



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.ER – Modulo risposte prova scritta

Dati dello studente e dell'esame

Cognome e nome: Alessandro Gattielli Matricola: 204185

Data: 03/06/24

Corso di laurea e canale di appartenenza:

- Laurea in Informatica, canale 1 (A-L, Prof. G. Perelli)
 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

Utente

Nome
Cognome
Sesso
Data nascita
Residenza

Habit
Destinazione
Destinazione Mezzi
N°Membri

Indisponibilità
DA Dati
A Dati

Prestazioni

Dati
N°Percorso
Posti letto

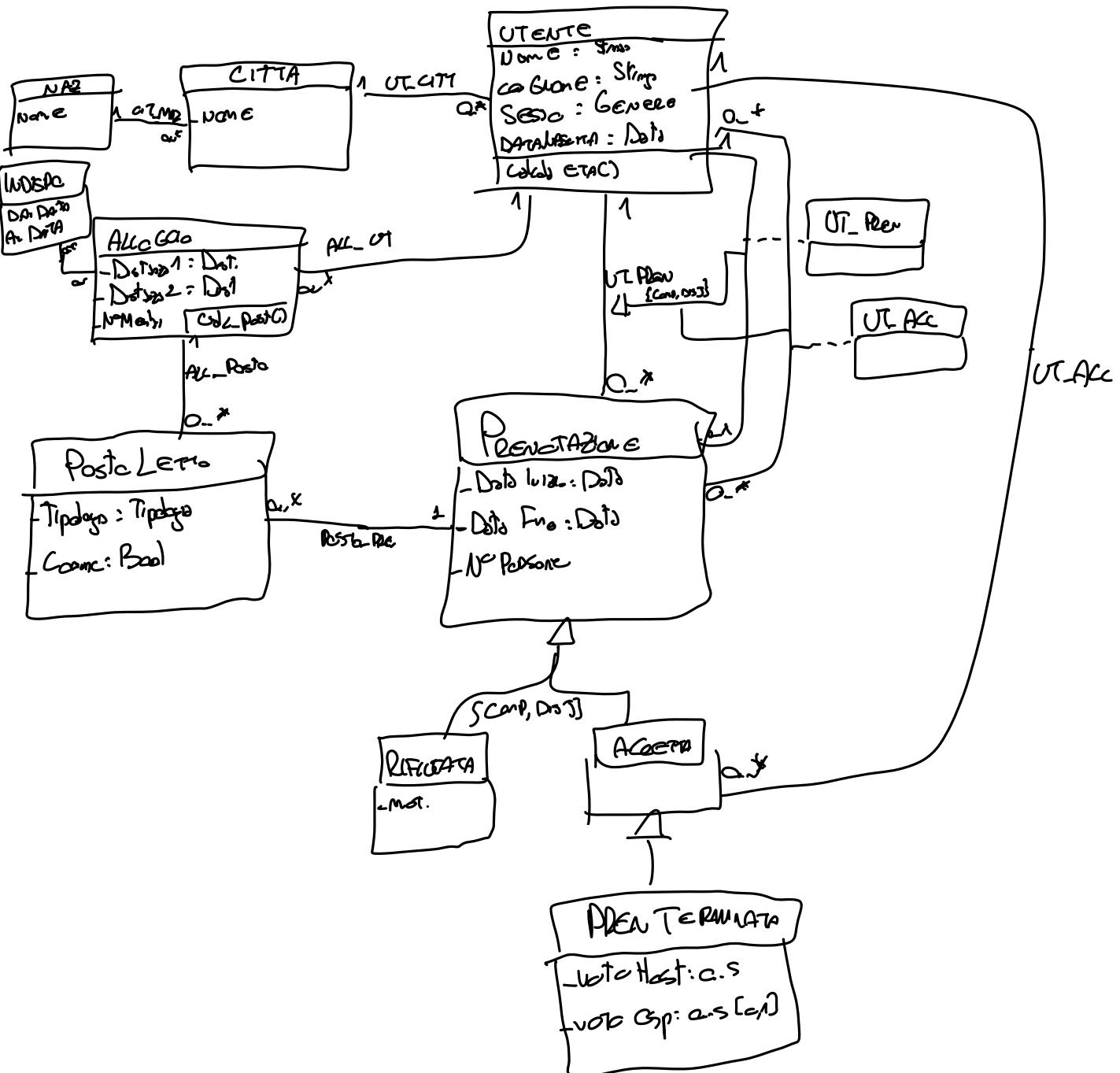
Risposta alla Domanda 1 (segue)

