Laboratorio

Corso di Basi di Dati e Web

A.A. 2021/2022

Progetto “Ospedale”

Autore:

*Neilivan.lajara@studenti.unimi.it*

Data di consegna:

‘20/09/2022’

Il progetto è già stato consegnato in precedenza? {sì/no} NO

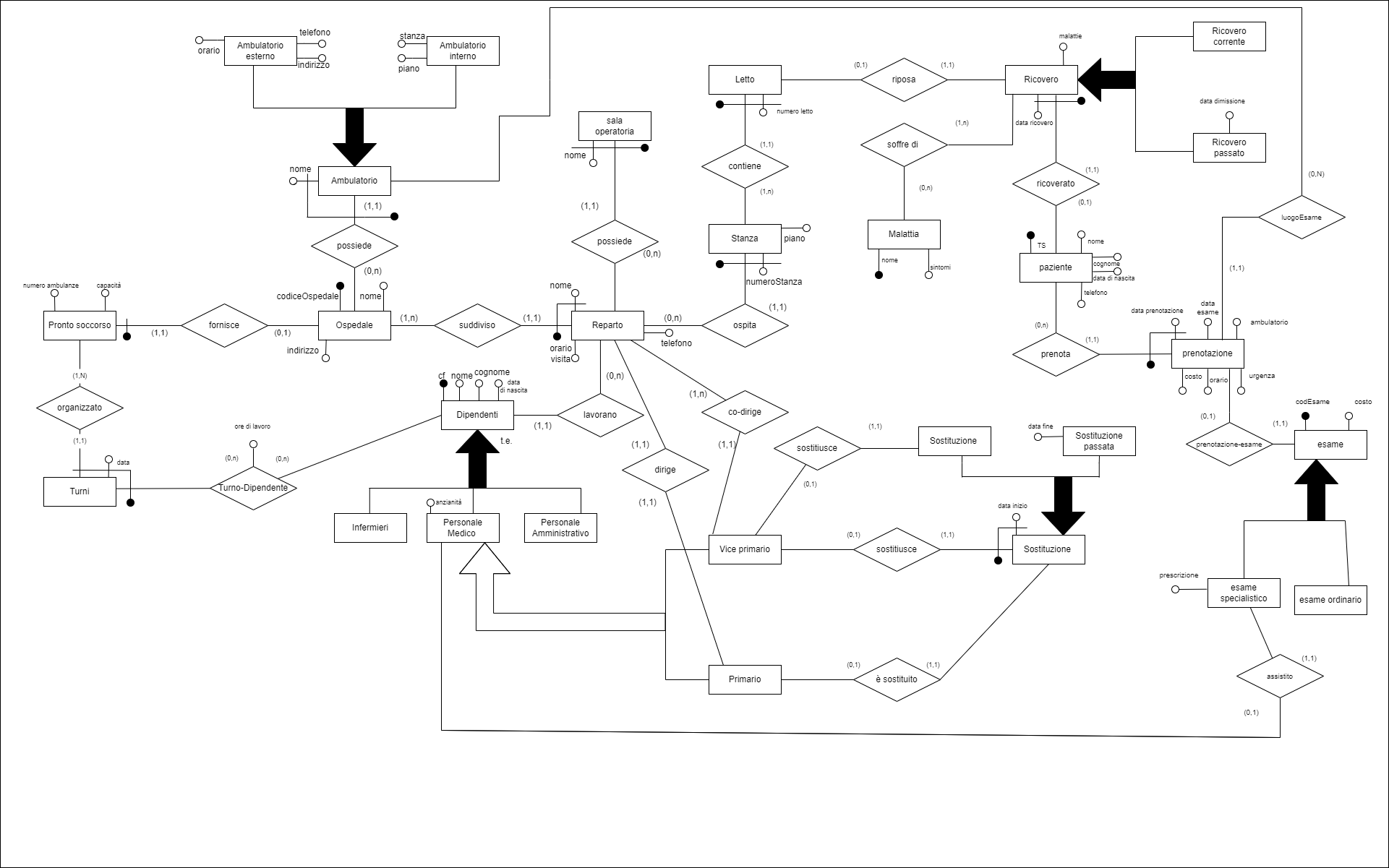
Se sì, indica tutte le date degli appelli in cui il progetto è stato consegnato:

