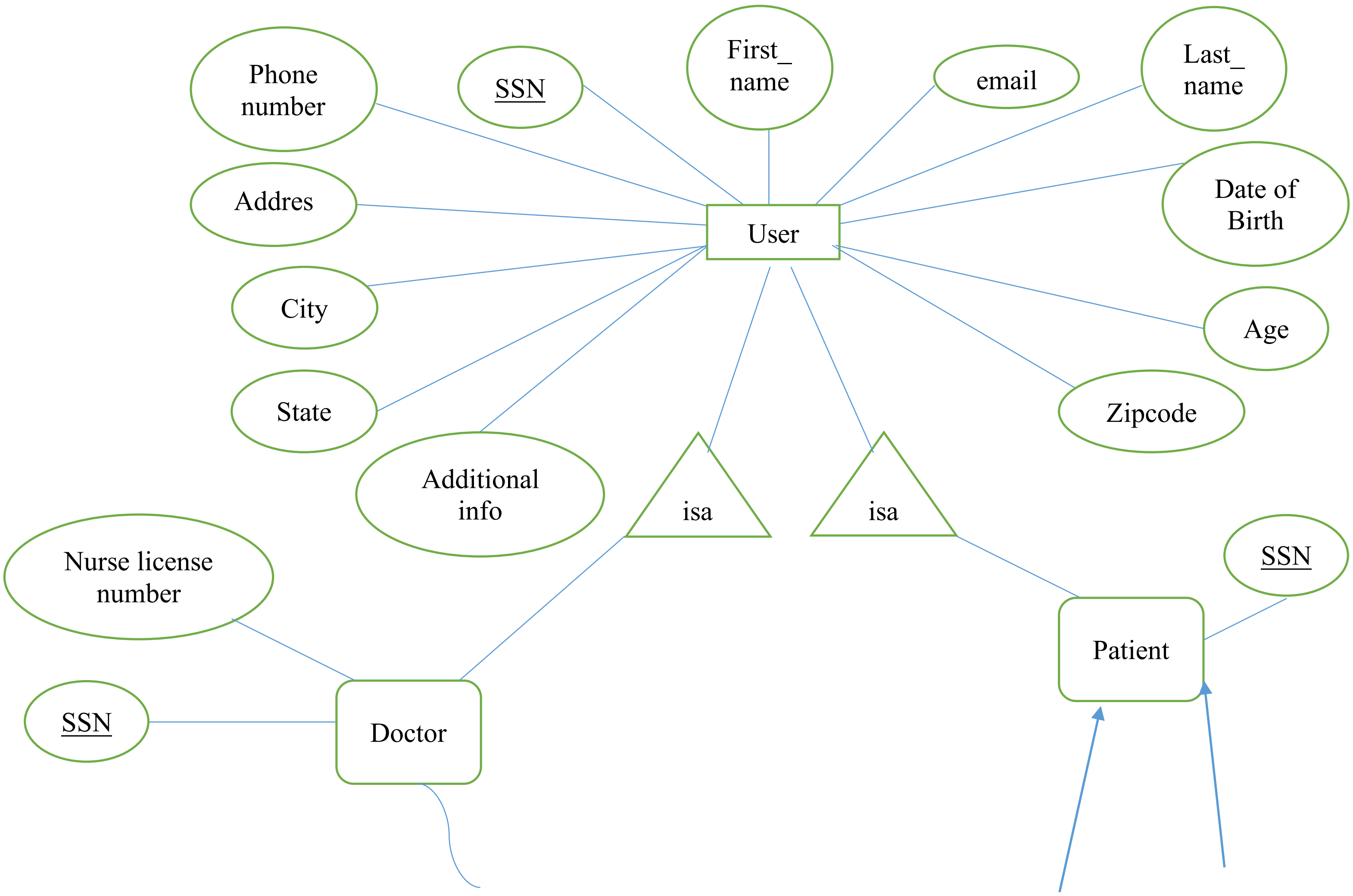
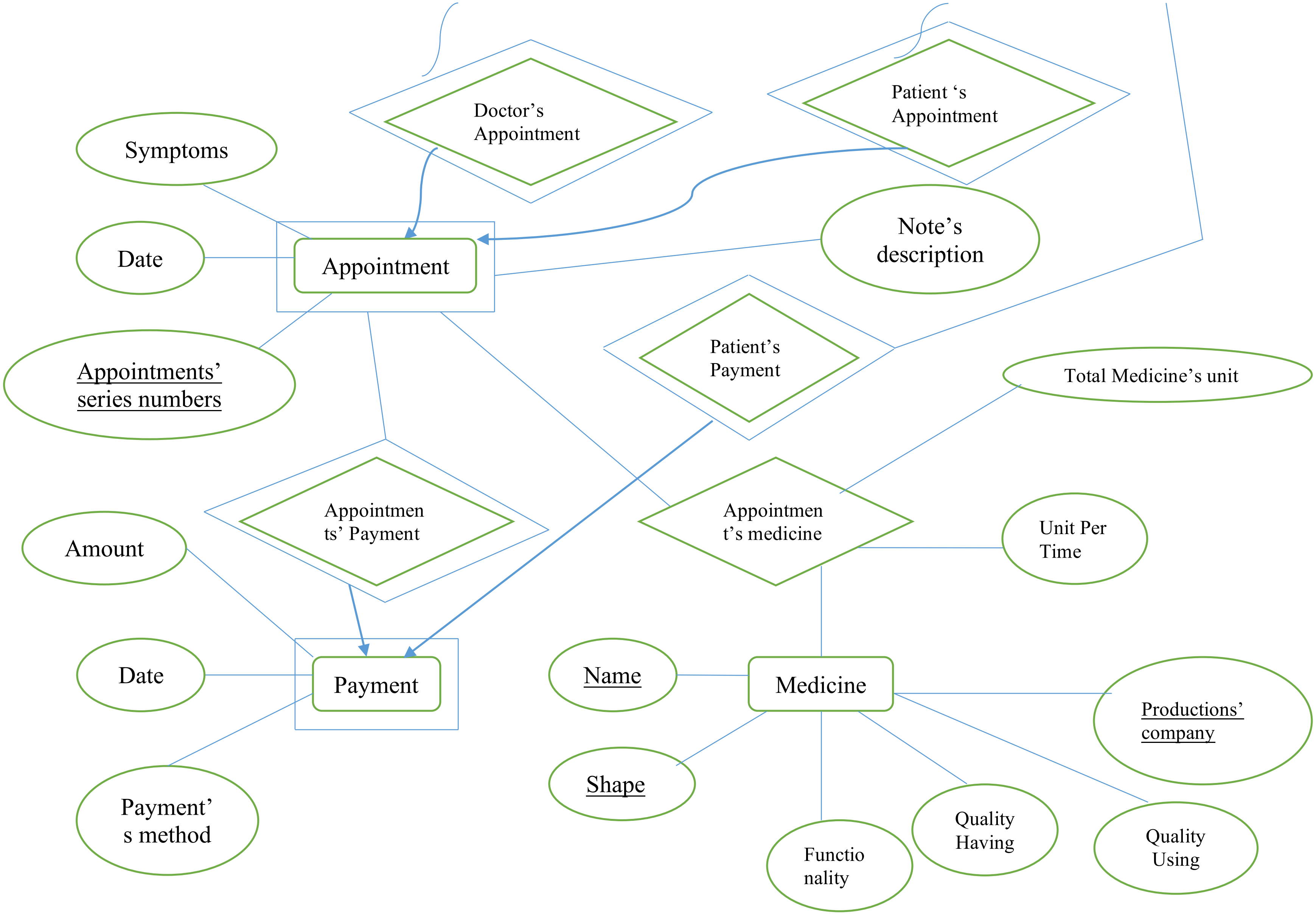
Name: Minh Vu, Chien-Hsiung Chao
Group: 5

