

HOSPITAL MANAGAMENT SYSTEM

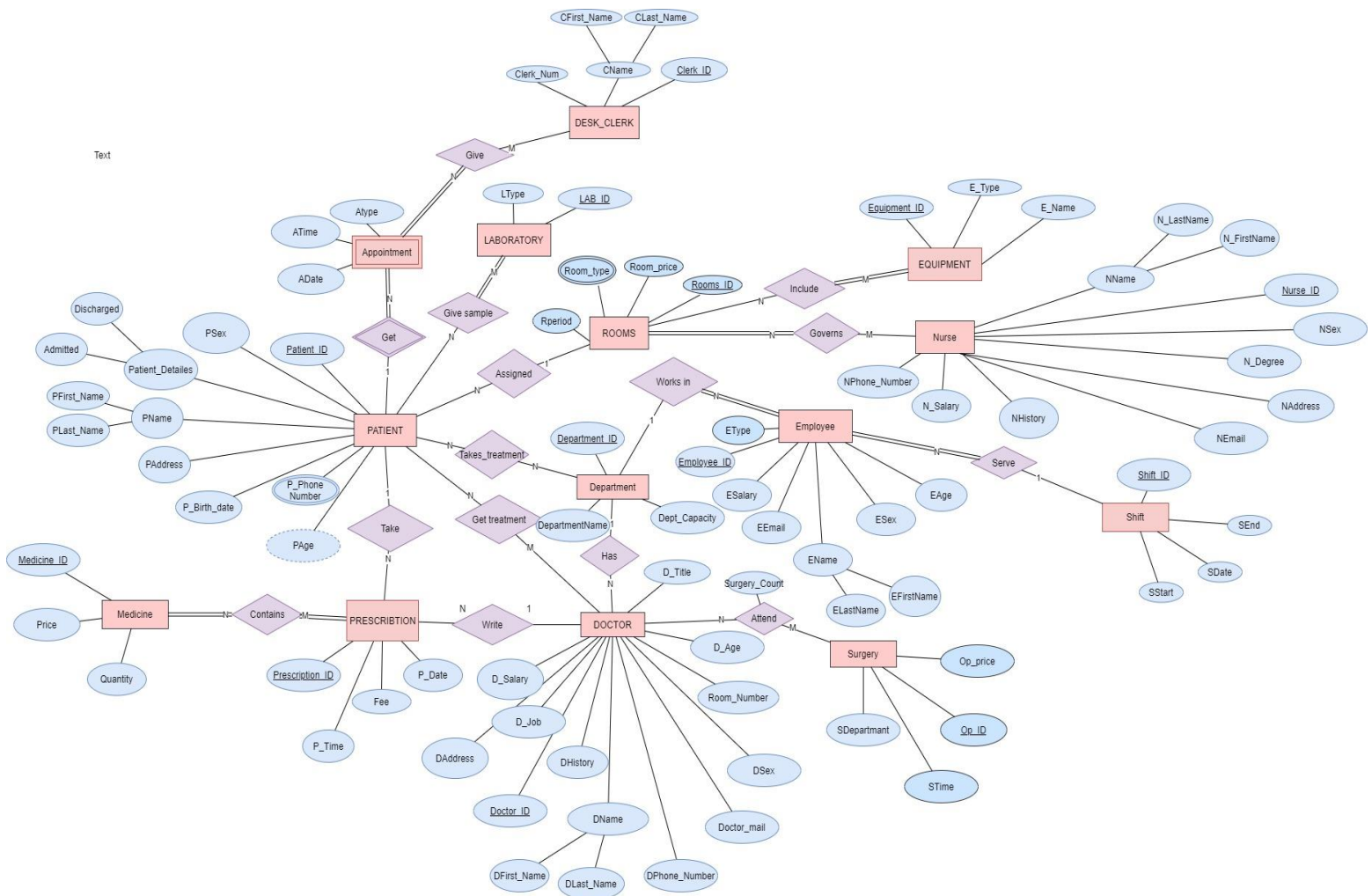
DESIGN DOCUMENT - GROUP 7

Zülal Karın 12622989076

Zeynep Ayca Tanışlı 33326013720

Ayfer Feyza Şahin 30932496534

HOSPITAL MANAGEMENT SYSTEM ER DIAGRAM



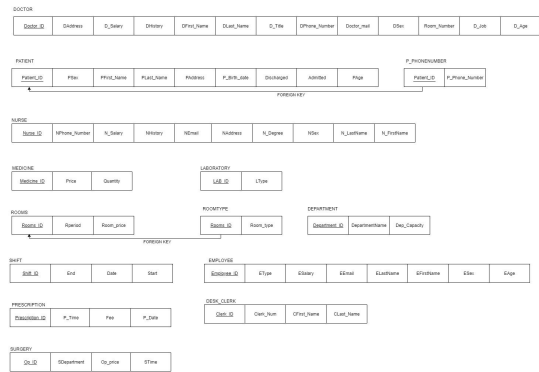
SPECIFICATION OF THE DOMAIN

- 1) We identified each patient with **Patient_ID**. We made sure that each row is different by using a Primary Key. Each patient has name, address, sex, phone number birth date, age, and details (for discharge procedures). Patients get appointment and take prescription. We defined the age of the patient as the derived attribute because we keep it together with the date of birth. Patients Takes_treatments in departments and Get_Treatments from doctor. In addition, samples go to the laboratory from the patients, patients can be assigned to rooms and receive treatment from doctors. Thanks to these connections and relationships, we can clarify our first query. The patient can receive an appointment, choose MR as the appointment type, so we can list the patients who have an MR appointment. Thanks to Patient_Details, which we define as a composite attribute, we can reach hospitalized patients.
- 2) We defined prescriptions as **Prescription_ID**. Each prescription has its time, hour, and cost. Besides, they contain medicine.
- 3) We defined the medicine as **Medicine_ID**. Each medicine has price and quantity.
- 4) We defined the **appointment system** of the hospital as a weak entity and each appointment has a time, date, and type.
- 5) We designed a Desk Clerk that delivers these appointments. We have identified each receptionist with their ID numbers (**Clerk_ID**) and the receptionists have their names and numbers.
- 6) We defined the laboratories with **LAB_ID**. Each laboratory has a type. (Such as Clinical Biochemistry, Clinical Microbiology, Clinical Haematology and Pathology)