Se sì, indica brevemente (tre righe al massimo) le modifiche apportate rispetto all’ultimo progetto consegnato:

1. Progettazione concettuale

*In questa sezione vanno riportate le conclusioni della progettazione concettuale. Non siamo interessati ai vari passaggi intermedi con i quali arrivate a costruire lo schema concettuale. Vogliamo lo schema concettuale e le considerazioni/vincoli finali.*

## 1.1 Schema ER

**

## 1.2 Vincoli di dominio

*In questa sezione vanno riportati e numerati tutti i vincoli che devono essere garantiti nella base di dati e che non si possono direttamente desumere dallo schema ER. La numerazione dei vincoli permette una più facile identificazione nelle fasi successive della progettazione ed implementazione.*

*------------------------------------------------------------------*

1. *Un dipendente può essere solo infermiere/personale medico/personale amministrativo.*
2. *Le ore di lavoro in Turno-Dipendente non devono essere più di 12 ore al giorno*
3. *Un primario e un vice primario che si sostituiscono devono lavorare nello stesso reparto*
4. *Un primario può dirigere solo un reparto*
5. *Un vice primario può co-dirigere un solo reparto*

*----------------------------------------------------------*

1.Un ospedale

1.1 è suddiviso da almeno un reparto

1.2 può avere o meno un pronto soccorso

1.3 può avere o meno uno o più ambulatori (solo 1 ambulatorio per tipo)

2.Un reparto

2.1 può avere o meno una sala operatoria

2.2 contiene un numero arbitrario di posti letto dentro una stanza

2.3 diretto da uno e un solo primario

2.4 può avere o meno dipendenti che ci lavorano

3.Ambulatorio

3.1 Un ambulatorio esterno non può essere interno e viceversa

* 1. può avere o meno essere luogo per degli esami
  2. Un ambulatorio deve essere interno oppure esterno

4.Dipendente

4.1 un infermiere non può essere un medico, ne del personale amministrativo e viceversa

4.2 un dipendente lavora solo per un ospedale e quindi reparto

* 1. un dipendente può lavorare o meno in un pronto soccorso

5,Personale medico

* 1. può essere o meno un primario, vice primario oppure non esserlo

6.Sostituzione

6.1 Un vice primario può sostituire solo un primario per volta, finché non finisce il periodo di sostituzione

* 1. Esistono sia dei primari che vice primari che non hanno mai fatto la sostituzione

7.Paziente

7.1 può o meno essere stato ricoverato

7.2 può essere stato ricoverato in periodi diversi

* 1. può prenotare o meno degli esami

8.Malattia

* 1. si mantengono informazioni anche di quelle malattie per cui non affetto da nessun paziente

