

PROGETTO BASI DI DATI

Valerio Riccardo, matricola: 2075517

Scanferla Marco, matricola: 2075523

DB OSPEDALIERO

1 ABSTRACT

Il sistema di database per gli ospedali è un progetto necessario per affrontare le sfide poste dalla crescente complessità dei dati sanitari e dall'esigenza di sistemi di gestione ospedaliera efficaci. Questa relazione descrive la progettazione e l'implementazione di un database relazionale che mira a migliorare la gestione dei servizi ospedalieri consentendo un accesso efficiente e ben organizzato alle informazioni critiche. Il database è in grado di gestire vari aspetti dell'ambiente ospedaliero, come i dati dei pazienti, i referti cioè i risultati delle visite effettuate, il personale e i reparti. Inoltre, il database può gestire anche i dati relativi ai medicinali somministrati ai pazienti, garantendo una corretta dispensazione e tracciabilità dei farmaci. L'obiettivo è quello di offrire una piattaforma che faciliti la gestione delle informazioni cliniche e amministrative in modo sicuro e scalabile, favorendo al tempo stesso una migliore collaborazione tra i vari servizi ospedalieri.

2 ANALISI DEI REQUISITI

2.1 Descrizione

Si vuole creare una base di dati composta da diverse entità che rappresentano vari aspetti di un sistema ospedaliero. Ecco una descrizione di ciascuna:

- **Paziente:** Questa entità rappresenta un utente dell'ospedale con dettagli personali e caratteristiche fisico-sanitarie.
 - Include informazioni come il codice fiscale, nome, cognome, indirizzo, e-mail, telefono, data di nascita e dettagli fisico-sanitari come altezza, peso, sesso, gruppo sanguigno, Rh e numero di malattie. Potrà essere relazionato Cartelle Cliniche, malattie, referti e un medico di base.
- **Cartella Clinica:** Questa entità rappresenta il record medico di un paziente.
 - Include il codice della cartella, la data, la frequenza e le pressioni minima e massima. Potrà essere relazionata al paziente.
- **Personale:** Questa entità rappresenta il personale dell'ospedale, che può essere un volontario o un dottore.
 - Include informazioni come il codice fiscale, nome, cognome, telefono e indirizzo. Potrà essere relazionato ai reparti e se dottore anche ai referti.
- **Medico di base:** Questa entità rappresenta il medico di famiglia esterno all'ospedale.
 - Include informazioni come il codice fiscale, nome, cognome, telefono e indirizzo. Potrà essere relazionato con i suoi pazienti.
- **Reparto:** Questa entità rappresenta i diversi reparti dell'ospedale, che possono essere cardiologia, oncologia, ortopedia e dermatologia.

- Include il codice del reparto e il telefono. Potrà essere relazionato al personale che ci lavora.
- **Malattia:** Questa entità rappresenta le malattie.
 - Include il nome della malattia e se è contagiosa o meno. Potrà essere relazionata al referto e al paziente.
- **Referto:** Questa entità rappresenta un referto medico.
 - Include il numero del referto, le note, la data e l'ora. Potrà essere relazionato al paziente, al dottore, alla medicina e alla malattia.
- **Medicina:** Questa entità rappresenta un medicinale.
 - Include il codice della medicina, il produttore, la forma, il prezzo singolo e il nome. Potrà essere relazionata con il referto e con il lotto.
- **Lotto:** Questa entità rappresenta un lotto di un tipo di medicine.
 - Include la quantità, la data dell'ordine e il costo. Potrà essere relazionato alla medicina.

2.2 Glossario

Codice Fiscale (CF): Il codice fiscale è un codice alfanumerico di 16 caratteri utilizzato per identificare in modo univoco i cittadini italiani ai fini fiscali ed amministrativi. È geneticamente determinato ed ereditato alla nascita, con contributi da entrambi i genitori. Il codice fiscale viene generato in base a vari dati personali, tra cui il cognome, il nome, la data di nascita, il sesso e il luogo di nascita.

