



Basi di Dati, Modulo 2

Sapienza Università di Roma

Facoltà di Ing. dell'Informazione, Informatica e Statistica

Laurea in Informatica

Prof. Toni Mancini

<http://tmancini.di.uniroma1.it>

Progetto 20080110 (P.20080110)

QuickHospital

Versione 2021-03-22

2

Specifiche dei Requisiti

Il sistema *QuickHospital* deve permettere di memorizzare e gestire informazioni circa i **pazienti** e i **medici** dell'ospedale nel quale viene installato. In particolare, dei **pazienti** interessano alcune informazioni anagrafiche (**nome, cognome e data di nascita**) ed i loro **recapiti, distinti in recapiti telefonici, recapito email e postale** (questi ultimi **unici**).

Per quanto riguarda i **medici** dell'ospedale invece, interessa mantenere informazioni sul loro **nome, cognome e data di nascita**, ed i **pazienti che hanno in cura**.

Un paziente può essere **ricoverato**, in una certa **data**, solo se una **precedente verifica della disponibilità dei posti letto** presenti nell'ospedale ha dato esito positivo.

Una volta effettuato il ricovero, il paziente ha assegnato un **posto letto** nell'ambito di una **stanza**; una stanza può contenere **da un minimo di 1 ad un massimo di 8 posti letto**. Le stanze hanno un **piano** ed un **settore** (interi positivi).

Il sistema deve inoltre permettere la memorizzazione dello **storico** di tutti i pazienti che sono stati **ricoverati e poi dimessi** nel tempo, con le informazioni relative ai posti letto occupati durante i diversi ricoveri.¹

Sono funzionalità specifiche del sistema la **registrazione** del ricovero di un paziente e della sua **dimissione** ad opera del **personale di accettazione**. Inoltre il sistema deve assistere i **medici** ottimizzando il loro **percorso di visite**.

In particolare, il sistema deve permettere di calcolare, su richiesta di un medico, il suo **itinerario delle visite**, ovvero un **insieme ordinato delle stanze cui accedere** (che sono tutte e sole le stanze che ospitano i pazienti che ha in cura).

L'ordinamento è dato in primo luogo dal **piano delle stanze** dei pazienti da visitare, ed in secondo luogo dal **settore di appartenenza di tali stanze** (entrambi in **ordine crescente**). I settori sono infatti numerati secondo un criterio di vicinanza topologica. Pertanto se un dato medico deve visitare le stanze $\{(7,4), (7,1), (1,3), (1,1), (3,4)\}$ dove la prima componente di ognuna è il piano e la seconda il settore, l'itinerario di visita proposto deve essere $[(1,1), (1,3), (3,4), (7,1), (7,7)]$.

¹Si assume per semplicità che durante il periodo di un ricovero il paziente non possa cambiare letto.



Oltre ai pazienti dell'ospedale, il sistema gestisce anche **prestazioni mediche** fatte da medici dell'ospedale a pazienti esterni. L'anagrafica di tali pazienti è registrata nel sistema (ad opera del **personale addetto alle prenotazioni**), con l'informazione aggiuntiva della **particolare prestazione medica richiesta** al personale ospedaliero (oltre che la **data richiesta**). Le prestazioni sono caratterizzate da una **specializzazione** richiesta (ad., ortopedia, dermatologia, ecc.) e una **descrizione più estesa**.

Di ogni **medico** il sistema deve conoscere la sua **specializzazione primaria** e le sue **specializzazioni secondarie**.

Data una prestazione richiesta da un paziente esterno (per una specializzazione s), il sistema deve restituire **l'insieme dei medici maggiormente idonei a soddisfarla**. Il criterio di idoneità è il seguente: **se esistono medici con specializzazione primaria pari ad s** , il risultato è l'insieme di tali medici. Altrimenti, il risultato è l'insieme dei medici che hanno **s tra le loro specializzazioni secondarie**.

Il sistema *QuickHospital* è accessibile ai **medici**, al **personale amministrativo** e a quelli dell'**ufficio prenotazioni**.



Sapienza Università di Roma
Facoltà di Ing. dell'Informazione, Informatica e Statistica, Laurea in Informatica
Insegnamento di **Basi di Dati, Modulo 2**
Prof. Toni Mancini
Dipartimento di Informatica
<http://tmancini.di.uniroma1.it>

Esame BD2.Esame.Risposte – Modulo risposte prova scritta

Dati dello studente e dell'esame

Cognome e nome: BIANCO SIMONE Matricola: 11

Data: 14/08/23

Corso di laurea e canale di appartenenza:

- Laurea in Informatica, canale 1 (A-L, Prof. M. Mancini)
- Laurea in Informatica, canale 2 (M-Z, Prof.ssa M. De Marsico)
- Laurea in Informatica in Modalità Teledidattica Unitelma Sapienza

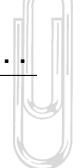
Firma di un membro della Commissione per
avvenuta identificazione:
.....

Rinuncia alla prova

- Desidero rinunciare a questa prova d'esame. Firma:



Questo modulo è ottimizzato per la stampa fronte-retro



Istruzioni e regole d'esame

Prima dell'esame

- Stampare questo modulo, preferibilmente fronte-retro, e rilegarlo con un fermaglio rimovibile, come quello disegnato in alto
- Compilare il frontespizio con i propri dati, come richiesto
- Scrivere la propria matricola nello spazio apposito nella parte alta di tutte le pagine

Durante l'esame

- La prova è dimensionata per essere svolta in circa 3 ore. Tuttavia, data la sua natura fortemente progettuale, la Commissione offre agli studenti la più ampia disponibilità di tempo, al fine ovviare ad eventuali (e limitati) errori di analisi/progettazione rilevati più a valle del ciclo di vita.

Il tempo massimo per la consegna è quindi rilassato a 5 ore (il massimo tempo compatibile con le disponibilità di aule).

- Scrivere le risposte negli spazi predisposti sotto le relative domande. Le ultime pagine sono vuote e possono essere usate come minute oppure, se puntate opportunamente, per contenere risposte in caso gli spazi appositi dovessero risultare insufficienti.
- Non è possibile usare alcun tipo di materiale didattico.
- In caso di necessità di ulteriori fogli (in proprio possesso), chiedere preventivamente alla Commissione una nuova procedura di controllo.
- La Commissione può rispondere solo a brevi domande inerenti al testo dei quesiti.
- Tra la seconda e la quarta ora d'esame, gli studenti possono effettuare **brevi pause** (uno studente alla volta) seguendo la seguente procedura:
 1. Alla lavagna è riportata una coda denominata 'Coda prenotazioni pause'. Sia n (un intero) l'elemento in fondo alla coda (si assuma $n = 0$ in caso di coda vuota).
 2. Recarsi alla lavagna ed aggiungere l'intero $n + 1$ come proprio contrassegno in fondo alla coda, seguito da una stringa a propria scelta (ad es., le proprie iniziali).
 3. Se il proprio contrassegno non è l'elemento affiorante della coda, tornare al lavoro in attesa che lo diventi.
 4. Consegnare tutti i fogli di lavoro e il testo d'esame alla Commissione ed uscire.
 5. Al rientro, cancellare il proprio contrassegno dalla coda di modo da permettere al successivo studente prenotato di uscire, e riprendere i fogli prima consegnati.