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: <i>Rapporto</i></p> <table border="1" data-bbox="249 1358 822 1386"> <thead> <tr> <th>attributo</th> <th>dominio</th> <th>moltep. (*)</th> </tr> </thead> </table>	attributo	dominio	moltep. (*)	<p>2 Tipo: Entità Relationship (cerchiare)</p> <p>Nome: <i>Rapporto</i></p> <table border="1" data-bbox="827 1358 1538 1386"> <thead> <tr> <th>attributo</th> <th>dominio</th> <th>moltep. (*)</th> </tr> </thead> </table>	attributo	dominio	moltep. (*)
attributo	dominio	moltep. (*)					
attributo	dominio	moltep. (*)					
<p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p> <p>[V. Rapporto. InizPra Fun]</p> $\forall p, i, f \quad [\text{Prodotto}(p) \wedge \text{InizPra}(p, i) \wedge \text{Funzione}(p, f)] \rightarrow i < f$ <p>[V. Rapporto. No hd]</p> $\forall p, i, f, p_1, i_1, f_1, p_2, i_2, f_2 \quad [\text{Prodotto}(p) \wedge \text{InizPra}(p, i) \wedge \text{Funzione}(p, f) \wedge \text{Inizhd}(p) \wedge \text{Postdhd}(p) \wedge \text{posto-pre}(p, p) \wedge \text{Allerg}(i) \wedge \text{All-posto}(A, p) \wedge \text{indisponibili}(i_1) \wedge \text{Inizhd}(i_1, i_2) \wedge \text{Fine hd}(i_1, i_2) \wedge \text{Indhd}(A, i_1)] \rightarrow (i_1 < i_2 \vee i_2 < i_1)$	<p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p> <p>[V. Rapporto. No hd]</p> $\forall p_1, p_2, i_1, i_2, f_1, f_2, p \quad [\text{Prodotto}(p_1) \wedge \text{Prodotto}(p_2) \wedge \text{Inizhd}(p_1, i_1) \wedge \text{Inizhd}(p_2, i_2) \wedge \text{Funzione}(p_1, f_1) \wedge \text{Funzione}(p_2, f_2) \wedge \text{Inizhd}(p_1, i_1) \wedge \text{Inizhd}(p_2, i_2) \wedge \text{Funzione}(p_1, f_1) \wedge \text{Funzione}(p_2, f_2) \wedge \text{Postdhd}(p) \wedge \text{posto-pre}(p_1, p) \wedge \text{posto-pre}(p_2, p)] \rightarrow (i_1 > f_1 \vee i_2 > f_2)$						

<input type="checkbox"/> Tipo: Entità Relationship (cerchiare)
Nome: Postaletta
attributo dominio moltep. (*)
(*) solo se diversa da (1,1)

<input type="checkbox"/> Tipo: Entità Relationship (cerchiare)
Nome: Utile
attributo dominio moltep. (*)
(*) solo se diversa da (1,1)

<input type="checkbox"/> Tipo: Entità Relationship (cerchiare)
Nome: Utile
attributo dominio moltep. (*)
(*) solo se diversa da (1,1)

<input type="checkbox"/> Tipo: Entità Relationship (cerchiare)
Nome: Prendere
attributo dominio moltep. (*)
(*) solo se diversa da (1,1)

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

Nome: (Indipendenza)/

attributo	dominio	moltep. (*)

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

Vincoli:

(Vincolo) $\exists k \text{ fnc}(h_k)$

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

Nome: Utak

attributo	dominio	moltep. (*)

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

Vincoli:

[$\forall v, p [Utak(v) \wedge \exists k fnc(h_k) \wedge T_{cont}(p) \wedge \forall t - pre(t, p)] \rightarrow [\forall p_1 T_{cont}(p_1) \wedge \forall t - pre(t, p_1) \exists k votato(t, p_1)]$]

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:

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

Nome:

attributo	dominio	moltep. (*)
-----------	---------	-------------

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

Vincoli:

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

Nome:

attributo	dominio	moltep. (*)
-----------	---------	-------------

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

Vincoli:

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

Nome:

attributo	dominio	moltep. (*)
-----------	---------	-------------

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

Vincoli:

