraArtery, fetalAorta, umbilicalVein, infirmary, doctor_id, and person_id.

2.10. Ultrasound

Ultrasound is a procedural checkup done on pregnant women every three weeks. This test shows the progression of the fetus inside the woman and is considered one of the vital checkups for tracking the pregnancy of a woman. It is represented with the following attributes: id, date, time, infirmary, description, doctor_id, person_id, and picture that represents the scan od the fetus in the current state.

2.11. CTG

A cardiotocograph (CTG) is a form of Doppler that uses only sound and doesn't produce an image. The simple, painless, and non-invasive procedure is done during pregnancy to check the baby's condition. The test is also known as a 'non-stress test (NST)'. This is because the baby is not under the 'stress' of labor, and nothing has been done to put stress on it. During the test, the doctor monitors the baby's heartbeat, first while the baby is resting and then while it's moving. Just as your heart beats faster when the woman is active, the baby's heart rate should

go up while it's moving or kicking. In this case, it is represented by the following attributes: id, date, time, doctor_id, person_id, infirmary, and descript.

2.12. Gynecological examination

A gynecological examination is one of the main checkups in the pregnancy and it is done by a gynecologist to determine that everything is all right with the woman's vagina. This table is represented with the following attributes: id, date, time, diagnosis, descript, infirmary, doctor_id, and person_id.

3. Theoretical introduction

3.1. Active databases

Active databases are databases that are based on relational databases, and with active databases, it is specific that they include active rules that are in the form of ECA (Eng. Event - Condition - Action). These databases enrich traditional relational databases with mechanisms that help in processing rules. Applications of active databases can include:

- advanced conditions of integrity
- views
- statistical monitoring
- workflow management etc.

An active database can be implemented in two different ways:

- by active rules (CREATE RULE)
- by triggers (CREATE TRIGGERS)

The implementation of triggers consists of two separate parts. Defining the trigger function and defining the trigger itself. When the trigger function is made it needs to be connected to some kind of trigger that starts when some event occurs inside the database. Possible times for triggers to activate are BEFORE or AFTER some event, and possible events in PostgreSQL are INSERT, DELETE, and UPDATE. All trigger functions have the same return type and that is the TRIGGER type.

3.2. Temporal databases

Temporal databases are specific types of relational databases that have temporal aspects built into them and they enable the work with this kind of data via temporal SQL language. These databases most likely include values like the following:

- **valid time** - time that represents some factual event that happened that is in the applicational domain
- **transaction time** - time that represents the moment when some event is logged into the database
- **bitemporal time** - any time that represents the combination of the two times listed above

Temporal types that PostgreSQL can use are the following:

- **time (timetz)** - time in a day (with a timezone)
- **date** - a simple date
- **timestamp (timestampz)** - a moment in time that includes date and time (with the timezone)
- **tsrange** - defined interval of time between two events
- **tstzrange** - defined interval of time between two events with a timezone included
- **daterange** - defined time interval between two dates

It is important to note that date type or timestamp type can be added or subtracted by any integer type of data. This will be used in this project in triggers to automatically appoint some checkups that are repetitive. Therefore it will be a combination of temporal and active databases to achieve some specific events.

4. Database model

Database model that is related to this specific case will be described in the form of an Entity-Relationship diagram. The model itself has been described in the previous chapter. Picture number 1. is the ERA diagram for this case.

