



Class: INFX 543 A

Project Team 7: Pierre Augustamar, Jenny Mount, Tin-Chia Lin

Database Topic: Personal Health Care System

Database Purpose:

The purpose of the Personal Health Care System is to allow anyone to access and coordinate his/her health by storing into a database vital information such as blood pressure readings, weight gain or loss, etc. Doctors, researchers, and nurse practitioners can use the stored data to analyze and provide feedback to help users manage their health. In addition, researchers can also use the data to warn the local health department of possible outbreaks.

Business Problems Addressed:

The Personal Health Care System is designed to address multiple problems that have occurred in the world of health care. But at the core this system is intended to help a user to get the best care possible from experienced doctors and as well getting an opportunity to be part of any successful experiments. To that end, it addresses the followings:

- Allow a user to manage his/her care by being able to get real-time feedback from doctors.
- Allow a user to get treatments from any experienced doctors no matter where those users or those doctors are located.
- Allow multiple doctors to provide care whether paid or pro-bono to any users that need treatments.
- Allow insured and non-insured alike to get the same opportunities to be seen by experienced doctors.
- Allow insured and non-insured alike to get the same opportunities to be part of a clinical experiment that was proved to be successful.
- Allow researchers to connect the dots between family members that could be carrying ancestral genes of some fatal disease.
- Allow researchers to mine data and provide the health department data regarding the possibility of an outbreak based on a zip code, a location, ethnicity, occupation, or a particular drug.
- Allow both doctors and researchers to collaborate on finding a cure for a particular disease.
- Allow researchers to have real-time data regarding the benefits of an experiment as well as the use of an individual drug to treat a disease.

Business Rules:

The business rules for this database are set based on the needs of a user to get the best health care that can be found anywhere or anytime on the planet. Our system is agnostic of gender, location, occupation and holder of an insurance card. These rules are motivated by the actions that will be taken by the main actors of our system. In this case, users, doctors, and researchers are considered to be the key motivators for the development of this system. The rules for this system are:

- A user should be identified by one gender and one gender only. The user can also specify "unknown" if desired.
- A user should have been diagnosed as having one or more chronic diseases before entering data into the health system.
- A user may or may not have an insurance card.
- A user should have a zip code that identifies the region and or the city where he/she currently resides. In addition, a time column will identity the time when the user lived at a specific zip code.
- A user may or may not be taking a drug (s) to manage his/her disease.
- A user may or may not have an occupation. However, for the purpose of this exercise in cases where a user has multiple occupations, we will only take the primary occupation.
- A doctor can provide treatments to one or more users.
- One or more doctors can monitor a user.
- A user may be related to other users who are also in the health system database
- A user can be associated with multiple ethnic backgrounds.
- A user may or may not record any of his/her health readings.
- A user should be associated with one birth location.
- A user can be associated with multiple living areas.
- A doctor must have at least one medical specialization.
- A researcher must have one or more medical specialties.
- A researcher should be engaged in at least one clinical experiments.
- A doctor can be associated with zero or more insurance services
- A user may have one or more ethnicities

Entities Definition and Relations to Other Entities:

Entity	Definition	Relation To other Entities
Diagnose	Diagnose is a supporting table for mapping all the possible diagnosis that could occur to a user. For the purpose of this exercise, a user is considered to be already diagnosed of a disease before joining the health_syste. Thus, he/she is using the health system to be monitored by doctors as well as to be part of any	1. A diagnose is associated with one or more user_diagnosis (entity)

	possible experiments being taken by one or more researchers.	
Doctor	In the health system database, the doctor (main entity) is the person with the skills to monitor the health of a user. This doctor should have one or more specializations in the field of medicine and also, should have the experience to provide treatments to one or more users that are in the system. This doctor should have a unique id that identifies him/her from others. He/she may or may not be associated with a particular insurance company depending if this is a paid or pro-bono work. He/she should have an email, and a phone number for contact purposes. He/she should have at least two years of experience in a particular field of specialization.	<ol style="list-style-type: none"> 1. One or more doctors can provide treatments to a unique user (Entity) 2. A doctor should have one or more specializations (entity) in the field of medicine. 3. A doctor is associated with one or more treatments (entity) 4. A doctor should be able to monitor one or more users in the Health_System (Entity)
Doctor_Treatment	This entity is a supporting table for the main doctor entity. It was designed to avoid a many to many relationships between a doctor and the treatments that he/she can provide.	<ol style="list-style-type: none"> 1. Zero or more doctor treatments are associated with a doctor (entity) 2. Zero or more doctor treatments are associated with the treatment (entity) mapping table.
Doctor_Specialization	This entity is a supporting table for the main doctor entity. It was designed to avoid a many to many relationships between a doctor and the number of specializations that he/she possesses.	<ol style="list-style-type: none"> 1. One or more doctor specializations are associated with a doctor (entity) 2. One or more doctor specializations are associated with the specialization (entity) mapping table.
Drug	The drug entity is a supporting table for mapping all the possible drugs that could be taken by a user.	<ol style="list-style-type: none"> 1. A drug is mapped to one or more user_Drug (entity)
Ethnicity	Ethnicity is used to identify a user's origin and his/her makeup. Researchers will use that information to investigate further if there are any genetic links based on the user's race backgrounds. Similarly, doctors will	<ol style="list-style-type: none"> 1. Multiple ethnicity types can be associated with a user (entity). For instance, a user could be both white and Chinese.