Al momento della consegna

- Ordinare tutti i fogli che si vuole far valutare e rilegarli con un fermaglio rimovibile. Non includere fogli che la Commissione non deve valutare (ad es., requisiti, minute), ma includere ovviamente il frontespizio.
- Consegnare i fogli ordinati **nelle mani** di un membro della Commissione. **Non lasciare l'aula senza la conferma, da parte della Commissione, del buon esito delle operazioni di consegna.**

In caso di rinuncia

- È possibile rinunciare alla consegna a partire dalla seconda ora d'esame. In caso di rinuncia, consegnare nelle mani della Commissione solo il frontespizio, dopo aver compilato e firmato la sezione dedicata.

Sommario delle domande

Si richiede di progettare l'applicazione descritta dalla specifica dei requisiti effettuando le fasi di Analisi concettuale dei requisiti e di Progettazione logica della base dati e delle funzionalità, utilizzando la metodologia vista nel corso.

In particolare (vengono indicati i tempi suggeriti per i diversi passi chiave):

Parte 1: Analisi concettuale dei requisiti Effettuare la fase di Analisi concettuale dei requisiti producendo lo schema concettuale per l'applicazione, che includa:

- Analisi dei dati (45 minuti; 75 minuti al massimo):
 - un diagramma ER concettuale (*)
 - il relativo dizionario dei dati
 - le specifiche dei domini concettuali non di tipo base
 - eventuali vincoli esterni, espressi utilizzando il linguaggio della logica del primo ordine (*)
- Analisi delle funzionalità:
 - un diagramma UML degli use-case (5 minuti; 10 minuti al massimo)
 - la segnatura di tutte le operazioni di use-case (10 minuti)
 - la specifica delle operazioni di use-case necessarie a modellare i requisiti contrassegnati dalla barra laterale (come quella qui a sinistra) in termini di precondizioni e postcondizioni, utilizzando il linguaggio della logica del primo ordine (*) (30 minuti; 60 minuti al massimo)

Parte 2: Progettazione della base dati e delle funzionalità Effettuare la progettazione della base dati e delle funzionalità a partire dallo schema concettuale prodotto nella Parte 1, ed in particolare eseguire i seguenti passi:

- Progettazione della base dati relazionale con vincoli:
 - Ristrutturazione del diagramma ER concettuale e dei vincoli esterni (20 minuti; 30 minuti al massimo):
 - * scelta del DBMS da utilizzare
 - * progettazione della corrispondenza tra i domini concettuali ed opportuni domini SQL (domini base o utente, oppure realizzati mediante relazioni aggiuntive) supportati dal DBMS scelto
 - * ristrutturazione del diagramma ER concettuale e dei vincoli esterni.
 - Produzione dello schema relazionale della base dati e dei relativi vincoli (*) (30 minuti; 60 minuti al massimo)
- Progettazione delle funzionalità (30 minuti; 45 minuti al massimo):
 - definizione della specifica realizzativa delle operazioni di use-case necessarie a modellare i requisiti contrassegnati dalla barra laterale, in modo conforme alla loro specifica concettuale prodotta nella fase di Analisi, in termini di algoritmi in pseudo-codice e comandi SQL. (*)

(*) Una risposta soddisfacente a questa domanda è condizione *necessaria* (ma non sufficiente) per superare la prova.

Le pagine seguenti contengono le domande specifiche a cui è richiesto rispondere, ulteriori delucidazioni per ogni singolo punto, e spazi per le risposte.

Le pagine da 33 in poi possono essere utilizzate per scrivere minute che non verranno valutate.



Questa pagina è stata intenzionalmente lasciata vuota

1 Analisi concettuale

Domanda 1 (10 minuti) Raffinare la specifica dei requisiti eliminando inconsistenze, omissioni e ridondanze e producendo un elenco numerato di requisiti il meno ambiguo possibile. (La risposta a questa domanda non sarà valutata, ma si consiglia di svolgere accuratamente questo passo, in quanto può facilitare di molto le attività di progetto.)

Risposta

PAZIENTE

- NOME, COGNOME, NASCITA
- RECAPITI TEL, EMAIL,
POSTALE (UNICO)

MEDICO

- NOME, COGNOME,
- NASCITA
- PAZIENTI
- SPECIALIZZAZIONI
(UNA PRIMARIA)

RICOVERO

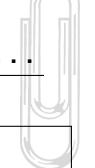
- SALO SE POSTI
DISPONIBILI
- INIZIO, FINE

STANZA

- [1, 8] POSTI LETTO
- PIANO, SETTORE

PRESTAZIONE MEDICA

- DATA
- MEDICO
- SPECIALIZZAZIONE
- DESCRIZIONE



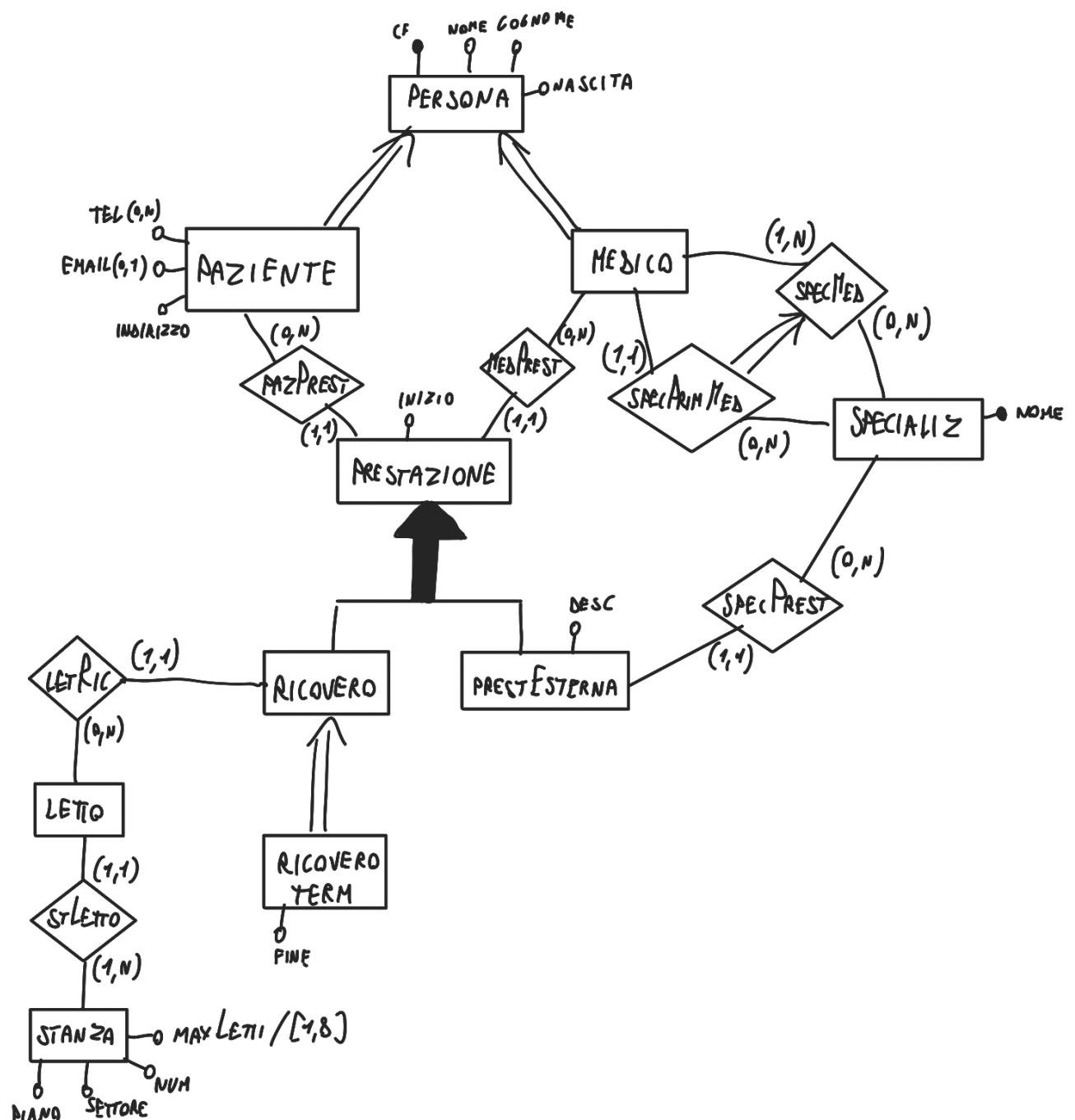
Risposta alla Domanda 1 (segue)