9Esame

9.1 un esame specialistico non può essere ordinario o viceversa

* 1. un esame è svolto presso un solo ambulatorio

10.Prenotazione

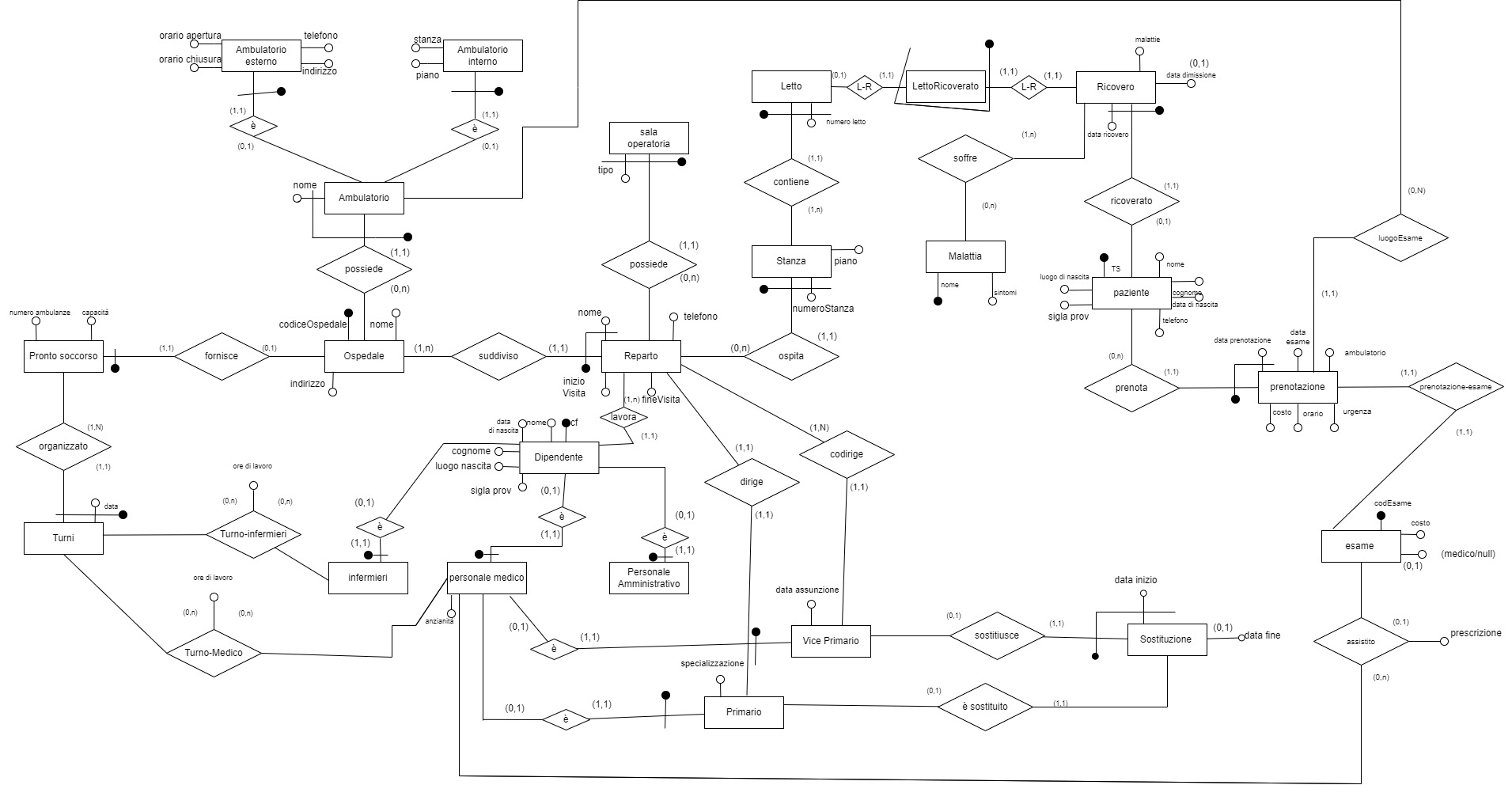
10.1 un paziente può prenotare un esame solo una volta al giorno

11.LettoRicoverato

11.1 Un letto può essere occupato solamente da un ricoverato al giorno

2. Progettazione logica

## 2.1 Schema ER ristrutturato

******

## 2.2 Vincoli di dominio

*La ristrutturazione dello schema può portare ad introdurre nuovi vincoli sullo schema. Indicare tutti i vincoli di dominio (compresi quelli già definiti nella progettazione concettuale) evidenziando quali sono stati aggiunti in fase di progettazione logica.*