	use this information to find out if particular treatments are effective based on the user's race.	
Experiment	The experiment entity is a supporting table for the main researcher entity. It is used as a mapping table to record all the possible experiments that can be addressed by one or more researchers.	1. One experiment is associated with one or more research_experiment (entity)
Family	The family table is a supporting table for mapping all the possible family members of a user. This family member is also a user of the health_System. For the purpose of this exercise, researchers and doctors will need to know whether or not a user's family also suffer from a similar disease for treatments and or research experiments.	1. A family is mapped to zero or more user_family (entity)
Gender	The gender table is used to identify a user's sex. A user should identify himself/herself as having a single sex even if it is undefined.	1. A gender is identified by one user and one user only
Health_System	In this database, the health_system entity is the central table. It is used to gather an overall history of a user's health. This table is accessed by users to enter their data, by doctors to monitor a user's health progress, and finally by researchers to mine data for particular health concerns.	1. One or more users can access the health_system. 2. One or more doctors can access the health_system. 3. One or more researchers can access the health_system.
Insurance	The insurance entity is a supporting table to the user table. A user may have one or no insurance card.	1. A user can have zero or one insurance card. 2. A doctor is associated with zero or one insurance company.
Location	The location entity is a supporting table to the user table. It is used as a mapping table to record possible birth and or a user's current living location.	1. A location can have one or more Location_User (entity)
Location_User	The location_user entity is a supporting table to the user table. It is used as a mapping table to record the birth location and as well all of the possible locations that a user may have lived at one point or another.	1. A user can have one or more location_User 2. A location can have one or more location_User

	Researchers will need to know all the possible locations that a user may have resided to track patterns of a potentially infectious disease.	
Readings	The readings entity is a supporting table to the user table. It is used as a mapping table to record possible health value. For instance, a user may have readings related to insulin intake due to diabetic complication.	1. A reading is associated with one more user_reading(entity)
Researcher	The researcher entity is one of the three main tables. It is used by the person who is a specialist in one or more disease control and is allowed to perform experiments by analyzing patterns found in the health system database.	1. One or more researchers can perform data analysis on the health_system (entity) 2. A researcher is tied to one or more researcher_specialization (entity) 3. A researcher is bound to one or more research_experiment (entity)
Researcher_Experiment	The researcher_Experiment entity is a supporting table to the researcher table. It is used as a mapping table so that we avoid using a many to many relations between researchers and possible experiments.	1. A researcher is associated with one or more researcher_experiment. 2. A researcher is tied to one or more researcher_specialization (entity) 3. An experiment is associated with one or more researcher_experiment.
Researcher_Specialization	The researcher_Specialization entity is a supporting table to the researcher table. It is used as a mapping table so that we avoid using a many to many relations between researchers and their specializations.	1. A researcher (entity) is associated with one or more researcher_specialization. 2. A specialization (entity) is tied for one or more researcher_specialization
Specialization	The specialization entity is a supporting table to both the researcher table and the doctor table. It is used to track all of the possible specializations in the field of the medicine.	1. A specialization (entity) is tied to one or more Doctor_Specialization (entity) 2. A specialization is bound to one or more researcher_specialization (entity)
Treatment	The treatment entity is a supporting table to the doctor table. It is used to track all of the possible treatments that could be administered by a physician.	1. A treatment is tied to one or more Doctor_Treatment (entity)