Gruppo Sanguigno (RH): Il gruppo sanguigno è una delle numerose caratteristiche biometriche e genetiche di un individuo e viene classificato tramite la presenza o l'assenza di antigeni sulla superficie dei globuli rossi. Esistono più di trenta sistemi diversi di classificazione, ma il più comune è il sistema AB0, che classifica il sangue in quattro gruppi: A, B, AB e 0. Ogni gruppo può essere Rh positivo o Rh negativo, a seconda della presenza o meno dell'antigene Rh D.

3. PROGETTAZIONE CONCETTUALE

3.1 Lista Entità

La Base di dati è formata dalle seguenti tabelle, tutti i campi sono da ritenersi non nulli a meno che non vi sia specificato diversamente di seguito.

-Paziente: Un utente dell'ospedale con dettagli personali e caratteristiche fisico-sanitarie.

- CF: varchar(16)
- Nome: varchar(50)

- Cognome: varchar(50)
- Indirizzo: varchar(100)
- e-mail: varchar(100)
- telefono: varchar(15)
- data di nascita: date
- **Caratteristiche fisico-sanitarie:** attributo composta da:
 - altezza: varchar(5)
 - peso: numeric(5,2)
 - sesso :varchar(10)
 - Gruppo sanguigno :varchar(3)
 - Rh: varchar(1)

-Cartella Clinica: Rappresenta il record medico di un paziente.

- Codice Cartella: varchar (10)
- data: date
- frequenza: numeric(3,0)
- pressione min :numeric(3,0)
- pressione max: numeric(3,0)

-Personale: Rappresenta il personale dell'ospedale.

- CF: varchar(16)
- Nome: varchar(50)
- Cognome: varchar(50)
- Telefono: varchar(15)
- indirizzo: varchar(100)
- È la generalizzazione di: **Volontari e Dottore**

-Medico di base: Rappresenta il Medico di famiglia esterno all'ospedale.

- CF: varchar(16)
- Nome: varchar(50)
- Cognome: varchar(50)
- Telefono: varchar(15)
- indirizzo: varchar(100)

-Reparto: Rappresenta i diversi reparti dell'ospedale.

- Codice reparto: varchar (5)
- Telefono :varchar(15)
- È la generalizzazione di: **Cardiologia, Oncologia, Ortopedia e Dermatologia.**

-**Malattia:** Rappresenta le malattie.

- Nome malattia: varchar(50)
- Contagiosa: varchar(2)

-**Referto:** Rappresenta un referto medico

- Numero Referto: numeric(10,0)
- Note: varchar(500)
- Data: date
- Ora: time

-**Medicina:** Rappresenta un medicinale

- Codice Medicina: varchar(10)
- Produttore: varchar(50)
- forma: varchar(50)
- Prezzo singolo: numeric(5,2)
- nome: varchar(50)

-**Lotto:** Rappresenta un lotto di un tipo di medicina

- data ordine (composta con medicina): date
- Quantità: numeric(5,0)
- costo: numeric(7,2)

3.2 Lista Relazioni

Lotto-Medicina Appartenenza:

- Un lotto contiene un solo tipo di medicina (1,1).
- Una medicina può appartenere a molti lotti (1, N).

Referto-Medicina Prescrizione:

- Una medicina può essere prescritta molte volte, ma anche nessuna (0, N).
- Un referto può prescrivere molte medicine, ma anche nessuna (0, N).

Referto- Malattia Risultato:

- IL risultato di un referto poi essere una malattia, ma anche nessuna (0, 1).
- Una malattia può essere il risultato di molteplici referti, ma anche di nessuno (0, N).

Paziente-Cartella Clinica Possiede:

- Un paziente può possedere molte cartelle cliniche, ma può anche non possederla (0, N).
- Una cartella clinica è posseduta da un paziente (1,1).

Personale–Reparto Assegnazione:

- Un membro del personale può essere assegnato a un solo reparto (1,1).
- Un reparto può essere assegnato a molto personale (1, N).

Paziente-Medico di base Supervisione:

- Un paziente è sotto la supervisione di un medico di base (1,1).
- Un medico di base supervisiona molti pazienti (1, N).

Dottore-Referto Produzione:

- Un referto è prodotto da un dottore durante una visita (1,1).
- Un dottore può produrre molti referti (1, N).