1. *Ogni ricoverato deve almeno avere una malattia*
2. *Un dipendente è sempre un istanza di infermiere/personale medico/personale amministrativo, non esistono dipendenti che non ricoprano uno di questi ruoli né esistono dipendenti che ricoprano più di un ruolo*
3. ***Una stanza non può avere più di 6 letti***
4. ***In prenotazione, l’urgenza può essere verde/rosso/***
5. *Un reparto è diretto solo da un primario*
6. *Un reparto è co diretto da almeno un vice primario*
7. *Si può avere solo un turno al giorno nel pronto soccorso per un massimo di 12 ore di lavoro*
8. ***Un reparto può essere co-diretto da massimo 3 viceprimario***
9. ***Un sala operatoria può essere di tipo “urgente” o “programmata” quindi per un massimo di due sale operatoria per reparto***
10. ***Un dipendente lavora in un ospedale (unique)***
11. ***Un paziente è ricoverato con una sola malattia***

## 2.3 Modello relazionale

*Riportare il modello relazionale e gli eventuali vincoli da considerare. Eventuali ottimizzazioni devono essere prodotte a partire da questa versione, analogamente a quanto visto durante le lezioni di laboratorio.*

Ospedale(codOspedale, nome, indirizzo)

Reparto(codOspedale, nomeReparto, inizioVisita, fineVisita, telefono)

salaoperatoria(codospedale, nomereparto, tiposala)

Dipendente(CF, cognome, nome, dataNascita, luogoNascita)

Lavora(CF, codOspedale, nomeReparto)

Medico(CF, esperienza)

Primario(CF, specializzazione)

Viceprimario(CF, dataAssunzione)

Infermieri(CF)

PersAmministrativo(cf)

Sostituzione(cfvice,dataSostituzione,cfprimario, datainizio, datafine)

Ambulatorio(nomeAmbulatorio, codOspedale)

AmbulatorioInterno(nomeAmbulatorio, codOspedale, piano, stanza)

AmbulatorioEsterno(nomeAmbulatorio, codOspedale, telefono, orarioApertura, orarioChiusura)

ProntoSoccorso(codOspedale, numeroAmbulanze, capienza)

Turni(codOspedale, dataTurno)

turnoMedico(codOspedale, dataturno, cf, orelavoro)

turnoInfermiere(codOspedale, dataturno, cf, orelavoro)

Dirige(codOspedale, nomeReparto, cfPrim)

Codirige(codOspedale, nomeReparto, cfVice)

Stanza(codOspedale, nomeReparto, numeroStanza, piano)

letto(codOspedale, nomeReparto, numeroStanza, numeroLetto)

lettoRicoverato(codOspedale, nomeReparto, numeroStanza, numeroLetto, ts,dataRicovero)

Malattia(nomeMalattia, sintomi)

Paziente(TS, cognome, nome, tel, dataNascita, luogo di nascita)

Ricovero(ts,dataricovero, *malattia,* , datadimissione)

Prenotazione(ts, dataPrenotazione, dataEsame, costo, orario, urgenza, nomeAmbulatorio, nomeOspedale)

Esame(codEsame, ts, dataesame, medico)

Assiste(codEsame, medico, prescrizione)

## 2.4 Codice SQL

*create table Ospedale(*

*codOspedale decimal(5) ,*

*nome varchar(20),*

*indirizzo varchar(20),*

*primary key(codospedale)*

*);*

*create table Reparto(*

*codOspedale decimal(5),*

*nomeReparto varchar(15) not null,*

*inizioVisite TIME ,*

*fineVisite TIME ,*

*telefono varchar(10) not null,*

*foreign key (codOspedale) references ospedale(codospedale) on delete cascade on update cascade,*

*primary key(codOspedale, nomeReparto)*

*);*

*create table salaoperatoria(*

*codOspedale decimal(5),*

*nomeReparto varchar(15),*

*tipoSala varchar(20),*

*foreign key (codospedale, nomereparto) references reparto(codospedale, nomereparto) on delete cascade on update cascade,*

*primary key(codospedale, nomereparto, tiposala)*

*);*

*create table Dipendente(*

*cf text ,*

*nome varchar(15) not null,*

*cognome varchar(15) not null,*

*dataNascita date not null,*

*luogoNascita varchar(20) not null,*

*primary key(cf)*

*);*

*create table lavora(*

*cf text default 'rimosso',*

*nomeReparto varchar(15),*

*codOspedale decimal(5),*

*foreign key (nomeReparto, codOspedale) references Reparto(nomeReparto, codOspedale) on delete cascade on update cascade,*