- 7) We identified the rooms where the patients are assigned with **Rooms_IDs**. There are periods, types, and fees to determine when rooms are available. Since more than one patient can stay in a room, the type of the rooms is multivalued attribute. Also, all rooms include equipment and nurse who governs rooms.
- 8) We identified the equipment in the hospital with **Equipment_ID**. Equipment has its type and name. Such as Sterilizer, Operating Table, Traction, Ceiling Lights, Air Blender, Gynaecological Table, Anaesthesia Device, Bedside Monitor ...
- 9) There are nurses assigned to rooms in our hospital and we identified these nurses with **Nurse_ID**. Each nurse has a name, surname, gender, number, salary, degree, address, and email. In addition, we added NHistory to assign nurses to cases related to it in the future.
- 10) A doctor identified by **Doctor_ID** each doctor has age, department, salary, address, history, name, surname, phone number, mail, title, sex, and room number. Also, a doctor write prescription and attend surgery. We put the Surgery_Count attribute for the Attend relation. In this way, we can count the surgeries requested from us in the second query from us.
- 11) We identified the surgeries as **Op_ID**. Surgeries have price, time, and departments.

- 12) We defined the employees by **Employee_ID**. Employees have their name, age, sex, email, salary, and type (cleaners, security guards, technicians ...). Employees work in various departments and serve shifts.
- 13) We have identified the departments with the **Department_IDs**. Each department has a name and capacity.
- 14) We defined the shifts with **Shift_ID**. Shifts have a start time (SStart), end time (SEnd) and date. (Since it is a start and end key word, we did not use them directly.)