Dottore-Referto Ottenimento:

- Un referto è ottenuto da un paziente durante una visita (1,1).
- Un paziente può ottenere molti referti (1, N).

Paziente-Malattia Diagnosi:

- Un paziente può avere più malattie, ma anche nessuna (0, N).
- Una malattia può essere diagnosticata a più pazienti, ma anche a nessuno (0, N).

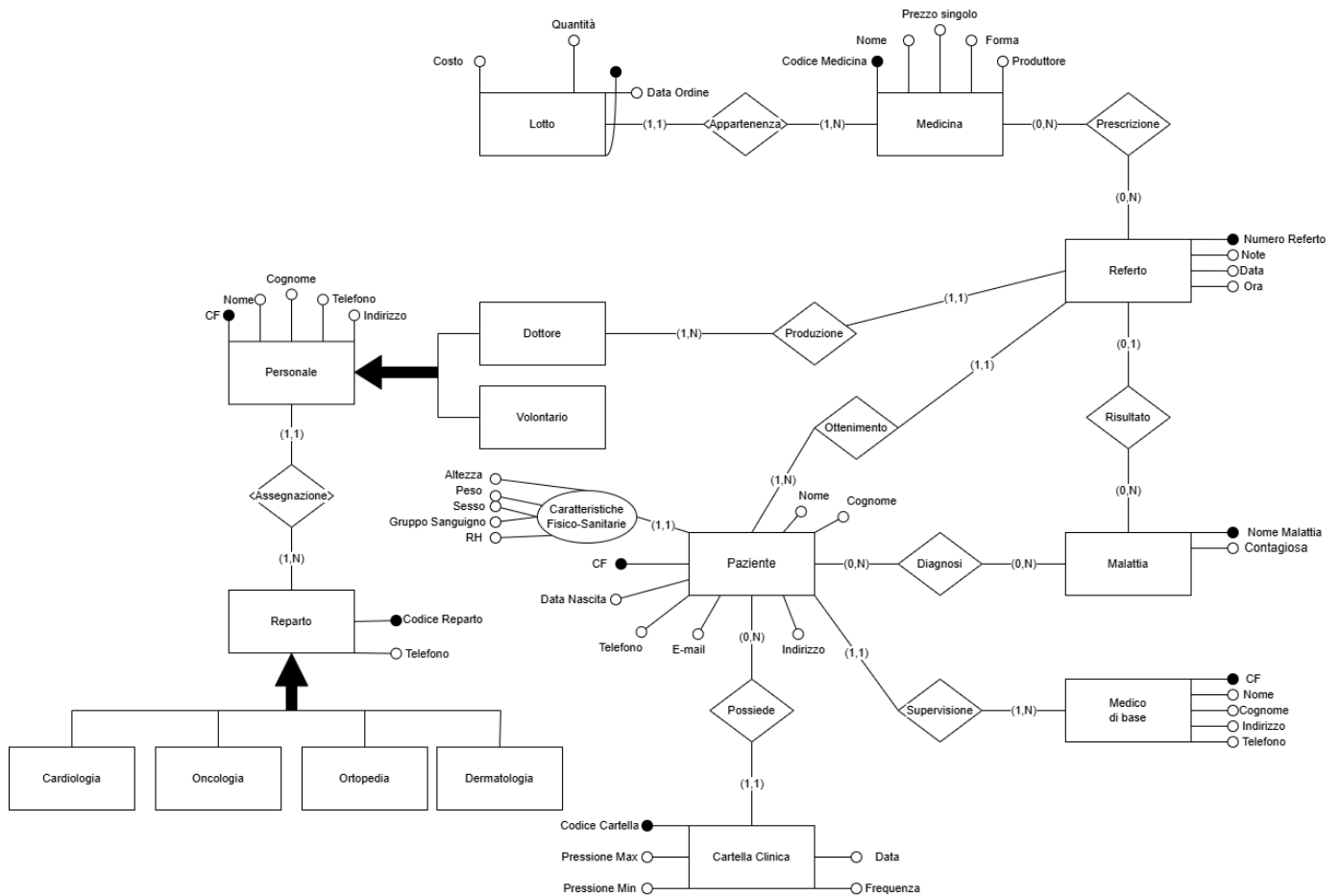
Paziente e l'attributo **Caratteristiche Fisico-Sanitarie**

- Un paziente ha come dati un elenco di Caratteristiche Fisico-Sanitarie (1,1).
- Delle Caratteristiche fisico-Sanitarie sono dati di un unico paziente (1,1).