*foreign key (cf) references dipendente(cf) on delete cascade on update cascade,*

*primary key(cf)*

*);*

*create table Medico(*

*cf text default 'rimosso',*

*esperienza decimal(2),*

*foreign key (cf) references Dipendente(cf) on delete cascade on update cascade,*

*primary key(cf)*

*);*

*create table Infermiere(*

*cf text default 'rimosso',*

*foreign key (cf) references dipendente(cf) on delete cascade on update cascade,*

*primary key(cf)*

*);*

*create table Primario(*

*cf text default 'rimosso',*

*specializzazione varchar(20) not null,*

*foreign key (cf) references medico(cf) on delete cascade on update cascade,*

*primary key(cf)*

*);*

*create table Viceprimario(*

*cf text default 'rimosso',*

*dataAssunzione date not null,*

*foreign key (cf) references medico(cf) on delete cascade on update cascade,*

*primary key(cf)*

*);*

*create table PersAmministrativo(*

*cf text default 'rimosso',*

*foreign key (cf) references dipendente(cf) on delete cascade on update cascade,*

*primary key(cf)*

*);*

*create table sostituzione(*

*cfvice text,*

*cfprim text,*

*dataInizio date not null,*

*dataFine date ,*

*primary key(cfvice, datainizio),*

*foreign key (cfvice) references viceprimario(cf) on delete cascade on update cascade,*

*foreign key (cfprim) references primario(cf) on delete cascade on update cascade*

*);*

*create table ambulatorio(*

*codOspedale decimal(5),*

*nomeAmbulatorio varchar(35)not null,*

*foreign key (codOspedale) references Ospedale(codOspedale)on delete cascade on update cascade,*

*primary key(nomeAmbulatorio, codOspedale)*

*);*

*create table AmbulatorioInterno(*

*codOspedale decimal(5) ,*

*nomeAmbulatorio varchar(35),*

*piano decimal (2),*

*stanza decimal(3),*

*foreign key (codOspedale, nomeAmbulatorio) references Ambulatorio( codOspedale, nomeAmbulatorio) on delete cascade on update cascade,*

*primary key(nomeAmbulatorio, codOspedale)*

*);*

*create table AmbulatorioEsterno(*

*codOspedale decimal(5) ,*

*nomeAmbulatorio varchar(35),*

*indirizzo varchar(35) not null,*

*telefono varchar(13) not null,*

*orarioApertura time,*

*orarioChiusura time,*

*foreign key (codOspedale, nomeAmbulatorio) references Ambulatorio( codOspedale, nomeAmbulatorio) on delete cascade on update cascade,*

*primary key(nomeAmbulatorio, codOspedale)*

*);*

*create table ProntoSoccorso(*

*codOspedale decimal(5),*

*capienza decimal(5),*

*numeroAmbulanze decimal(2) default 2,*

*foreign key (codOspedale) references Ospedale(codOspedale)on delete cascade on update cascade,*

*primary key (codOspedale)*

*);*

*create table turni(*

*codOspedale decimal(5),*

*dataTurno date not null,*

*foreign key (codOspedale) references prontosoccorso(codOspedale) on delete cascade on update cascade,*

*primary key(codOspedale, dataTurno)*

*);*

*-------------------------------*

*create table turnoinfermiere(*

*codOspedale decimal(5),*

*dataturno date,*

*cf text,*

*oreLavoro decimal(2),*

*foreign key (codOspedale, dataturno) references turni(codospedale, dataturno) on delete cascade on update cascade,*

*foreign key (cf) references infermiere(cf)on delete cascade on update cascade,*

*primary key(codospedale, dataturno, cf)*

*);*

*create table turnoMedico(*

*codOspedale decimal(5),*

*dataturno date,*

*cf text default 'rimosso',*

*oreLavoro decimal(2),*

*foreign key (codOspedale, dataturno) references turni(codospedale, dataturno) on delete cascade on update cascade,*