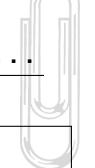
Domanda 2 (45 minuti; 75 minuti al massimo) Proseguire la fase di Analisi Concettuale dei requisiti, producendo un diagramma ER concettuale per l'applicazione, il dizionario dei dati ed eventuali vincoli esterni.

Una risposta soddisfacente a questa domanda è condizione *necessaria* (ma non sufficiente) per superare la prova.

Diagramma ER

Produrre un diagramma ER concettuale per l'applicazione in termini di entità, relationship, attributi, relazioni is-a, generalizzazioni (disgiunte) complete e non.





Risposta alla Domanda 2 (segue)

Dizionario dei dati Per ogni entità e relationship del diagramma ER **con** attributi o vincoli:

- Definire il dominio e la molteplicità degli attributi (se diversa da (1,1))
- Definire eventuali vincoli esterni in logica del primo ordine estesa con teoria degli insiemi e semantica di mondo reale, usando il seguente alfabeto:
 - Un simbolo di predicato $E/1$ per ogni entità E .
Semantica di $E(x)$: x è una istanza di E .
 - Un simbolo di predicato $D/1$ per ogni dominio D .
Semantica di $D(x)$: x è un valore di D .
 - Un simbolo di predicato r/n ($n > 0$) per ogni relationship n -aria r .
Semantica di $r(x_1, \dots, x_n)$: x_1, \dots, x_n è una istanza di r .
 - Un simbolo di predicato $a/2$ per ogni attributo a di entità
Semantica di $a(x, v)$: uno dei valori dell'attributo a dell'istanza x è v .
 - Un simbolo di predicato $a/(n+1)$ per ogni attributo a di relationship n -aria.
Semantica di $a(x_1, \dots, x_n, v)$: uno dei valori dell'attr. a dell'istanza (x_1, \dots, x_n) della relat. è v .
 - Opportuni simboli di predicato (soggetti a *semantica di mondo reale*) per gestire confronti tra valori di domini numerici o comunque ordinati (tra cui $</2$, $\leq/2$, $>/2$, $\geq/2$).
 - Il predicato di uguaglianza $=/2$ (la cui interpretazione è la relazione che lega ogni elemento del dominio di interpretazione solo con se stesso).
 - Opportuni simboli di costante (soggetti a *semantica di mondo reale*), tra cui *adesso*, interpretato come il valore del dominio DataOra che rappresenta l'istante corrente.

Risposta

<p>1 Tipo: Entità Relationship (cerchiare)</p> <p>Nome: <u>PERSONA</u></p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>attributo</th> <th>dominio</th> <th>moltep. (*)</th> </tr> </thead> <tbody> <tr> <td>NOME</td> <td>STINGA</td> <td></td> </tr> <tr> <td>COGNOME</td> <td>STINGA</td> <td></td> </tr> <tr> <td>NASCITA</td> <td>DATA</td> <td></td> </tr> <tr> <td>CF</td> <td>CODFIS</td> <td></td> </tr> </tbody> </table> <p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p> <ul style="list-style-type: none"> • $\forall p. PERSONA(p) \wedge \text{COMALTEZZA}(p) \rightarrow MEDICO(p)$ • $\forall p. PERSONA(p) \rightarrow MEDICO(p) \vee PAZIENTE(p)$ <p><i>Nota</i> indica che dopo sia stato ristrutturato</p>	attributo	dominio	moltep. (*)	NOME	STINGA		COGNOME	STINGA		NASCITA	DATA		CF	CODFIS		<p>2 Tipo: Entità Relationship (cerchiare)</p> <p>Nome: <u>PAZIENTE</u></p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>attributo</th> <th>dominio</th> <th>moltep. (*)</th> </tr> </thead> <tbody> <tr> <td>TEL</td> <td>TELEFONO</td> <td>(0, N)</td> </tr> <tr> <td>EMAIL</td> <td>EMAIL</td> <td>(0, 1)</td> </tr> <tr> <td>INDRIZZO</td> <td>INDRIZZO</td> <td></td> </tr> </tbody> </table> <p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p> <ul style="list-style-type: none"> • $\forall p, pr. PAZIENTE(p) \wedge MEDICO(pr) \rightarrow MEDPREST(p, pr)$ • $\forall p, pr, pr', i, i'. PAZIENTE(p) \wedge MEDPREST(p, pr) \wedge PAZPREST(p, pr') \wedge pr \neq pr' \wedge INIZIO(pr, i) \wedge INIZIO(pr', i') \rightarrow \exists t. DATAOMA(t) \wedge (i \leq t \wedge \forall t'. PRENE(p, t) \rightarrow t \leq i) \wedge (i' \leq t \wedge \forall t'. PRENE(pr', t) \rightarrow t \leq i')$ 	attributo	dominio	moltep. (*)	TEL	TELEFONO	(0, N)	EMAIL	EMAIL	(0, 1)	INDRIZZO	INDRIZZO	
attributo	dominio	moltep. (*)																										
NOME	STINGA																											
COGNOME	STINGA																											
NASCITA	DATA																											
CF	CODFIS																											
attributo	dominio	moltep. (*)																										
TEL	TELEFONO	(0, N)																										
EMAIL	EMAIL	(0, 1)																										
INDRIZZO	INDRIZZO																											

<input checked="" type="checkbox"/> 3	Tipo: Entità	Relationship (cerchiare)
Nome:	<u>SPECIALIZ</u>	
attributo	dominio	moltep. (*)
NOME		STRINGA
(*) solo se diversa da (1,1)		
Vincoli:		
<ul style="list-style-type: none"> • <i>V. SPECIALIZ. CICLO MEDICO</i> R $\forall s, m, p \text{ } SPECIALIZ(s) \wedge PREST_ESTERNA(p) \wedge$ $\text{SPEC_PREST}(p, s) \wedge MED_PREST(p, m)$ $\rightarrow SPEC_MED(m, s)$ 		