3.3 Lista Generalizzazioni

- La tabella **Personale** è una generalizzazione totale ed esclusiva delle tabelle **Dottore** e **Volontario**. In quanto sono gli unici elementi del personale.
- La tabella **Reparto** è una generalizzazione totale ed esclusiva delle tabelle **Cardiologia**, **Oncologia**, **Ortopedia** e **Dermatologia**. In quanto questi sono gli unici reparti dell'ospedale.

3.4 Schema E-R



4 PROGETTAZIONE LOGICA

4.1 Analisi delle Ridondanze

Nello schema E-R è presente un attributo ridondante ed è **Costo** nell'entità **Lotto**, l'attributo può essere ricavato moltiplicando la **quantità** per il **prezzo singolo** in **Medicina**.

Concetto	Tipo	Volume
Medicina	Entità	50
Lotto	Entità	1000

Operazione 1:

La modifica di un elemento prezzo singolo della medicina necessiterà della correzione del costo di tutti i lotti relazionati a quella medicina.

Supponiamo che questa operazione avvenga 10 volte in un mese.

Con ridondanza:

Concetto	Costrutto	Accessi	Tipo
Medicina	Entità	1	scrittura
Lotto	Entità	20	Scrittura

La media di lotti per medicina è $1000/50=20$

Il numero di accessi sarà quindi:

$$((2*1) + (2*20)) * 10 = 420$$

Senza ridondanza:

Concetto	Costrutto	Accessi	Tipo
Medicina	Entità	1	scrittura

Il numero di accessi sarà quindi:

$$(2*1) * 10 = 20$$

Operazione2:

Il controllo del costo dei lotti relazionati ad una medicina

Supponiamo che questa operazione avvenga 30 volte in un mese.

Con Ridondanza:

Concetto	Costrutto	Accessi	Tipo
Lotto	Entità	1	Lettura

Il numero di accessi sarà quindi:

$$1 * 30 = 30$$

Senza Ridondanza:

Concetto	Costrutto	Accessi	Tipo
Medicina	Entità	1	Lettura
Lotto	Entità	20	Lettura

Il numero di accessi sarà quindi:

$$2 * 30 = 60$$

Conclusione:

Con Ridondanza = 420 + 30 = 450

Senza Ridondanza = 20 + 60 = 80

Questo dimostra che l'attributo ridondante **Costo** aumenta gli accessi al Database (450 > 80), per questo abbiamo deciso di **rimuoverlo**.

4.2 Eliminazione delle Generalizzazioni

Generalizzazione Personale:

Nel contesto di un sistema di gestione ospedaliera, è fondamentale distinguere tra i vari ruoli, come Dottori e Volontari, e le loro specifiche relazioni. Creare tabelle separate per ogni ruolo, "**Dottori**" e "**Volontari**", permette di gestire efficacemente le relazioni specifiche di ciascun ruolo. L'aggiunta di nuove relazioni, come "**Assegnazione Dottore**" e "**Assegnazione Volontario**", tra il personale e i reparti, consente di mantenere il database organizzato e facile da gestire, pur rispettando le specificità di ogni ruolo. Questo approccio ci permette di mantenere il nostro database organizzato e facile da gestire, pur rispettando le specificità di ogni ruolo.

Generalizzazione Reparto:

Un sistema di gestione ospedaliera ha diversi reparti, come **Cardiologia**, **Oncologia**, **Ortopedia** e **Dermatologia**. Tutti i reparti hanno le stesse proprietà.

Per organizzare meglio queste informazioni, abbiamo creato una tabella generale chiamata "**Reparto**". Questa tabella ha colonne per gli attributi che sono validi per tutti i reparti.

Inoltre, per differenziare i vari reparti, abbiamo aggiunto un attributo chiamato "**Tipo**" alla tabella "**Reparto**". Questo attributo conterrà il nome specifico del reparto, come Cardiologia, Oncologia, Ortopedia o Dermatologia.