*foreign key (cf) references medico(cf)on delete cascade on update cascade,*

*primary key(codospedale, dataturno, cf)*

*);*

*create table dirige(*

*codOspedale decimal(5),*

*nomeReparto varchar(35),*

*cf text default 'rimosso',*

*foreign key (codOspedale, nomeReparto) references Reparto(codOspedale, nomeReparto) on delete cascade on update cascade,*

*foreign key (cf) references primario(cf)on delete cascade on update cascade,*

*primary key(codOspedale, nomeReparto, cf)*

*);*

*create table codirige(*

*codOspedale decimal(5),*

*nomeReparto varchar(15),*

*cf text default 'rimosso',*

*foreign key (codOspedale, nomeReparto) references Reparto(codOspedale, nomeReparto) on delete cascade on update cascade,*

*foreign key (cf) references viceprimario(cf)on delete cascade on update cascade,*

*primary key(codOspedale, nomeReparto,cf)*

*);*

*create table stanza(*

*codOspedale decimal(5),*

*nomeReparto varchar(15),*

*numeroStanza decimal (3),*

*piano decimal(2),*

*foreign key(codOspedale, nomeReparto) references reparto(codOspedale, nomeReparto) on delete cascade on update cascade,*

*primary key(codospedale, nomereparto, numerostanza)*

*);*

*create table Paziente(*

*ts varchar(20),*

*cognome varchar(15) not null,*

*nome varchar(15) not null,*

*tel varchar(15) not null,*

*datanascita date not null,*

*luogonascita varchar(20) not null,*

*primary key (ts)*

*);*

*create table malattia(*

*nomeMalattia varchar(30) primary key,*

*sintomi text*

*);*

*----------------------------------------------------------------------------------------*

*create table ricovero(*

*ts varchar(20) ,*

*dataricovero date not null,*

*malattia varchar(30),*

*datadimissione date,*

*foreign key (ts) references paziente(ts) on delete cascade on update cascade,*

*foreign key (malattia) references malattia(nomemalattia),*

*primary key(ts, dataricovero)*

*);*

*create table letto(*

*codOspedale decimal(5),*

*nomeReparto varchar(15),*

*numeroStanza decimal (3),*

*numeroLetto decimal(1),*

*foreign key (codospedale, nomereparto, numerostanza) references Stanza (codospedale, nomereparto, numerostanza) on delete cascade on update cascade,*

*primary key (codospedale, nomereparto, numerostanza,numeroletto)*

*);*

*create table lettoricoverato(*

*codOspedale decimal(5),*

*nomeReparto varchar(15),*

*numeroStanza decimal (3),*

*numeroLetto decimal(1),*

*ts varchar(20),*

*dataricovero date,*

*foreign key (codospedale, nomereparto, numerostanza) references Stanza(codospedale, nomereparto, numerostanza) on delete cascade on update cascade,*

*foreign key (ts, dataricovero) references ricovero(ts, dataricovero) on delete cascade on update cascade,*

*primary key (codospedale, nomereparto, numerostanza,numeroletto, dataricovero)*

*);*

*create table prenotazione(*

*ts text,*

*dataPrenotazione date not null,*

*dataEsame date not null,*

*costo decimal(10,2),*

*orarioesame time not null,*

*urgenza varchar(10),*

*codOspedale decimal(5),*

*nomeAmbulatorio varchar(35),*

*foreign key(codOspedale, nomeambulatorio) references ambulatorio(codospedale, nomeambulatorio) on delete cascade on update cascade,*

*CONSTRAINT chk\_Frequency CHECK (urgenza IN ('verde', 'gialla', 'rossa')),*

*foreign key (ts) references paziente(ts) on delete cascade on update cascade,*

*UNIQUE (ts, dataesame),*

*primary key(ts, dataprenotazione)*

*);*

*create table esame(*

*codEsame varchar(100),*

*ts varchar(20),*

*dataesame date,*

*medico text,*

*FOREIGN KEY (ts, dataesame) REFERENCES public.prenotazione(ts, dataesame) on delete cascade on update cascade,*