<input checked="" type="checkbox"/> 5	Tipo: Entità	Relationship (cerchiare)
Nome:	<u>PREST_ESTERNA</u>	
attributo	dominio	moltep. (*)
DESC		STRINGA
(*) solo se diversa da (1,1)		
Vincoli:		

<input checked="" type="checkbox"/> 4	Tipo: Entità	Relationship (cerchiare)
Nome:	<u>ARRESTAZIONE</u>	
attributo	dominio	moltep. (*)
INIZIO		DATA ORA
(*) solo se diversa da (1,1)		
Vincoli:		
<ul style="list-style-type: none"> • <i>V. ARRESTAZIONE. NASCITA PAZ</i> $\forall p, pr, n, i \text{ } ARRESTAZIONE(pr) \wedge INIZIO(pr, i) \wedge$ $\text{PAZ_PREST}(pr, p) \wedge NASCITA(p, n) \rightarrow N < i$ 		
<ul style="list-style-type: none"> • <i>V. ARRESTAZIONE. NASCITA MED</i> $\forall m, pr, n, i \text{ } ARRESTAZIONE(pr) \wedge INIZIO(pr, i) \wedge$ $\text{MED_PREST}(pr, m) \wedge NASCITA(m, n) \rightarrow N < i$ 		

<input checked="" type="checkbox"/> 6	Tipo: Entità	Relationship (cerchiare)
Nome:	<u>RICOVERO TERM</u>	
attributo	dominio	moltep. (*)
PINE		DATA ORA
(*) solo se diversa da (1,1)		
Vincoli:		
<ul style="list-style-type: none"> • <i>V. RICOVERO TERM. PINE</i> R $\forall r, i, x \text{ } RICOVERO_TERM(r) \wedge INIZIO(r, i) \wedge$ $\text{PINE}(r, x) \rightarrow i < x$ 		

7	Tipo: Entità Relationship (cerchiare)	
Nome:	STANZA	
attributo	dominio	moltep. (*)
NUM	INTERO>0	
MANO	INTERO>0	
SETTORE	INTERO>0	
MAXLETTI	INTERO[1,8]	

(*) solo se diversa da (1,1)

Vincoli:

$$\begin{aligned}
 & \text{• } V. STANZA. MAXLETTI \\
 & \forall s, \max \text{STANZA}(s) \wedge \text{MAXLETTI}, (s, \max) \\
 & \rightarrow |\{l \mid \text{STANZA}(s, l)\}| \leq \max \\
 & \text{NOTA: indica che algor sia stato} \\
 & \text{implementato come trigger}
 \end{aligned}$$

9	Tipo: Entità Relationship (cerchiare)	
Nome:	LETTO	
attributo	dominio	moltep. (*)

(*) solo se diversa da (1,1)

Vincoli:

$$\begin{aligned}
 & \text{• } V. LETTO. OCCUPAZIONE \\
 & \forall l, r, x, i, i' \\
 & \text{LETTO}(l) \wedge \text{LettoRic}(x, l) \wedge \text{Letto}(x', l) \wedge \\
 & x \neq x' \wedge \text{INIZIO}(x, i) \wedge \text{INIZIO}(x', i') \rightarrow \\
 & \exists t \text{ DATOAN}(t) \wedge (i \leq t \wedge (\forall f \text{ FINE}(x, f) \rightarrow t \leq f)) \\
 & \wedge (i' \leq t \wedge (\forall f' \text{ FINE}(x', f') \rightarrow t \leq f)))
 \end{aligned}$$

8	Tipo: Entità Relationship (cerchiare)	
Nome:	
attributo	dominio	moltep. (*)

(*) solo se diversa da (1,1)

Vincoli:

10	Tipo: Entità Relationship (cerchiare)	
Nome:	
attributo	dominio	moltep. (*)

(*) solo se diversa da (1,1)

Vincoli:

<p>11 Tipo: Entità Relationship (cerchiare)</p> <p>Nome:</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left; padding: 2px;">attributo</th> <th style="text-align: left; padding: 2px;">dominio</th> <th style="text-align: left; padding: 2px;">moltepl. (*)</th> </tr> </thead> <tbody> <tr> <td colspan="3" style="height: 100px;"></td> </tr> </tbody> </table> <p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p>	attributo	dominio	moltepl. (*)				<p>13 Tipo: Entità Relationship (cerchiare)</p> <p>Nome:</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left; padding: 2px;">attributo</th> <th style="text-align: left; padding: 2px;">dominio</th> <th style="text-align: left; padding: 2px;">moltepl. (*)</th> </tr> </thead> <tbody> <tr> <td colspan="3" style="height: 100px;"></td> </tr> </tbody> </table> <p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p>	attributo	dominio	moltepl. (*)			
attributo	dominio	moltepl. (*)											
attributo	dominio	moltepl. (*)											

<p>12 Tipo: Entità Relationship (cerchiare)</p> <p>Nome:</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left; padding: 2px;">attributo</th> <th style="text-align: left; padding: 2px;">dominio</th> <th style="text-align: left; padding: 2px;">moltepl. (*)</th> </tr> </thead> <tbody> <tr> <td colspan="3" style="height: 100px;"></td> </tr> </tbody> </table> <p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p>	attributo	dominio	moltepl. (*)				<p>14 Tipo: Entità Relationship (cerchiare)</p> <p>Nome:</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left; padding: 2px;">attributo</th> <th style="text-align: left; padding: 2px;">dominio</th> <th style="text-align: left; padding: 2px;">moltepl. (*)</th> </tr> </thead> <tbody> <tr> <td colspan="3" style="height: 100px;"></td> </tr> </tbody> </table> <p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p>	attributo	dominio	moltepl. (*)			
attributo	dominio	moltepl. (*)											
attributo	dominio	moltepl. (*)											

15 Tipo: **Entità | Relationship** (cerchiare)

Nome:

attributo	dominio	moltepl. (*)

(*) solo se diversa da (1,1)

Vincoli:

17 Tipo: **Entità | Relationship** (cerchiare)

Nome:

attributo	dominio	moltepl. (*)

(*) solo se diversa da (1,1)

Vincoli:

16 Tipo: **Entità | Relationship** (cerchiare)

Nome:

attributo	dominio	moltepl. (*)

(*) solo se diversa da (1,1)

Vincoli:

18 Tipo: **Entità | Relationship** (cerchiare)

Nome:

attributo	dominio	moltepl. (*)

(*) solo se diversa da (1,1)

Vincoli:

Ulteriori vincoli esterni, specifica di eventuali operazioni ausiliarie invocate da tali vincoli, e specifica dei domini concettuali non di tipo base

DOMINIO CODFIS:

COD: STRINGA ALFANUMERICA
SECONDO STANDART

DOMINIO INDIRIZZO:

VIA: STRINGA
CIVICO: INTERO 0-70
CAP: INTERO 0-70
CITTÀ: STRINGA
NAZIONE: STRINGA

DOMINIO TELEFONO:

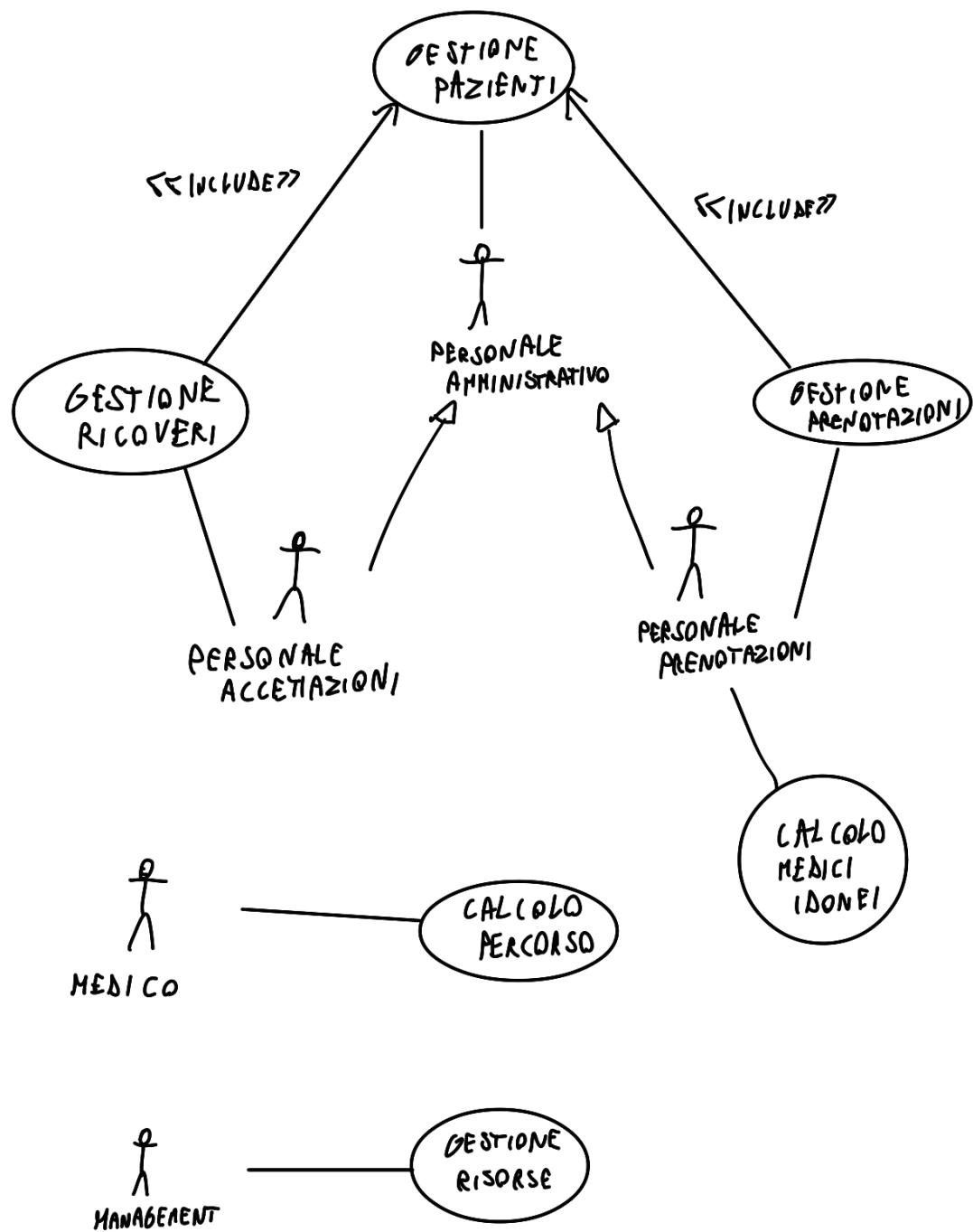
PRE: STRINGA NUMERICA
SUFF: STRINGA NUMERICA

DOMINIO EMAIL:

E: STRINGA SECONDO
STANDARD

Risposta alla Domanda 2 (segue)

Domanda 3 (5 minuti; 10 minuti al massimo) Proseguire la fase di Analisi Concettuale dei requisiti, producendo un diagramma UML degli use-case che definisca ad alto livello tutte le funzionalità richieste al sistema.

Risposta

Domanda 4 (10 minuti) Proseguire la fase di Analisi Concettuale dei requisiti definendo le operazioni degli use-case.

In particolare, per ogni use-case definito nella risposta alla **Domanda 3** definire la **segnatura** di tutte le operazioni che lo compongono, in termini di nome dell'operazione, nomi e dominio concettuale degli argomenti, dominio concettuale dell'eventuale valore di ritorno.

[1] Specifica use-case: ...**GESTIONE PAZIENTI**..... (nome use-case)

Operazioni dello use-case:

REGISTRA PAZIENTE(*CF: CodFis, n: STRINGA, c: STRINGA, d: DATA, t: TELEFONO (0, N), E: EMAIL (0, 1), i: INDIRIZZO*): PAZIENTE

STORICO PAZIENTE(*p: PAZIENTE*): PRESTAZIONE (0, N)

[2] Specifica use-case: ...**GESTIONE RICOVERI**..... (nome use-case)

Operazioni dello use-case:

AMMETTI PAZIENTE (*p: PAZIENTE, m: MEDICO, d: DATAORA*): RICOVERO

DIMETTI PAZIENTE (*r: RICOVERO*)

XNUM Posti DISPONIBILI (*r: STANZA, d: DATAORA*): INTERO ≥ 0

[3] Specifica use-case: ...**GESTIONE PRENOTAZIONI**..... (nome use-case)

Operazioni dello use-case:

REGISTRA PREST ESTERNA(*p: PAZIENTE, m: MEDICO, s: SPECIALIZ, d: STRINGA, f: DATAORA*): PREST ESTERNA

4 Specifica use-case: ... CALCOLO MEDICI IDONEI (nome use-case)

Operazioni dello use-case:

■ CALCOLA MEDICI IDONEI (S: SPECIALIZ): MEDICO (0, N)

5 Specifica use-case: ... CALCOLO PERCORSO (nome use-case)

Operazioni dello use-case:

■ CALCOLA PERCORSO (M: MEDICO): (A: STANZA, N: INTERO > 0) (0, N)

✗ SORT BY PIANO SETTORE (A: STANZA, A': STANZA): BOOLEAN

6 Specifica use-case: ... GESTIONE RISORSE (nome use-case)

Operazioni dello use-case:

AGGIUNGI LETTO (A: STANZA): LETTO

RIMUOVI LETTO (L: LETTO)

7 Specifica use-case: (nome use-case)

Operazioni dello use-case:

Domanda 5 (30 minuti; 60 minuti al massimo) Proseguire la fase di Analisi Concettuale dei requisiti producendo le specifiche concettuali per le operazioni di use-case, **limitandosi** a quelle necessarie a modellare i requisiti contrassegnati dalla barra laterale (come quella qui a sinistra). In particolare, per ogni operazione, definire segnatura, precondizioni e postcondizioni utilizzando il linguaggio della logica del primo ordine. Si assuma lo stesso vocabolario definito alla **Domanda 2**.

Una risposta soddisfacente a questa domanda è condizione *necessaria* (ma non sufficiente) per superare la prova.