This is our database Entity Relation




Doctor's Appointment
Patient 's Appointment
Symptoms
Date
Appointment
Note's description
Appointments' series numbers
Patient's Payment
Total Medicine's unit
Appointments' Payment
Appointment's medicine
Unit Per Time
Amount
Date
Payment
Name
Medicine
Productions' company
Shape
Payment's method
Quality Having
Quality Using
Functionality

In our database, we have some tables:

User (ssn, first_name, last_name, age, email, address, city, state, zip_code, phone, date_of_birth)
Doctor (*doctor_ssn, doctor_license_number)*
Doctor_Specialist (*doctor_ssn, doctor_specialist_subject)*
Patient (patient_ssn)
Appointment (*appointment_series_number, doctor_ssn, patient_ssn, symptoms, date, note*)
Payment (*appointment_series_number, date, total_amount_of_money, payment's method, paying_amount_of_money, rest_amount_of_money)*
Medicine (*medicine_name, medicine_shape, medicine_producer , functionality, quanlity_having, quanlity_using)*
Appointment_Medicine (*appointment_series_number, medicine_name, medicine_shape, medicine_producer, unit_per_time, total_amount_of_medicine)*

In this database we assume,

Any appointment record has the same symptoms, date and note's description at the same time.
Any appointment record has only one patient
Any payment record has to input in the different date

Based on this database, we could a minimal Functional Dependency:

User relaction:

ssn->first_name, last_name, age, email, address, city, state, zip_code, phone, date_of_birth

Doctor relation:

doctor_ssn → doctor_license_number, doctor_specialist_subject

This is a violation because doctor_license_number is unique and it could generate every attribute, therefore this is a candidate key and it violates 3NF. To eliminate this violation, we spread this into 2 relation

doctor_license_number: *doctor_ssn*, doctor_license_number

Both of these attribute is the candidate key, but we choose doctor_ssn is a primary key

doctor_specialist_subject: *doctor_ssn*, doctor_specialist_subject

Appointment relation:

appointment_series_number → doctor_ssn, patient_ssn, symptoms, date, note_description

Payment relation:

appointment_series_number, date → total_amount_of_money, payment's method, paying_amount_of_money, rest_amount_of_money

Medicine relation:

medicine_name, medicine_shape, medicine_producer → functionality, quanlity_having, quanlity_using

Appointment_Medicine relation:

appointment_series_number, medicine_name, medicine_shape, medicine_producer → _unit_per_time, total_medicine’s unit