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

Nome:

attributo	dominio	moltep. (*)
-----------	---------	-------------

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

Vincoli:

15 Tipo: Entità | Relationship (cerchiare)

Nome:

attributo	dominio	moltep. (*)
-----------	---------	-------------

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

Vincoli:

17 Tipo: Entità | Relationship (cerchiare)

Nome:

attributo	dominio	moltep. (*)
-----------	---------	-------------

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

Vincoli:

16 Tipo: Entità | Relationship (cerchiare)

Nome:

attributo	dominio	moltep. (*)
-----------	---------	-------------

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

Vincoli:

18 Tipo: Entità | Relationship (cerchiare)

Nome:

attributo	dominio	moltep. (*)
-----------	---------	-------------

(*) 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

$$\text{GENTE} = \{\text{"Maschio"}, \text{"Femmina"}\}$$

$$\text{DISTANZA} = \{km: \text{Lunga} \geq 1, \text{metri} \text{ intero} \geq 0\}$$

Operazione Class

Allegato:

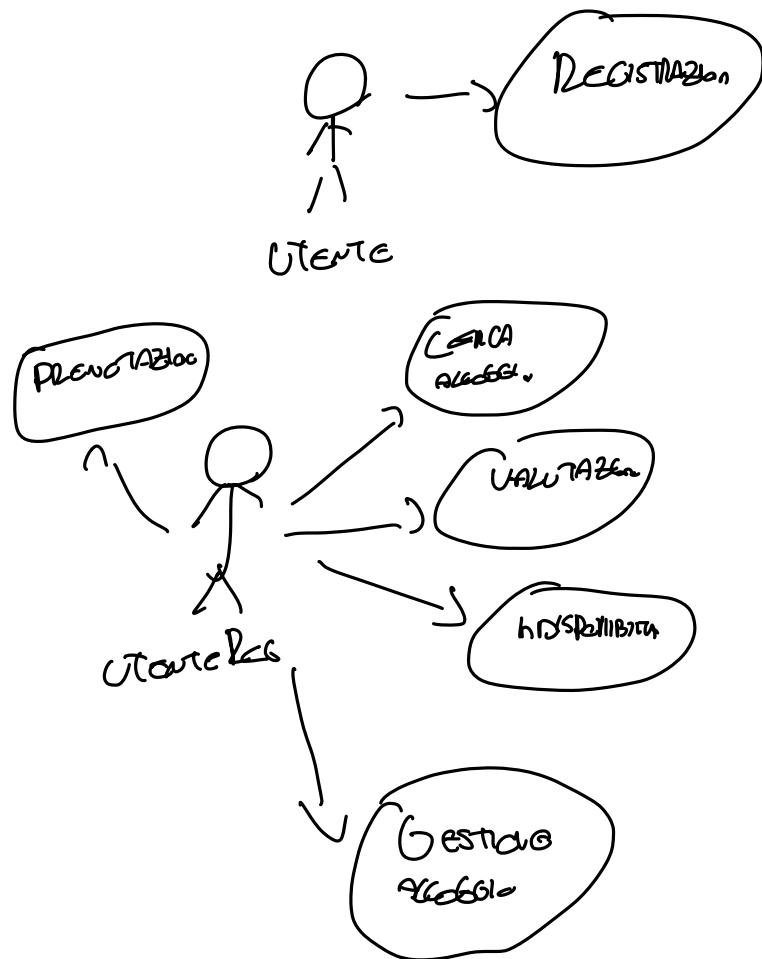
calc-post(): Intero > 0

pre: nessuno
post: $\exists R \in \{Cpl, M\} \text{ Post}(R) \cap \text{Acl-Post}(Th_{sp, pl})$

Result = 14

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: REGISTRAZIONE (nome use-case)

Operazioni dello use-case:

- REGISTRARc (v: Stms, L: Stmp, Dn: Dati, g: Genere) : Utente

2 Specifica use-case: PAGAMENTO (nome use-case)

Operazioni dello use-case:

- PAGAMENTO (D: Dati; D1, D2, np: Intero > 0) : Piatto

3 Specifica use-case: CERCA_ALLOGGIO (nome use-case)

Operazioni dello use-case:

- CERCA_ALLOGGIO (Cti, Df, L, NP) : Alloggio [L-*]

4 Specifica use-case: ...Indisp... (nome use-case)

Operazioni dello use-case:

Indisp... (p1:Dato; p2:Dato; p3:Dato) : Indisp...