Risposta

CALCOLA MED / DONEI (α : SPECIALIZZ): MEDICO ($0, n$)

PRE:

POST:

$$M_p = \{m \mid \text{SPECPRIMMED}(m, \alpha)\}$$

$$M_d = \{m \mid \text{SPECMED}(m, \alpha)\}$$

$$(M_p \neq \emptyset \rightarrow \text{RESULT} = M_p) \wedge (M_p = \emptyset \rightarrow \text{RESULT} = M_d)$$

CALCOLA PERCORSO (m : MEDICO): (α : STANZA, n : INTERO > 0) ($0, n$)

PREF:

POST:

$$S = \{n \mid \begin{array}{l} \text{STANZA}(\alpha) \wedge (\exists l, x \text{ STLETTORIC}(l, \alpha)) \\ \wedge \text{LETTORIC}(x, l) \wedge \text{MEDRIC}(x, m) \\ \wedge \neg \text{RICOVEROTERM}(x) \end{array}\}$$

$$\text{RESULT} = \text{SORTED}(S, \text{SORT BY PIANOSETTO})$$

SORT BY PIANO SETTORE (α : STANZA, α' : STANZA): BOOLEAN

PRE:

POST:

diammi p, p', α, α' tali che

$$\text{PIANO}(\alpha, p) \wedge \text{PIANO}(\alpha', p') \wedge \text{SETTORE}(\alpha, \alpha') \wedge \text{SETTORE}(\alpha', \alpha')$$

$$((p < p' \vee (p = p' \wedge \alpha \leq \alpha')) \rightarrow \text{RESULT} = \text{TRUE})$$

$$((p > p' \vee (p = p' \wedge \alpha > \alpha')) \rightarrow \text{RESULT} = \text{FALSE})$$

Risposta alla Domanda 5 (segue)

$\text{NUM POSTI DISPONIBILI}(\alpha: \text{STANZA}, d: \text{DATAORA}) : \text{INTERO} \geq 0$

PRE:

POST:

$$L = \left\{ l \mid \begin{array}{l} \text{STLETT}(l, \alpha) \wedge (\nexists r, i \text{ RICOVERO}(r) \wedge \text{RICLETT}(l, r)) \\ \wedge \text{TRICOVTERM}(r) \wedge \text{INIZIO}(r, i) \wedge i \leq d \end{array} \right\}$$

$$\text{RESULT} = |L|$$

$\text{AMMETTI PAZIENTE}(\rho: \text{PAZIENTE}, m: \text{MEDICO}, d: \text{DATAORA}) : \text{RICOVERO}$

PRE: $\exists \alpha \text{ STANZA}(\alpha) \wedge \text{NUM POSTI DISPONIBILI}(\alpha, d) \geq 0$

POST:

• Sia l tale che

$$\begin{aligned} & \text{LETTO}(l) \wedge (\exists \alpha \text{ STANZA}(\alpha) \wedge \text{STLETT}(l, \alpha) \\ & \quad \text{NUM POSTI DISPONIBILI}(\alpha, d) \geq 0) \end{aligned}$$

• VIENE AGGIUNTO L'ELEMENTO d

• VENGONO AGGIUNTI I PREDICATI:

$\text{RICOVERO}(l)$

$\text{RICLETT}(l, d)$

$\text{PAZPREST}(d, \rho)$

$\text{MEDPREST}(l, m)$

$\text{INIZIO}(l, d)$

2 Progettazione della base dati e delle funzionalità

Domanda 6 (20 minuti; 30 minuti al massimo) Iniziare la fase di progettazione logica della base di dati decidendo il DBMS da utilizzare e ristrutturando lo schema ER concettuale, il dizionario dei dati e i vincoli esterni. In particolare:

- progettare una corrispondenza tra i domini concettuali ed opportuni domini SQL (domini base o utente, oppure realizzati mediante relazioni aggiuntive) supportati dal DBMS scelto
- eliminare attributi multivale o composti
- eliminare relazioni is-a e generalizzazioni
- definire un identificatore primario per ogni entità
- valutare se e come aggiungere ridondanza in maniera controllata
- ristrutturare i vincoli esterni per renderli consistenti con la struttura del nuovo diagramma.

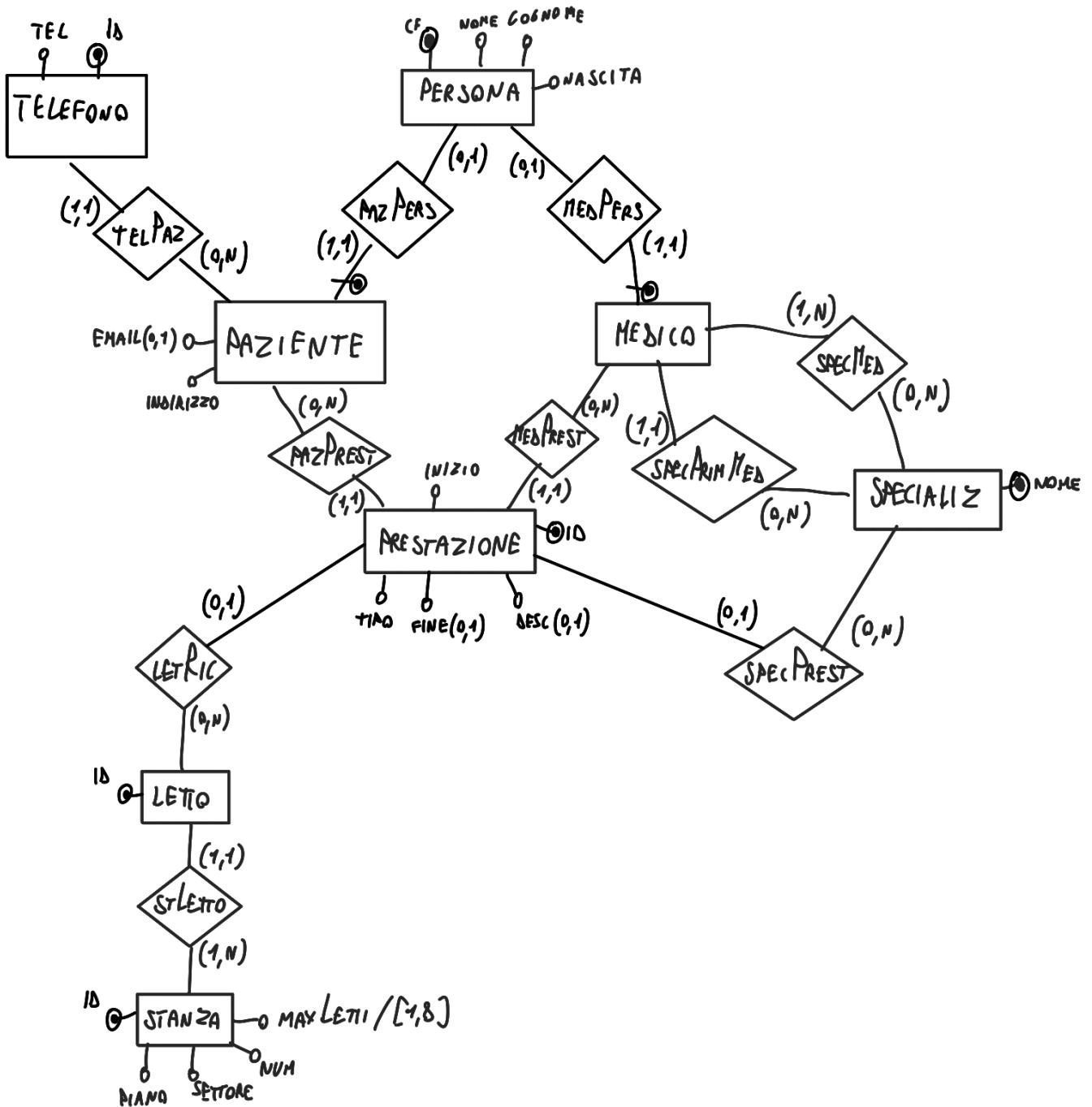
Descrivere brevemente le principali scelte effettuate.

