**Εργασία 1η Bonus**

Όνομα: Βασίλειος Παπουτσόπουλος

ΑΜ: 144172

Θέμα: Εφαρμογή σε για Κράτηση επισκέψεων/ραντεβού σε μια ιδιωτική κλινική

Υλοποίηση: Java Swing στο περιβάλλον Netbeans IDE

για τη γραφική διεπαφή χρήστη

Βάση δεδομένων αποθηκευμένη σε Oracle (στον SQL

Developer)

\*σημείωση: σκέφτομαι προς το παρόν να το δουλέψω στον SQL Developer

*Λίγα λόγια για τη Βάση Δεδομένων της εφαρμογής:*

Αρχικά, η εκκίνηση της εφαρμογής θα γίνει με την επιλογή

1. Administrator ή
2. Απλού Χρήστη

Και ανάλογα με την επιλογή αυτή, ο απλός χρήστης ή ο administrator θα έχει και τις ανάλογες δυνατότητες.

Πιο συγκεκριμένα, ένας administrator θα έχει τις εξής δυνατότητες:

* Θα μπορεί να βάζει, να τροποποιεί, ή να διαγράφει τα ραντεβού με κάποιον γιατρό αναλόγως το όνομα και το επάγγελμα του (π.χ. Οφθαλμίατρος, Καρδιολόγος, Πνευμονολόγος, κλπ)
* Μπορεί να βάζει ραντεβού παραπάνω από μία φορά στο όνομα κάποιου πελάτη, αλλά διαφορετικές ώρες και ημερομηνίες του προγράμματος
* **ΔΕΝ** θα επιτρέπει στους πελάτες να κλείσουν ραντεβού την ημερομηνία και την ώρα που είναι κρατημένο από κάποιον άλλον πελάτη

Από την άλλη, ο απλός χρήστης θα έχει τις εξής δυνατότητες:

* Δε θα μπορεί να κλείσει ραντεβού
* Θα μπορεί όμως να δει τα ραντεβού τα οποία έχουνε γίνει

\*σημείωση : Επίσης σκέφτομαι να βάλω κι ένα Search ανάλογα με το επώνυμο αυτού που έκλεισε το ραντεβού, ή ανάλογα με το ονοματεπώνυμο του γιατρού, κάτι τέτοιο, δεν έχω αποφασίσει ακόμα

*Ακολουθεί η κανονικοποιημένη μορφή των πινάκων που θα περιέχει η Βάση Δεδομένων:*

Asthenis (IDAstheni, Epwnymo, Onoma, Thlefwno, Fylo, Username)

Giatros (IDGiatrou, Epwnymo, Onoma, Thlefwno, Eidikothta)

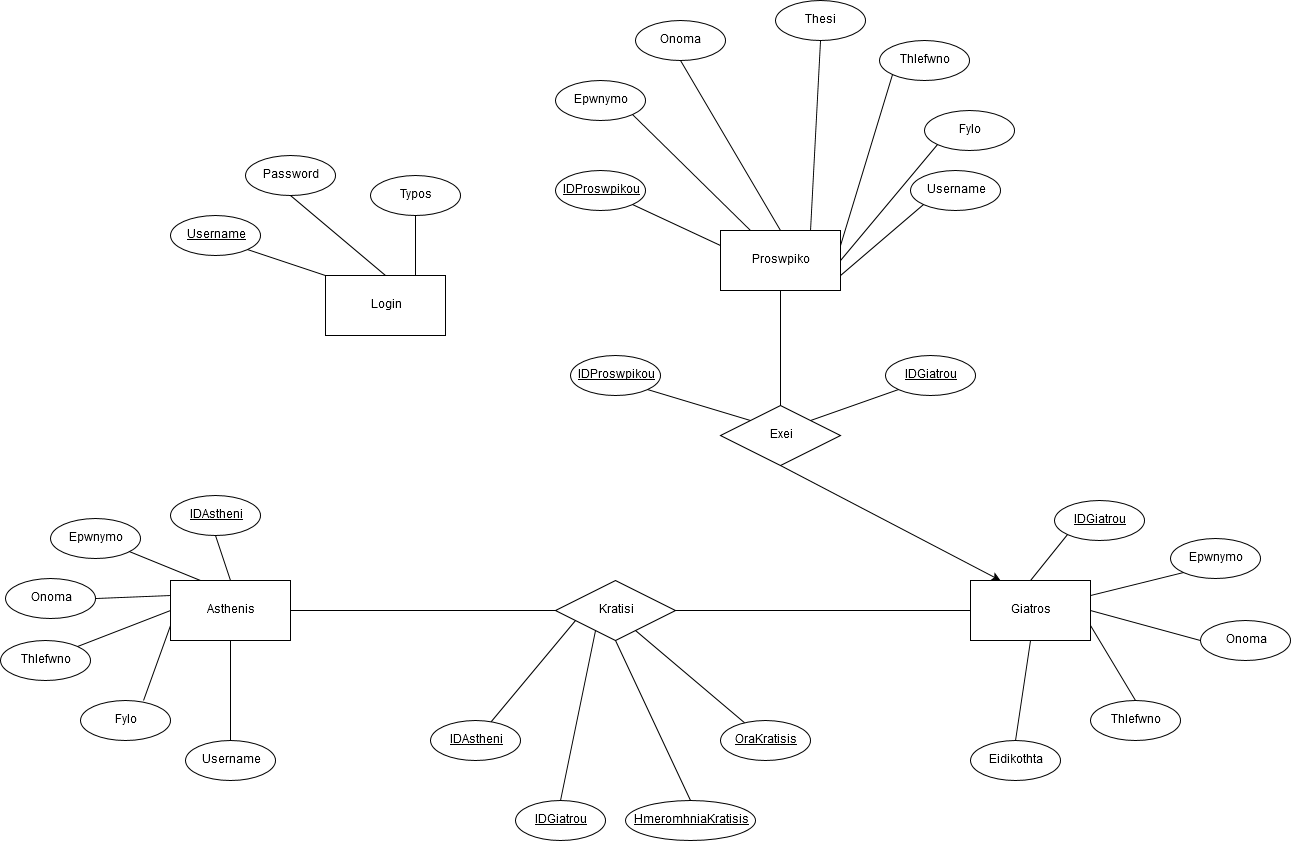
Kratisi (IDAstheni, IDGiatrou, HmeromhniaKratisis, OraKratisis)