5 Specifica use-case: ...Gestio...c. Alloggi... (nome use-case)

Operazioni dello use-case:

Gestione Alloggio (p1:Dato; p2:Dato; p3:Interazione) : Alloggio

Accetta - prn (p:Protocollo) : Accetto

Refusa - prn (p:Protocollo, m:Spieg) : Refusa

6 Specifica use-case: ...VALUTAZIONE... (nome use-case)

Operazioni dello use-case:

VALUTA (v:int>0)

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

Dominio ($U: S_{in}, L: S_{out}, D: D_{in}, g: G_{out}$): Utile

P Re:

Post: Modifica il file esteso $\Rightarrow M_b \neq N_{out}$

Eboati di D aggiunto: p

Flotti di D nuovo: nessuno

Tuple Aggiunte:

none(p, n)

legge(p, c)

DATA-NEXT(p, D_i)

Genere = 6

Result: p

Prova ($D: D_{in}, D_i, D_{out}, np: h_{max} > 0, a: Alleg.$): Pronto

pre: $np \neq 0 \wedge$ l'alloc di nuovo l' use-case, sia Posto $\{p\} \mid$ post- $pc(p) \wedge$ post- $al(a, p)$

Post:

Modifica il file est $\Rightarrow M_b \neq M_{out}$

Eboati di D aggiunto: p: Pronto

Flotti di D nuovo: nessuno

Tuple Aggiunte:

DATA-NEXT(p, D_i)

DATA-FUL(p, D_i)

UT-PRN(C_i, P)

Varie cond. np associazion post- $pc(p, p)$ per ogg. $p \in Post$

Risposta alla Domanda 5 (segue)

CERCA-Alleggio (Qi, Dt, C, NP) : Alleggi, L...*)

Pre: $D \in Dt \wedge \text{Alleg}(c) \wedge Ni \geq 2$

Post: Non modifca il livello estensionale $\Rightarrow M_o = M_{o'}$

$A = \left\{ \begin{array}{l} \text{Alleggi}(c) \wedge \left[\exists u \text{ Utet}(c) \wedge \text{ut_all}(c, u) \wedge \text{ut_c_it}(u, c) \right] \wedge \\ \exists k \text{ calc_part}(c, k) \wedge k \in NP \wedge \left[\forall i, ii, fi \text{ Indepart}(c) \wedge \text{all_ind}(c, i) \wedge \text{Indepart}(c, ii) \wedge \right. \\ \left. \text{Indepart}(c, fi) \right] \rightarrow D \in Dt \wedge \forall i \in NP \exists c_i \end{array} \right\}$

Risultato A

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.