DBMS da utilizzare **POSTGRESQL**.....

Corrispondenza tra domini concettuali e domini supportati dal DBMS

- `CREATE DOMAIN INT-GZ AS INTEGER CHECK (VALUE >= 0)`
- `CREATE DOMAIN INT-GEZ AS INTEGER CHECK (VALUE >= 0)`
- `CREATE DOMAIN STR AS VARCHAR(200)`
- `CREATE DOMAIN CF AS STR CHECK (isValidCF(VALUE))`
- `CREATE DOMAIN EMAIL AS STR CHECK (isValidEmail(VALUE))`
- `CREATE TYPE INDIRIZZO AS (
 VIA STR,
 CIVICO INT-GZ,
 CAP INT-GZ,
 CITA' STR,
 NAZIONE STR
)`
- `CREATE TYPE TELEFONO AS(
 PAE STR,
 SUFF STR,
)`
- `CREATE TYPE TIPO_PREST AS ENUM('RIC', 'EST')`

Diagramma ER ristrutturato



Breve descrizione delle scelte effettuate durante la ristrutturazione

NOTA: indica che
non è stato implementato
certo trigger (es:
come v. di envelope)

Vincoli esterni introdotti o modificati durante la fase di ristrutturazione

(si omettano i vincoli esterni la cui formulazione è rimasta identica a seguito della ristrutturazione)

• V. PERSONA. COMPLETEZZA T

$$\forall x \text{ PERSONA}(x) \rightarrow (\exists p \text{ PAZPERS}(x, p)) \vee (\exists m \text{ MEDPERS}(x, m))$$

x • V. PERSONA. NO_SVOMEDICO

$$\forall x, p, m \text{ PERSONA}(x) \wedge \text{PAZPERS}(x, p) \wedge \text{PAZPREFST}(p, pr) \wedge \text{MEDPREFST}(m, pr) \rightarrow \neg \text{MEDPERS}(x, m)$$

x • V. SPEC_PRIM_MED_ISA

$$\forall m, s \text{ MEDICO}(m) \wedge \text{SPECPRIMMED}(m, s) \rightarrow \text{SPECMED}(m, s)$$

x • V. PRESTAZIONE. RICOVERO

$$\forall p \text{ PRESTAZIONE}(p) \rightarrow (\exists l \text{ LETRIC}(p, l) \leftrightarrow \text{TIPO}(p, 'RIC'))$$

x • V. PRESTAZIONE. ASTERNA

$$\forall p \text{ PRESTAZIONE}(p) \rightarrow (\exists a, d \text{ SPECREST}(p, a) \wedge \text{DESC}(p, d) \leftrightarrow \text{TIPO}(p, 'RIC'))$$

x • V. PRESTAZIONE. PINE

$$\forall p \text{ PRESTAZIONE}(p) \rightarrow (\exists f \text{ PINE}(p, f) \rightarrow \text{TIPO}(p, 'RIC'))$$

Risposta alla Domanda 6 (segue)

x • V. RICOVERO TERM. FINI

$$\forall p, i \text{ PRESTAZIONE}(p) \wedge \text{INIZIO}(p, i) \rightarrow (\forall k \text{ FINE}(p, k) \rightarrow i < k)$$

• V. SPECIALIZ. CICLO MEDICO

$$\begin{aligned} \forall s, m, p \quad & \text{SPECIALIZ}(s) \wedge \text{PRESTAZIONE}(p) \wedge \text{TIPO}(p, 'EST') \wedge \text{SPECREST}(p, s) \\ & \wedge \text{MEDPREST}(p, m) \rightarrow \text{SPEC_MED}(m, s) \end{aligned}$$

Domanda 7 (30 minuti; 60 minuti al massimo) Proseguire la fase di progettazione logica della base di dati producendo lo schema relazionale della base dati e i relativi vincoli a partire dallo schema ER ristrutturato.

Una risposta soddisfacente a questa domanda è condizione *necessaria* (ma non sufficiente) per superare la prova.

1 Relazione	<u>TELEFONO</u>	(nome)	Derivante da:	entità	relationship (cerchiare)
Attributi	<u>ID</u>	TEL	PAZ		
Domini	INTEGER	TELEFONO	CF		

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

FK: ID
FK: (PAZ) REF PAZIENTE(PERS)

La relazione accorda le relazioni che implementano le seguenti relationship: **TELPAZ**

2 Relazione	<u>PERSONA</u>	(nome)	Derivante da:	entità	relationship (cerchiare)
Attributi	<u>CF</u>	NOME	COGNOME	MASCITA	
Domini	CODFIS	STR	STR	TIMESTAMP	

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

La relazione accorda le relazioni che implementano le seguenti relationship:

3 Relazione	<u>PAZIENTE</u>	(nome)	Derivante da:	entità	relationship (cerchiare)
Attributi	<u>PERS</u>	EMAIL*	INDRIZZO		
Domini	CP	EMAIL	INDRIZZO		

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

fk: (PERS) REF PERSONA(CF)

La relazione accorda le relazioni che implementano le seguenti relationship: **PAZPERS**

4 Relazione	<u>MEDICO</u>	(nome)	Derivante da:	entità	relationship (cerchiare)
Attributi	<u>PERS</u>	SPEC PRIM			
Domini	CF	STR			

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

fk: (PERS) REF PERSONA(CF)

Fk: (PERS, SPECPRIM) REF SPECMED(MED, SPEC)

FK: (SPECPRIM) REF SPECIALIZ(NOME)

La relazione accorda le relazioni che implementano le seguenti relationship: **MEDPERS, SPECPRIMMED**

5 Relazione	<u>SPECMED</u>	(nome)	Derivante da:	entità	relationship (cerchiare)
Attributi	<u>MED</u>	SPEC			
Domini	CF	STR			

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

fk: (MED) REF MEDICO(PERS)

FK: (SPEC) REF SPECIALIZ(NOME)

La relazione accorda le relazioni che implementano le seguenti relationship:

6 Relazione .SPECIALIZ... (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>NAME</u>	
Domini STR	

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

La relazione accorda le relazioni che implementano le seguenti relationship:

7 Relazione .PRESTAZIONE... (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>ID</u> INIZIO FINE* PAZ MED TIPO DESC* LETTO*	
Domini INTEGER TIMESTAMP TIMESTAMP CF CF TIPO_PREST STR INTEGER	

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

FK: (PAZ)REF PATIENTE(PERS) ENN: TIPO='EST' \leftrightarrow SPEC=NULL
 FK: (MED)REF MEDICO(PERS) ENN: FINE=NULL \rightarrow INIZIO < FINE
 FK: (LETO)REF LETTO(ID) ENN: DESC=NULL \rightarrow TIPO='EST'

ENN: PAZ \neq MED
 FK: (SPEC)REF SPECIALIZ(nome)
 ENN: TIPO='RIC' \leftrightarrow LETTO=NULL
 ENN: FINE=NULL \rightarrow TIPO='NC'