User	The user entity is one of the three main tables in this database. It can be used by anyone who is diagnosed with a chronic disease such as blood pressure. This table will have all the pertinent information to identify uniquely a user and thus doctors and researchers will be able to create a profile that includes treatments and or experiments where applicable.	<p>1. A user can have zero or more occupation (entity)2. A user is associated with one zip (entity) code 3. A user can have one more ethnicity (entity) background. 4. A user can have zero or more family member (User_Family) who is also using the health system. 5. A user can be tied to one and only one gender. 6. A user can have zero or one insurance (entity) card. 7. A user can have one birth location and one or more living location (Location_User)8. A user can be taken zero or more drugs (entity) to manage a disease. 9. A user can be treated by zero or more doctors (entity)10. The health_System (entity) can have records for one or more users. 11. A user can have one or more user_readings (entity) to keep track or a chronic disease level. 12. A user diagnosis is tied to one or more user_diagnosis (entity).</p>
User_Diagnosis	The user_diagnosis entity is used as a mapping table to avoid having to have a many to many relationships between users and the diagnosis that they could have. It is used as a support table for the user table.	<p>1. A user diagnosis is tied to one or more user_diagnosis (entity). 2. A diagnosis is linked to one or more user_diagnosis (entity)</p>
User_Family	The user_family entity is used as a mapping table to avoid having to have a many to many relationships between users and the number of family members they may have that are also in the health system. This entity is used as a support table for the user table.	<p>1. A family can be associated to one or more user_family 2. A user is linked to one or more user_family</p>
User_Readings	The user_readings entity is used as a mapping table to avoid having to have a many to many relationships between users and the number of readings that could have recorded. This entity is used as a support table for the user table.	<p>1. A reading can be associated with one or more user_readings 2. A user is linked to one or more user_readings</p>

Occupation	The occupation entity is used as a support table to the user table. It is used to identity the type of work that the user performs. This can be used by doctors and researchers to figure out the reason of a symptom.	1. A user can have zero or more occupations. However, for this exercise only the current occupation will be taken into consideration.
Zip	The zip entity is a supporting table to the user user table. It is used to pin point a location where the user resides	1. A user should have one and only one zip.
Zip_User	The Zip_user table is a mapping table between Zip and User table	1. A user may have multiple zip codes, but with a time column will be able to track the last zip code for the user.
Insurance_Doctor	The insurance_Doctor table is a mapping table between insurance and doctor table	1. A doctor may be associated with multiple insurance services. The mapping table helps to avoid a many to many relationships.
Ethnicity_User	The ethnicity_User table is a mapping table between user and eithnicity	2. A user may have multiple ethnicity as there is a possibility that he/she is mixed

Entities Design Decisions Attributes and Constraints:

Entity	Design Decisions	Attributes	Constraints
Diagnose	This entity contains two distinct attributes that are required to be used. It is designed to host all sort of mapping data related to a particular	Diagnoseld, Diagnose_category	Key: (Primary) Diagnoseld

	diagnosis such as diabetes.		
Doctor	<p>We design this entity with seven specific attributes. Only one attribute, insuranceid is not a required to be field as it is possible for a doctor not to be associated with an insurance company. This entity has both a doctorId and a SS# field. There is an argument that only one unique values is needed, but in this design, we decided to keep both because we want to use doctorid to tie in other entities and leave the SS# for tax and insurance purposes. A doctor should have a specialization to be part of this system. And to avoid a many to many relationships between this entity and specialization, we created a mapping table that has all possible medical specializations related to a doctor. Also, to avoid a many to many relationships between a doctor and the treatments that he/she is providing, we created a mapping table for recording all the possible treatments.</p>	DoctorId, Name, Specialization, InsuranceId, SS#, Email, Phone, LengthOfExperience	Key: (Primary) DoctorId (Foreign) SpecializationId, InsuranceId
Doctor_Treatment	<p>This entity contains three distinct attributes. None of these attributes are required to be used as a</p>	DoctorId, treatmentId, treatment_startdate	Key: (Foreign) DoctorId, TreatmentId

	doctor may or may not provide treatments. The reason for a doctor not providing treatment is because he/she possibly can just be checking the health system to find out if there are effective treatments for a particular disease.		
Doctor_Specialization	This entity contains two distinct attributes that are required to be used as a doctor has to have a specialization to access the health system and as well as to monitor a user.	DoctorId, specializationId	Key: (Foreign) DoctorId, SpecializationId
Drug	This entity contains two distinct attributes that are required to be used. It is designed to host all sort of mapping data related to a particular drug.	DrugId, Drug_name	Key: (Primary) DrugId
Ethnicity	This entity is designed with two attributes that are required to be filled out.	EthnicityId, Ethnicity_Type	Key: (Primary) EthnicityId
Experiment	The experiment table contains two attributes that are required to be used to map all of the possible mappings.	ExperimentId, Experiment_name	Key: (Primary) ExperimentId
Family	This entity contains two distinct attributes that are required to be used. It is designed to host all sort of mapping data that would create a relationship between multiple users of the health_system.	FamilyId, family_Relation	Key: (Primary) familyId
Gender	This entity contains two distinct attributes that are required to be used.	GenderId, Type	Key: (Primary) GenderId