In questo modo, possiamo usare una sola tabella per gestire le informazioni essenziali di tutti i reparti, e allo stesso tempo mantenere la possibilità di distinguere tra le varie specializzazioni.

4.3 Partizionamenti e Accorpamento

Partizionamento di Paziente:

Nella gestione sanitaria, l'entità "**Paziente**" si può suddividere in due sottocategorie differenti: "**Paziente**" e "**Caratteristiche fisico-sanitarie**". La categoria "paziente" comprende dati personali come il CF, nome, cognome, data di nascita, telefono, e-mail e indirizzo. Mentre, "caratteristiche fisico-sanitarie" includono dati specifici sulla salute del paziente, come altezza, peso, sesso, gruppo sanguigno e fattore Rh.

Tra Paziente e Caratteristiche fisico-sanitarie, si instaura una relazione "**dati paziente**". Questa relazione assicura che i dati personali del paziente siano sempre associati alle sue caratteristiche fisico-sanitarie. Questo consente una gestione più efficace e personalizzata delle cure sanitarie, garantendo allo stesso tempo la privacy e la sicurezza dei dati del paziente.

4.4 Normalizzazione della Relazione

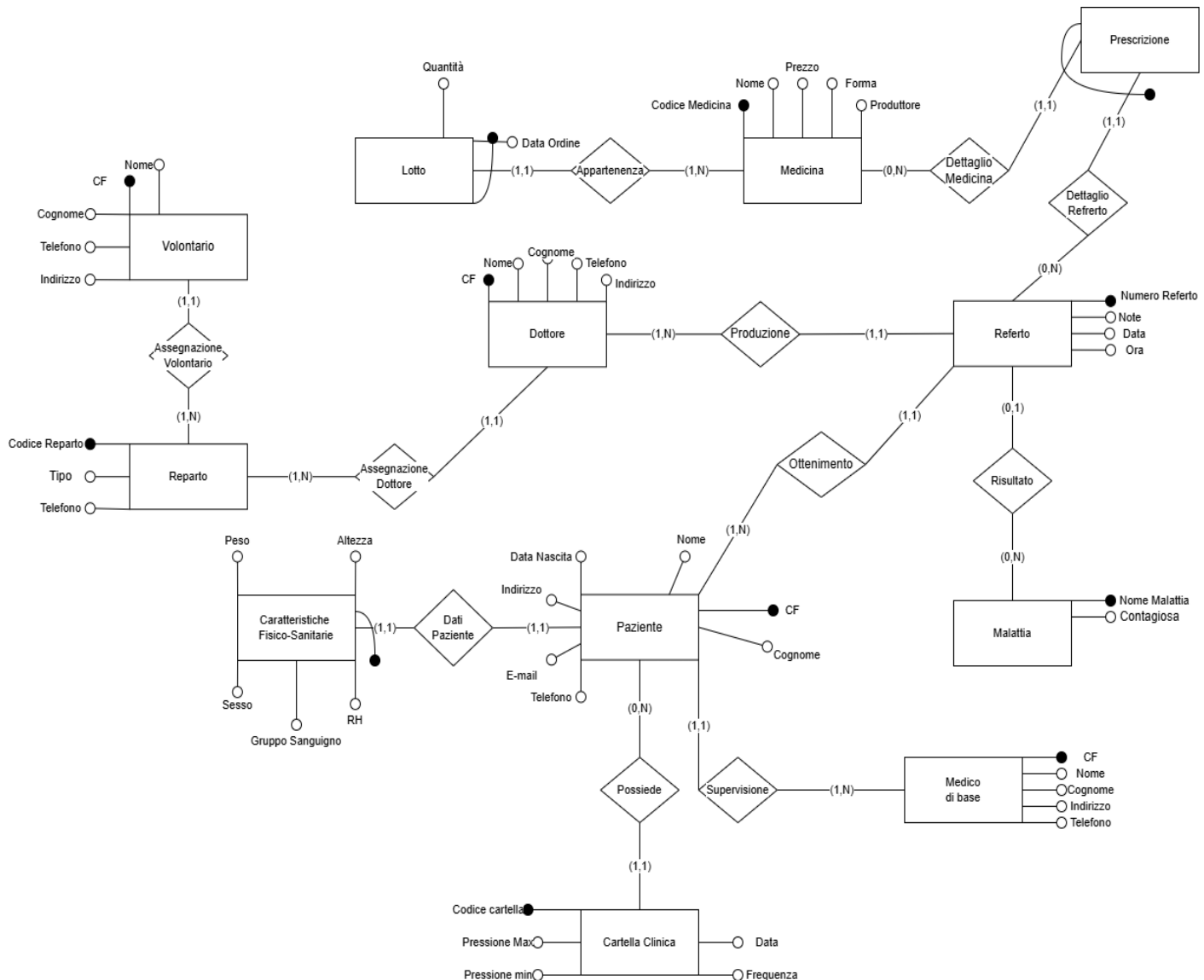
La relazione originale tra le entità **Referto** e **Medicina** era di tipo molti-a-molti. Per risolvere questa situazione, si è introdotta una nuova tabella denominata **Prescrizione**. Questa tabella rappresenta il legame tra un referto e una medicina, e ha come attributi le chiavi primarie di Referto e Medicina. In seguito, si sono create due nuove relazioni uno-a-molti: una chiamata **Dettaglio Referto** tra Referto e Prescrizione, e una chiamata **Dettaglio Medicina** tra Medicina e Prescrizione. Rimozione della relazione molti-a-molti **diagnosi** tra le entità **Paziente** e **Malattia**. Questa informazione è ottenibile attraverso l'entità **Referto**. In questo modo, sono state eliminate le relazioni molti-a-molti iniziali, ottimizzando così la struttura e la coerenza dei dati nello schema ER.