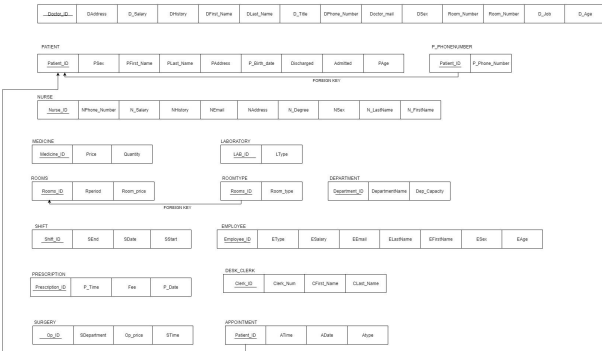
RELATIONAL SCHEMA

Step 1: Mapping of Regular Entity Sets



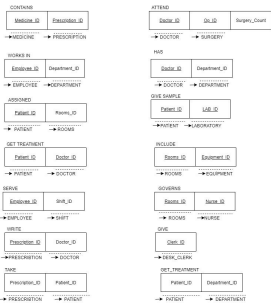
- In this step, firstly we set the strong entities in a relation R that contains all the basic attributes of E.
- Then we choose the key attributes of E as the primary key for R.
- If E is the chosen key as a composite (E (PFirst_Name, PLast_Name)), then the set of the simple attributes inside that form will together be the primary key of R.
- Our entity set has two multi-valued attributes (P_Phone Number, Room_type) M of E. we did not include them in the relation R. Moreover we create a foreign key constraint that includes the primary key attributes of E because we did not create a new relation schema RM which contains the M as the attribute and the primary key of E.

Step 2: Mapping of Weak Entity Sets



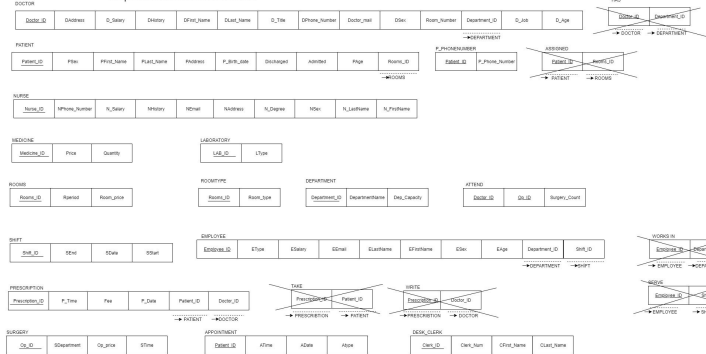
- We created one weak entity set W in the ER schema and then we include a relation R and it contains all simple attributes of W as attributes R.
- In addition, it includes foreign key attributes of R the primary keys of the relations that correspond to the owner entity set.
- The combination of the primary key of the owner and the partial key of the weak entity form W, if any, is the primary key of R.

Step 3: Mapping of Relationship Sets



- We created a relational schema for the relationship set R.
- We use all of the primary keys of contributing relation schemas in the relation schema's attributes, and if there are any informative attributes, we use them as well.
- Firstly we decide on the primary key:
 - 1:1 case: We need to choose the primary key of any of the two participating entity sets.
 - M: N case: We need to choose the union of all primary keys of the participating entity sets.
 - 1:N or N:1 case: We select the primary key in relation to the entity set having the N ratio.
 - N:ary case with one entity having 1 ratio: We select the union of all primary keys in the participating entity set with a cardinality ratio of less than one.
 - N:ary case: We select the union of all primary keys of the entity sets that are participating.
- We need to add related foreign key constraints, and for the part recursive relationship sets, the attribute names are based on the role names.

Step 4: Combination of Schemas



- For N:1 and 1:N relationships, where N is the total number of participants, by shifting its attributes, combine the schema corresponding to the relationship set into the entity set with the N cardinality ratio.
- Then we adjust the foreign primary constraint if necessary.
- In 1:1 relationship sets, either side may serve as the "many" side; however, choose the side with the most participation. Otherwise, NULL values may be present in any of the relation's tuples.