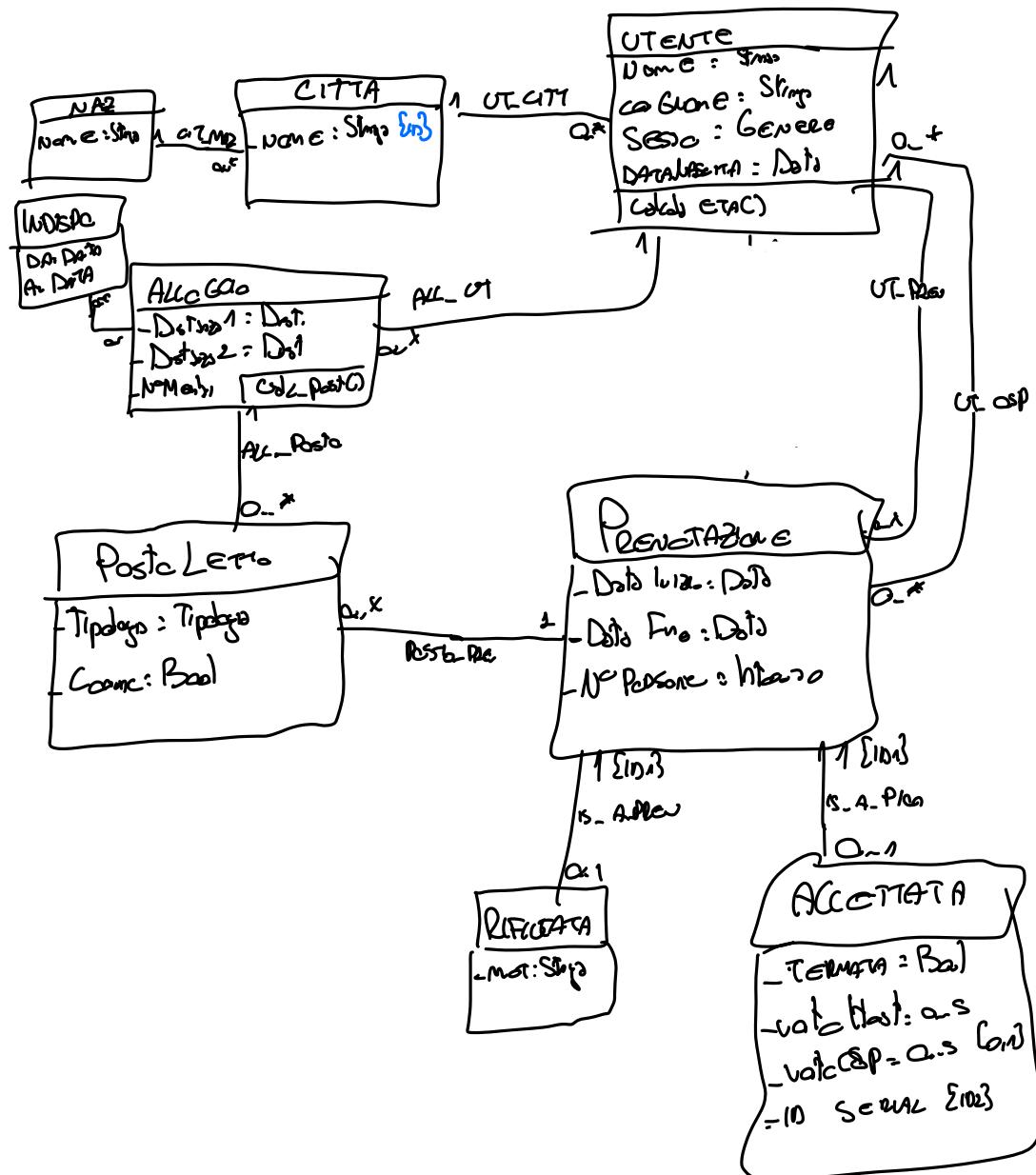
Descrivere brevemente le principali scelte effettuate.

Post SQL

DBMS da utilizzare
 Corrispondenza tra domini concettuali e domini supportati dal DBMS

Crotc Domn.	String	as	varchar	(check not null)
Crotc Domn.	Volc	as	Integer	(check value >= 0 and value <= 5)
Crotc type	base	as	Enum ('Mash.', 'Fcam')	
Crotc Domn.	Intervc	as	Integer	(check value > 0)
Crotc type			Distr = {K1, K2, ... ; metr = hbcrc}	
Crotc type	Tipologis	as	enum : { 'Singl', 'Doppio', 'Duo' }	

Diagramma ER ristrutturato



MANGANO TUTTI GLI IN = SERVIR {IN}

Breve descrizione delle scelte effettuate durante la ristrutturazione

Ho scelto ~~l'ordine~~ c'è un solo db. la generalizzazione. Riferimento - Accettato rispetto a

Vincolo Esterno

Ho Dato la Generalizzazione Prendendo Cognizione di Vincolo Esterno

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)

Prodotti:

[V. Prodotto - Riferito Accettato]

[V. Produttore_Gestore_distro]

[V. Accettato_Voti_Dopo_Totale]

DO_Matrice

[V. Produttore_No_Pro_Div]

[V. Produttore_No_Pro_Div]

$\forall p_1, p_2, i_1, i_2, f_1, f_2, p_1.A, p_2.A$
 $\text{Produttore}(p_1) \wedge \text{Accettato}(p_1) \wedge \text{Distributore}(p_2) \wedge \text{Accettato}(p_2)$
 $\text{Terminata}(p_1, \text{FALSE}) \wedge \text{Terminata}(p_2, \text{FALSE}) \wedge \text{Inizio_Pro}(p_1, i_1) \wedge$
 $\text{Terminata}(p_1, \text{TRUE}) \wedge \text{Terminata}(p_2, \text{TRUE}) \wedge \text{Inizio_Pro}(p_2, i_2) \wedge \text{Posto_in}(p_1) \wedge \text{Posto_in}(p_2) \wedge \text{posto_per}(p_1, p_2) \wedge \text{posto_per}(p_2, p_1)$