4.5 Scelta Identificatori Primari

- **Paziente** avrà come identificatore primario **CF**.
- **Caratteristiche Fisico-Sanitarie** avrà come identificatore primario **Paziente** quindi **CF**.
- **Dottore** avrà come identificatore primario **CF**.
- **Volontario** avrà come identificatore primario **CF**.

- **Lotto** avrà come identificatore primario **Medicina** quindi **Codice Medicina**.
- **Reparto** avrà come identificatore primario **Codice Reparto**.
- **Prescrizione** avrà come identificatori primari **Referto** e **Medicina**, quindi **Numero Referto** e **Codice Medicina**.

4.6 Diagramma E-R ristrutturato



4.7 Descrizione Schema Relazionale

- **Paziente** (CF, Nome, Cognome, Data Nascita, Indirizzo, E-mail, Telefono)
- **Caratteristiche Fisco-Sanitarie** (Paziente.CF, Altezza, Peso, Sesso, Gruppo Sanguigno, RH)
- **Cartella Clinica** (Codice Cartella, Data, Pressione Max, Pressione min, Frequenza)
- **Medico di Base** (CF, Nome, cognome, Indirizzo, Telefono)
- **Referto** (Numero Referto, Note, Data, Ora)
- **Malattia** (Nome Malattia, Contagiosa)
- **Medicina** (Codice Medicina, Nome, Prezzo, Forma, Produttore)
- **Lotto** (Medicina, Data Ordine, Quantità)
- **Dottore** (CF, Nome, cognome, Indirizzo, Telefono)

- **Reparto** (Codice Reparto, Tipo, Telefono)
- **Volontario** (CF, Nome, cognome, Indirizzo, Telefono)
- **Prescrizione** (Referto, Medicina)

4.8 Vincoli di Integrità Referenziale

- **MedicoDiBase.CF** < **Paziente.MedicoDiBase**
- **Paziente.CF** < **CaratteristicheFisicoSanitarie.Paziente**
- **Paziente.CF** < **CartellaClinica.Paziente**
- **Reparto.CodiceReparto** < **Dottore.Reparto**
- **Paziente.CF** < **Referto.Paziente**
- **Dottore.CF** < **Referto.Dottore**
- **Malattia.NomeMalattia** < **Referto.Malattia**
- **Medicina.CodiceMedicina** < **Lotto.Medicina**
- **Reparto.CodiceReparto** < **Volontario.Reparto**
- **Referto.NumeroReferto** < **Prescrizione.Referto**
- **Medicina.CodiceMedicina** < **Prescrizione.Medicina**