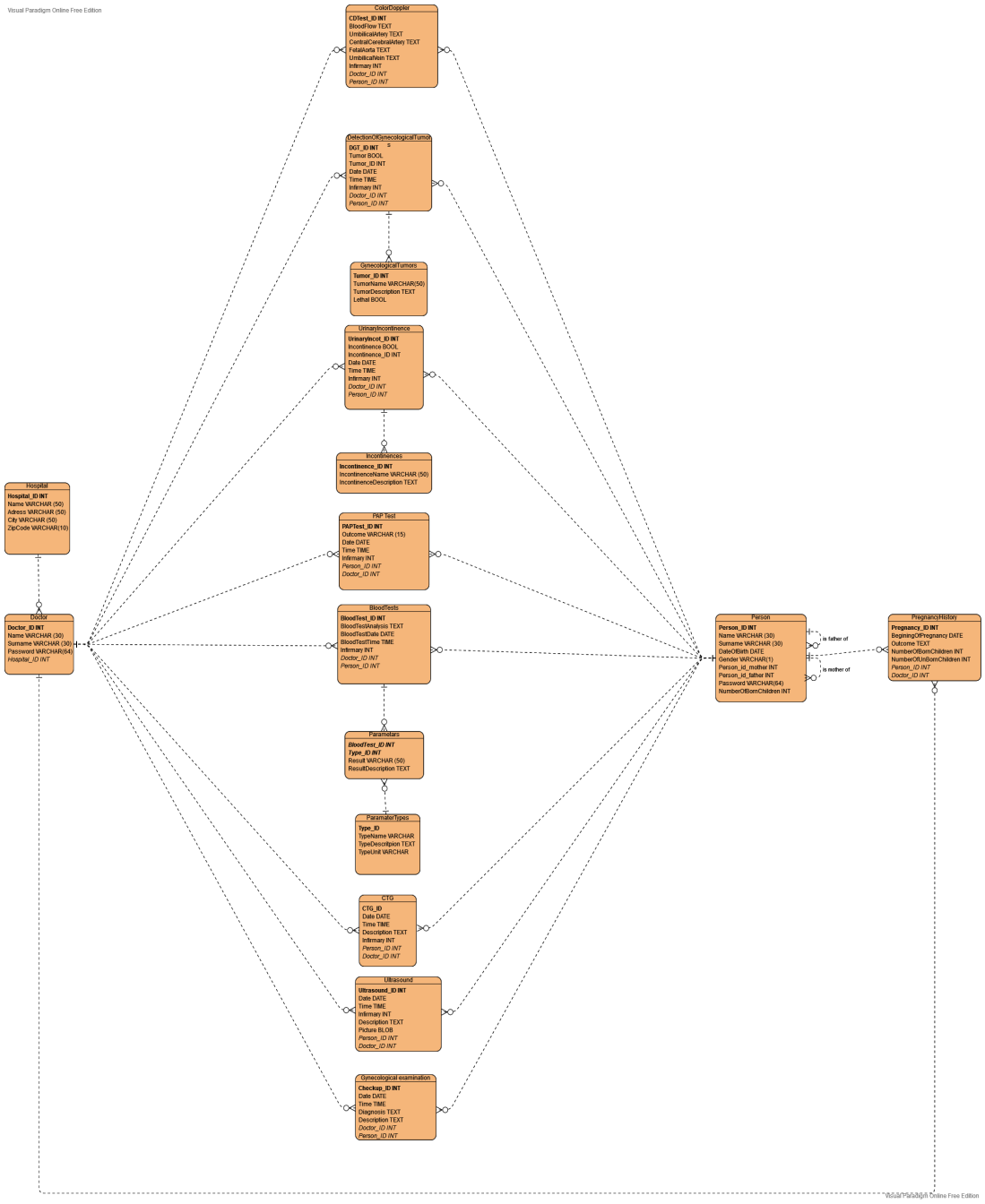

Figure 1: ERA model of the database

5. Implementation

In the script called TBP-SCRIPT_all.sql is written all that is needed to deploy the database on any server. The creation of the tables that are listed above, the trigger functions, and the triggers themselves are also implemented inside the script. It is important to note that all the insert statements are also written inside that script.
The list of triggers that are implemented in the database is the following:

- The year of birth cant be less than 1900
- Check that the parents exist in the list of registered persons (if the parents are not registered in the database, we put id 0 for the mother and id 1 for the father)
- Check the gender of the parents when registering (a father cannot be female, mother can't be male)
- Check that the date of the test that is being recorded is not before the date of the last test.
- Check that the gender of the person entered for the test is female
- Verifying that the password has a capital letter, a small letter, a number, and has at least 8 characters
- Display the estimated date of the next test according to the progress of the pregnancy
- Check that the test date is not before the date of the beginning of the pregnancy
- Check that the test date is not in the future
- Remove the number of babies waiting to be born in pregnancy after the birth
- After registration send back the ID of the doctor or person

The database is implemented in the pgAdmin tool and is locally hosted by the user for the time being. The tool itself provides all that is needed for the user to host the database and is compatible with PostgreSQL. The interface for the database is coded in the C# programming language and the use of that interface will be shown in the following chapter of this paper.

6. Use cases

7. Conclusion

This application wouldn't be able to work if there were no active or temporal aspects. The triggers that are implemented in this project serve as the aspect of active databases while the time component, such as dates and times, serves as the aspect of temporal databases. Through this project, we learned how to effectively use the triggers as well as the temporal aspect of the database. The application is ready for use by doctors and pregnant women, but there is a lot more space for improving the state of this application by adding more functionalities and improving the scalability and speed of the database.

List of Figures