Health_System	The health_system is designed like a fact table. It contains primarily foreign key values. All of the attributes except for drug-related attributes are required to be filled. It has no primary key as that will be implied with a subset of foreign keys.	Date_Time, DiagnosedId, UserId, DoctorId, ResearcherId, DrugId, ZipCodeId, Drug_Dosage, Drug_Frequency, ExperimentId, TreatmentId, LocationId	Key: (Foreign) UserId, DoctorId, ResearcherId
Insurance	The insurance table has five distinct attributes. Since a user may not have insurance, thus not all the fields are required to be filled out. For the purpose of this exercise, we will assume that a doctor will be working with one or zero insurance company.	InsuranceId, insurance_name, phone, insurance_category, other_noinsurance	key: (Primary) InsuranceId
Location	The location table has two distinct attributes. Since this is used as a mapping table, then each of the attributes is required to be used.	LocationId, Location_Name	Key: (Primary) LocationId
Location_User	The location_user table has four distinct attributes. Since this is used as a mapping table, then each of the attributes is required to be used. There will be one entry for the birth_location and one or more entries for the living_location.	UserId, Birth_Location, Living_Location, LocationId	Key: (Foreign) UserId, LocationId
Readings	The reading table has two distinct attributes. Since this is used as a mapping table, then each of the attributes is required to be used.	ReadingId, Reading_name	Key: (Primary) ReadingId

Researcher	The researcher table contains six distinct attributes. The researcherId and the SS# are both unique values. However, only the researcherId is used as the primary key to identify a particular researcher	ResearcherId, Name, SS#, Email, Phone	Key: (Primary) ResearcherId
Researcher_Experiment	The researcher_experiment table contains five distinct attributes. All except for one attribute, the experiment_enddate are required to be filled out.	ExperimentId, ResearcherId, Experiment_startDate, Experiment_EndDate	Key: (Foreign key) ExperimentId, ResearcherId
Researcher_Specialization	The researcher_specialization table contains two distinct attributes. All of those attributes are required to be filled out.	SpecializationId, ResearcherId	Key: (Foreign key) Specialization, ResearcherId
Specialization	The specialization table contains two distinct attributes. All of those attributes are required to be filled out.	SpecializationId, Specialization_Name	Key: (Primary) SpecializationId
Treatment	The treatment table contains two distinct attributes. All of those attributes are required to be filled out.	TreatmentId, Treatment_Name	Key: (Primary) TreatmentId

User	The user table contains multiple foreign keys that are tied to either mapping tables or tables that have a particular role to the user. Even though in the ERD it shows that a user can have zero or more occupation, but for the purpose of this exercise we will only keep track of the most recent occupation. Email, SS# and UserId could be used as unique entry to identify a particular user, but for joining to other tables that reference the User table, we decided to have UserId has the primary key	UserId, Name, EthnicityId, BirthDate, Address, DiagnoseId, ZipCodeId, SS#, Email, Phone, OccupationId, DoctorId, InsuranceId	Key:(Primary) UserId(Foreign key) EthnicityId, DiagnoseId, ZipCodeId, OccupationId, DoctorId, InsuranceId
User_Diagnosis	The user_diagnosis table contains three distinct attributes that are required to be filled. It maps data between the user table and the diagnose table.	UserId, DiagnoseId, DiagnoseDate	Key (Foreign Key) UserId, DiagnoseId
User_Family	The user_family table contains two distinct attributes that are required to be filled only if a user has family members that are also in the health system database.	UserId, familyId	Key: (Foreign Key): UserId, FamilyId
User_Readings	The user_readings table contains two distinct attributes that are required to be filled. Each user is required to be collecting health values that will be monitored by doctors and mined by researchers.	ReadingId, UserId	Key: (Foreign Key): UserId, ReadingId

Occupation	<p>The occupation table contains three distinct attributes. They are not</p> <p>Required to be filled out as a user may or may not have an occupation</p>	OccupationId, occupation_Name, occupation_Experience	Key: (Primary): OccupatonId
Zip	The zip table contains two distinct values that are required to be filled out.	ZipCodeId, City_name	Key: (Primary Key): ZipcodeId
Zip_User	The Zip_User table contains four distinct columns that are required	Zip_UserId, ZipCodeId, UserId, Time	Key (Primary key): Zip_UserId
Insurance_Doctor	The insurance_Doctor table contains three columns that are required	Doctor_InsuranceId, DoctorId, InsuranceId	Key (Primary key): Doctor_InsuranceId
Ethnicity_User	The Ethnicity_user table contains three columns	EthnicityId, UserId, Ethnicity_UserId	Key (Primary key): Ethnicity_UserId