5 QUERY E INDICI

5.1 Lista Query

1. **Nome del paziente, data del record medico e note del referto medico:** Questa query restituisce il nome del paziente, la data del suo record medico e le note del suo referto medico.

```

SELECT
    "Paziente"."Nome",
    "Paziente"."Cognome",
    "CartellaClinica"."Data",
    "Referto"."Note"
FROM
    "Paziente"
JOIN
    "CartellaClinica" ON "Paziente"."CF" = "CartellaClinica"."Paziente"
JOIN
    "Referto" ON "Paziente"."CF" = "Referto"."Paziente";

```

Nome character varying (50)	Cognome character varying (50)	Data date	Note character varying (500)
Giorgio	Verdi	2023-06-01	Visita di routine, nessun problema rilevato.
Francesca	Neri	2023-06-02	Diagnosi di influenza, prescritto riposo e fluidi.
Luca	Bianco	2023-06-03	Controllo pressione, valori leggermente elevati.
Clara	Rosso	2023-06-04	Visita di controllo per artrite, stabile.
Marco	Giallo	2023-06-05	Segnalazione di mal di testa frequenti, esami ulteriori richie...
Sara	Russo	2023-06-10	Visita di routine, nessun problema rilevato.
Davide	Romano	2023-06-11	Diagnosi di influenza, prescritto riposo e fluidi.
Laura	Colombo	2023-06-12	Controllo pressione, valori leggermente elevati.

2. **Numero di pazienti per medico di base (parametrica):** Questa query restituisce il numero di pazienti per un medico di base specifico, in questo caso 'Maria Rossi'. Puoi sostituire 'Maria' e 'Rossi' con un nome e un cognome specifici quando esegui la query. Maria Rossi deve avere almeno un paziente per far parte del database; quindi, restituirà il numero di pazienti di Maria Rossi.

```

SELECT
    "MedicoDiBase"."Nome",
    "MedicoDiBase"."Cognome",
    COUNT("Paziente"."CF") AS "Numero di Pazienti"
FROM
    "MedicoDiBase"
JOIN
    "Paziente" ON "MedicoDiBase"."CF" = "Paziente"."MedicoDiBase"
WHERE
    "MedicoDiBase"."Nome" = 'Maria' AND
    "MedicoDiBase"."Cognome" = 'Rossi'
GROUP BY
    "MedicoDiBase"."Nome",
    "MedicoDiBase"."Cognome"
HAVING
    COUNT("Paziente"."CF") > 0;

```

Nome character varying (50)	Cognome character varying (50)	Numero di Pazienti bigint
Maria	Rossi	1

3. **Media dell'età dei pazienti per sesso (parametrica):** Questa query calcola l'età media dei pazienti in base al sesso. Il sesso è specificato come 'M' (maschio) nella clausola WHERE. Puoi sostituire 'M' con un altro valore per ottenere l'età media dei pazienti di un sesso diverso.

```
SELECT    "CaratteristicheFisicoSanitarie"."Sesso",
          AVG(EXTRACT(YEAR FROM AGE("Paziente"."DataDiNascita"))) AS "Età Media"
FROM      "Paziente"
JOIN      "CaratteristicheFisicoSanitarie" ON "Paziente"."CF" = "CaratteristicheFisicoSanitarie"."Paziente"
WHERE     "CaratteristicheFisicoSanitarie"."Sesso" = 'M'
GROUP BY  "CaratteristicheFisicoSanitarie"."Sesso";
```

Sesso	Età Media
character varying (10) 🔒	numeric 🔒
M	43.6666666666666667

4. **Numero di referti per malattia:** Questa query restituisce il numero di referti per ciascuna malattia.

```
SELECT    "Malattia"."NomeMalattia",
          COUNT("Referto"."NumeroReferto") AS "Numero di Referti"
FROM      "Malattia"
LEFT JOIN  "Referto" ON "Malattia"."NomeMalattia" = "Referto"."Malattia"
GROUP BY  "Malattia"."NomeMalattia";
```