SERIAL:ID
 SPEC
 STR

La relazione accorda le relazioni che implementano le seguenti relationship: PAZPREST, MEDPREST, LETTIC, SPECPREST

8 Relazione .LETTO..... (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>ID</u> STANZA	
Domini INTEGER INTEGER	

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

SERIAL:ID
 FK: (STANZA)REF STANZA(ID)

La relazione accorda le relazioni che implementano le seguenti relationship: STLETTO

9 Relazione .STANZA..... (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>ID</u> NUM PIANO SETTORE MAX-LETTI	
Domini INTEGER INT_6Z INT_6Z INT_6Z INTEGER	

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

SERIAL:ID
 DOMINIO: MAX-LETTI > 4 \wedge MAX-LETTI < 8
 INC: (STANZA) \subseteq LETTO(STANZA)

La relazione accorda le relazioni che implementano le seguenti relationship:

10 Relazione (nome)	Derivante da: entità relationship (cerchiare)
Attributi	
Domini	

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

La relazione accorda le relazioni che implementano le seguenti relationship:

11 Relazione (nome)	Derivante da: entità relationship (cerchiare)
Attributi	
Domini	

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

La relazione accorda le relazioni che implementano le seguenti relationship:

12 Relazione (nome)	Derivante da: entità relationship (cerchiare)
Attributi	
Domini	

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

La relazione accorda le relazioni che implementano le seguenti relationship:

13 Relazione (nome)	Derivante da: entità relationship (cerchiare)
Attributi	
Domini	

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

La relazione accorda le relazioni che implementano le seguenti relationship:

14 Relazione (nome)	Derivante da: entità relationship (cerchiare)
Attributi	
Domini	

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

La relazione accorda le relazioni che implementano le seguenti relationship:

15 Relazione (nome)	Derivante da: entità relationship (cerchiare)
Attributi	
Domini	

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

La relazione accorda le relazioni che implementano le seguenti relationship:

16	Relazione (nome)	Derivante da: entità relationship (cerchiare)
Attributi		
Domini		

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

La relazione accorda le relazioni che implementano le seguenti relationship:

17	Relazione (nome)	Derivante da: entità relationship (cerchiare)
Attributi		
Domini		

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

La relazione accorda le relazioni che implementano le seguenti relationship:

18	Relazione (nome)	Derivante da: entità relationship (cerchiare)
Attributi		
Domini		

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

La relazione accorda le relazioni che implementano le seguenti relationship:

19	Relazione (nome)	Derivante da: entità relationship (cerchiare)
Attributi		
Domini		

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

La relazione accorda le relazioni che implementano le seguenti relationship:

20	Relazione (nome)	Derivante da: entità relationship (cerchiare)
Attributi		
Domini		

Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

Vincoli (foreign key, inclusione, altra chiave, di ennupla, di dominio):

La relazione accorda le relazioni che implementano le seguenti relationship:

Ulteriori vincoli esterni

Per ogni ulteriore vincolo esterno (non ancora espresso perché non definibile mediante vincoli di chiave, foreign key, ennupla, dominio, inclusione), progettare un trigger che lo implementi, definendo: (a) gli eventi da intercettare (inserimento, modifica, eliminazione di ennuple); (b) quando intercettare tali eventi (appena prima o subito dopo l'evento intercettato); (c) la relativa funzione in pseudo-codice con SQL immerso che implementa il controllo del vincolo.

- T. STANZA. MAXLETTI

- INSERIMENTO IN LETTO
- PRE-OPERAZIONE

$$Q = \left(\begin{array}{l} \text{SELECT COUNT}(l.\text{id}) \text{ AS } c, \\ \text{STANZA } A \\ \text{FROM LETTI } l, \text{STANZA } A \\ \text{WHERE } l.\text{STANZA} = \text{NEW.STANZA} \\ \text{AND } l.\text{STANZA} = A.\text{id} \end{array} \right)$$

IF $Q.c < Q.m$

 COMMIT

ELSE

 GENERA ERRORE

- T. PERSONA. COMPLETEZZA

- INSERIMENTO IN PERSONA
- PRE-OPERAZIONE

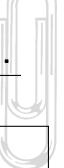
$$\text{IS VALID} = \left(\text{EXISTS} \left(\begin{array}{l} \text{SELECT *} \\ \text{FROM PAZIENTE } P, MEDICO } m \\ \text{WHERE } P.\text{PERS} = \text{NEW.CF} \text{ OR } m.\text{PERS} = \text{NEW.CF} \end{array} \right) \right)$$

IF IS VALID

 COMMIT

ELSE

 GENERA ERRORE



Risposta alla Domanda 7 (segue)

Domanda 8 (30 minuti; 45 minuti al massimo) Proseguire la fase di progettazione dell'applicazione producendo le specifiche realizzative delle operazioni di use-case definite per modellare i requisiti contrassegnati dalla barra laterale della specifica dei requisiti.

In particolare, per ogni operazione definire la segnatura, in termini di nome dell'operazione, nomi e dominio SQL degli argomenti, dominio SQL dell'eventuale valore di ritorno, e un algoritmo in pseudo-codice con SQL immerso che verifichi le precondizioni e garantisca il raggiungimento delle postcondizioni definite in fase di Analisi.

Una risposta soddisfacente a questa domanda è condizione *necessaria* (ma non sufficiente) per superare la prova.

Risposta

CALCOLA MED / DOMEI (A: STR): INSieme (<m: CF>)

$Q = \left(\begin{array}{l} \text{SELECT } M.ID \\ \text{FROM MEDICI } M \\ \text{WHERE } M.SPECPRIM = :A \end{array} \right)$

IF $Q \neq \text{NULL}$
RETURN Q

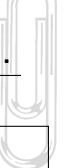
ELSE
 $Q' = \left(\begin{array}{l} \text{SELECT } M.MED \text{ AS } m \\ \text{FROM SPECMED } M \\ \text{WHERE } M.P. SPEC = :A \end{array} \right)$

RETURN Q'

CALCOLA PERCORSO (m: CF): LISTA (<A: INTEGER>)

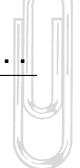
$Q = \left(\begin{array}{l} \text{SELECT } A.ID \\ \text{FROM STANZA } A, LETTO } l, \text{ PRESTAZIONE } P \\ \text{WHERE } A.ID = l.STANZA \text{ AND } P.TIPO = 'ALC' \text{ AND } \\ P.LETTO = l.ID \text{ AND } P.MED = :m \\ \text{ORDER BY } A.PIANO, A.SETTORE \end{array} \right)$

RETURN Q



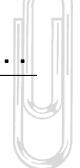
Risposta alla Domanda 8 (segue)

Tempo totale stimato per svolgere questa prova: 180 minuti (tempo totale concesso: 300 minuti).
[Spazio per minute. Questa pagina non sarà valutata a meno che non sia puntata da pagine precedenti.]



[Spazio per minute. Questa pagina non sarà valutata a meno che non sia puntata da pagine precedenti.]

[Spazio per minute. Questa pagina non sarà valutata a meno che non sia puntata da pagine precedenti.]



[Spazio per minute. Questa pagina non sarà valutata a meno che non sia puntata da pagine precedenti.]