*UNIQUE (ts, dataesame),*

*FOREIGN KEY (medico) REFERENCES public.medico(cf) on delete cascade on update cascade,*

*primary key(codEsame)*

*);*

*create table assiste(*

*codEsame varchar(100),*

*medico text,*

*prescrizione varchar(255),*

*foreign key (codEsame) references esame(codesame) on delete cascade on update cascade,*

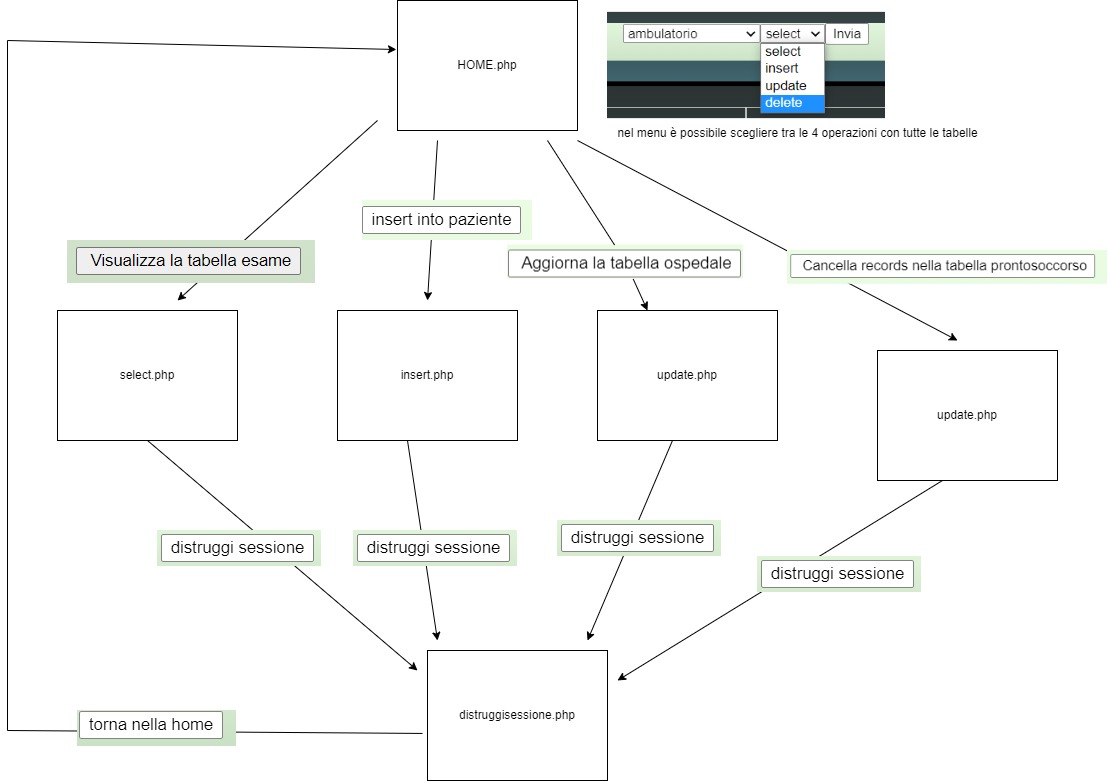
*foreign key (medico) references medico(cf) on delete cascade on update cascade,*

*primary key (codEsame, medico)*

*);*

3. Progettazione del sito

*Rappresentare uno schema che descriva il flusso di navigazione all’interno del sito. Ogni pagina deve essere rappresentata da un rettangolo che contenga il nome (univoco) della pagina e l’elenco delle funzionalità principali. Indicare come l’utente può spostarsi tra una pagina e l’altra.*

**

*Nome del database PostgreSQL: Ospedale*

*Nome della cartella sotto htdocs: \Ospedale*