Login (Username, Password, Typos)

Proswpiko (IDProswpikou, Epwnymo, Onoma, Thesi, Thlefwno, Fylo, Username)

Exei (IDProswpikou, IDGiatrou)

*Διάγραμμα ER*



\*σημείωση: Το διάγραμμα το σχεδίασα μέσω της ιστοσελίδας <https://www.draw.io/>

Επίσης, προς το παρόν σκέφτομαι να φτιάξω τους πίνακές μου κάπως έτσι:

drop TABLE LOGIN;

drop TABLE ASTHENIS;

drop TABLE PROSWPIKO;

drop TABLE GIATROS;

drop TABLE KRATISI;

drop table EXEI;

create TABLE Login(Username VARCHAR(15) PRIMARY KEY,

Password Varchar(8),

Typos Varchar(10) check (Typos in ('Ασθενής','Προσωπικό')));

create TABLE Asthenis (IDAstheni Integer NOT NULL PRIMARY KEY,

Epwnymo Varchar(25),

Onoma Varchar(20),

Thlefwno Varchar(10),

Fylo Varchar(10) check (Fylo in ('Άνδρας','Γυναίκα')),

Username VARCHAR(15),

CONSTRAINT a\_Username foreign key(Username) references Login);

/\* Κάνω ένα sequence και ένα trigger για να αυξάνεται αυτόματα το IDAstheni \*/

CREATE SEQUENCE asthenis\_sequence

START WITH 1

INCREMENT BY 1;

CREATE OR REPLACE TRIGGER ASTHENIS\_TRIGGER

BEFORE INSERT ON ASTHENIS FOR EACH ROW

BEGIN

:new.IDAstheni := asthenis\_sequence.nextval;

END;

create TABLE Proswpiko (IDProswpikou Integer NOT NULL PRIMARY KEY,

Epwnymo Varchar(25),

Onoma Varchar(20),

Thesi Varchar(40),

Thlefwno Varchar(10),

Fylo Varchar(10) check (Fylo in ('Άνδρας','Γυναίκα')),

Username VARCHAR(15),

CONSTRAINT p\_Username foreign key(Username) references Login);

/\* Κάνω ένα sequence και ένα trigger για να αυξάνεται αυτόματα το IDProswpikou \*/

CREATE SEQUENCE PROSWPIKO\_SEQUENCE

START WITH 1

INCREMENT BY 1;

CREATE OR REPLACE TRIGGER PROSWPIKO\_TRIGGER

BEFORE INSERT ON PROSWPIKO FOR EACH ROW

BEGIN

:new.IDProswpikou := PROSWPIKO\_SEQUENCE.nextval;

END;

create TABLE Giatros (IDGiatrou Integer NOT NULL PRIMARY KEY,

Epwnymo Varchar(25),

Onoma Varchar(20),

Thlefwno Varchar(10),

Eidikothta Varchar(40));

/\* Κάνω ένα sequence και ένα trigger για να αυξάνεται αυτόματα το IDGiatrou \*/

CREATE SEQUENCE GIATROS\_SEQUENCE

START WITH 1

INCREMENT BY 1;

CREATE OR REPLACE TRIGGER GIATROS\_TRIGGER

BEFORE INSERT ON GIATROS FOR EACH ROW

BEGIN

:new.IDGiatrou := Giatros\_SEQUENCE.nextval;

END;

create TABLE Exei (IDProswpikou Integer NOT NULL,

IDGiatrou Integer NOT NULL,

primary key (IDProswpikou, IDGiatrou),

CONSTRAINT d\_IDProswpikou foreign key(IDProswpikou) references Proswpiko,

CONSTRAINT d\_IDGiatrou foreign key(IDGiatrou) references Giatros);

create TABLE Kratisi (IDAstheni Integer NOT NULL,

IDGiatrou Integer NOT NULL,

HmeromhniaKratisis Date,

OraKratisis Varchar(5) CHECK (OraKratisis LIKE '\_\_:\_\_'),

primary key (IDAstheni, IDGiatrou, HmeromhniaKratisis, OraKratisis),

CONSTRAINT k\_IDAstheni foreign key(IDAstheni) references Asthenis,

CONSTRAINT k\_IDGiatrou foreign key(IDGiatrou) references Giatros);

\*σημείωση: Σκέφτομαι σύμφωνα με τους πίνακες που να δώσω τις ιδιότητες του Απλού Χρήστη στον Ασθενή και του Administrator στο Προσωπικό.