NomeMalattia	Numero di Referti
[PK] character varying (50) ✎ 🔒	bigint 🔒
HIV	0
Diabete	0
Influenza	2
Ipertensione	0
Artrite	1
Epatite	0
Cancro	0

5. **Numero di pazienti per medico di base con almeno un paziente:** Questa query restituisce il numero di pazienti per ciascun medico di base, ma solo per i medici di base che hanno almeno N pazienti (N può essere modificato in base alle esigenze). In questo

caso $N = 0$, quindi restituirà il nome e cognome dei Medici di base e la quantità di pazienti per ognuno.

```
SELECT
    "MedicoDiBase"."Nome",
    "MedicoDiBase"."Cognome",
    COUNT("Paziente"."CF") AS "Numero di Pazienti"
FROM
    "MedicoDiBase"
JOIN
    "Paziente" ON "MedicoDiBase"."CF" = "Paziente"."MedicoDiBase"
GROUP BY
    "MedicoDiBase"."Nome",
    "MedicoDiBase"."Cognome"
HAVING
    COUNT("Paziente"."CF") > 0;
```

Nome character varying (50) 🔒	Cognome character varying (50) 🔒	Numero di Pazienti bigint 🔒
Maria	Rossi	1
Simone	Grigio	1
Giulia	Ferrari	1
Riccardo	Blu	1
Luigi	Bianchi	1
Alessandro	Bianchi	1
Laura	Bianco	1
Sonia	Gialli	1
Chiara	Rossi	1
Andrea	Esposito	1
Paolo	Marrone	1
Elisa	Verde	1

- Medici che hanno fatto più di N referti:** Questa query restituisce i medici che hanno almeno più di N referti (N può essere modificato in base alle esigenze). In questo caso $N=0$, quindi restituirà il nome e cognome dei dottori e il numero di referti.

```

SELECT
    "Dottore"."Nome",
    "Dottore"."Cognome",
    COUNT("Referto"."NumeroReferto") AS "Numero di Referti"
FROM
    "Dottore"
JOIN
    "Referto" ON "Dottore"."CF" = "Referto"."Dottore"
GROUP BY
    "Dottore"."Nome",
    "Dottore"."Cognome"
HAVING
    COUNT("Referto"."NumeroReferto") > 0;

```

Nome character varying (50)	Cognome character varying (50)	Numero di Referti bigint
Alberto	Neri	1
Giovanni	Ferro	1
Luca	Legno	1
Elena	Pietra	1
Francesca	Bianchi	1
Sara	Acqua	1
Anna	Rossi	1
Marco	Martini	1

5.2 Indici

Indice sulla colonna “CF” della tabella “Paziente”: Molte delle tue query coinvolgono operazioni JOIN sulla colonna “CF” della tabella “Paziente”. Un indice su questa colonna può velocizzare queste operazioni JOIN. Inoltre, la colonna “CF” è usata come chiave primaria; quindi, un indice su di essa può migliorare le prestazioni delle operazioni di ricerca e ordinamento.

```
CREATE INDEX "idx_paziente_cf" ON "Paziente"("CF");
```

Indice sulla colonna “MedicoDiBase” della tabella “Paziente”: Questa colonna è spesso utilizzata nelle clausole WHERE e JOIN delle tue query. Un indice su questa colonna può migliorare le prestazioni di queste operazioni.

```
CREATE INDEX "idx_paziente_medicodibase" ON "Paziente"("MedicoDiBase");
```

6 CODICE IN C PER LA CONSULTAZIONE DEL DATABASE

Il codice in C utilizzato per consultare il database si trova in un file chiamato main.c.

Questo file si trova nella directory del progetto, dove sono presenti anche le librerie necessarie per la sua esecuzione.

Le librerie richieste dal file osp.c sono specificate nella directory delle dipendenze.

Per compilare il codice e ottenere l'eseguibile, `e necessario utilizzare il seguente comando GCC:

```
gcc -o programma_db osp.c -Idependencies/include -Ldependencies/lib -lpq
```

Questo comando compila il file sorgente osp.c e linka le librerie necessarie dalla cartella dependencies, creando un eseguibile chiamato programma db. L'opzione -lpq specifica la libreria libpq per la connessione a PostgreSQL, che è essenziale per interagire con il database.