$(i_2 > i_1 \vee i_1 > i_2)$

Risposta alla Domanda 6 (segue)
 [V. Produttore, Rifiuti] OR Accettata]

$\forall p, p', p''$
 $\text{Prototipico}(p) \wedge [\exists \text{Accett}(p), \text{isARec}(p, p')] \rightarrow [\exists \text{pr} \text{Rifiuti}(p) \wedge \text{isARec}(p, p'')]$

[V. Produttore, Utente, Dispos]

$\forall p, u$
 $\text{Prototipico}(p) \wedge (\text{Utente}(u) \wedge \text{UTprod}(u, p)) \rightarrow \text{UTdis}(u, p)$

[V. Accett], Utente, Disponibile]

$\forall p, u_1, u_2$
 $(\text{Accett}(p) \wedge \text{Termin}(p, \text{true})) \rightarrow [\exists u_1, u_2 \text{ votabili}(u_1, p) \wedge \text{votosp}(u_2, p)]$

[V. Accettato, Votabili]

$\forall p$
 $\text{Accett}(p) \wedge \text{Termin}(p, \text{true}) \rightarrow \exists v \in \text{votabili}(p, p)$

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>Azione</u> (nome)	Derivante da: <u>entità</u> relationship (cerchiare)
--	--

Attributi <u>NOME</u>							
-------------------------	--	--	--	--	--	--	--

Domini <u>STANGA</u>							
------------------------	--	--	--	--	--	--	--

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:

2 Relazione <u>CITTA</u> (nome)	Derivante da: <u>entità</u> relationship (cerchiare)
---------------------------------------	--

Attributi <u>ID</u>	<u>NOME</u>	<u>NAZIONE</u>					
-----------------------	-------------	----------------	--	--	--	--	--

Domini <u>SERIAL</u>	<u>STRANIA</u>	<u>STRANGA</u>					
------------------------	----------------	----------------	--	--	--	--	--

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: Azione references Nazionale

La relazione accorda le relazioni che implementano le seguenti relationship:

3 Relazione <u>UTENTE</u> (nome)	Derivante da: <u>entità</u> relationship (cerchiare)
--	--

Attributi <u>ID</u>	<u>NOME</u>	<u>Cognome</u>	<u>Gender</u>	<u>DATI_NATO</u>	<u>CITTÀ</u>		
-----------------------	-------------	----------------	---------------	------------------	--------------	--	--

Domini <u>SERIAL</u>	<u>STRANIA</u>	<u>STRANGA</u>	<u>Gender</u>	<u>DATI</u>	<u>Integ</u>		
------------------------	----------------	----------------	---------------	-------------	--------------	--	--

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: CITTA references CITTA

La relazione accorda le relazioni che implementano le seguenti relationship:

4 Relazione <u>WORKSPACES</u> (nome)	Derivante da: <u>entità</u> relationship (cerchiare)
--	--

Attributi <u>ID</u>	<u>DATI1</u>	<u>DATI_F</u>					
-----------------------	--------------	---------------	--	--	--	--	--

Domini <u>SERIAL</u>	<u>DATI</u>	<u>DATI_F</u>					
------------------------	-------------	---------------	--	--	--	--	--

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):

check (Cognome,DATI,F)

La relazione accorda le relazioni che implementano le seguenti relationship:

5 Relazione <u>Allegato</u> (nome)	Derivante da: <u>entità</u> relationship (cerchiare)
--	--

Attributi <u>ID</u>	<u>Dichiar1</u>	<u>Dichiar2</u>	<u>Nomefile</u>	<u>UTENTE</u>			
-----------------------	-----------------	-----------------	-----------------	---------------	--	--	--

Domini <u>SERIAL</u>	<u>Dichiar</u>	<u>Dichiar</u>	<u>Nomefile</u>	<u>Integ</u>			
------------------------	----------------	----------------	-----------------	--------------	--	--	--

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: utente references utente

La relazione accorda le relazioni che implementano le seguenti relationship:

6 Relazione .bd.All..... (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>Allegato</u> <u>Indispibilita</u>	
Domini <u>Integ</u> <u>Integ</u>	

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: Allegato references AllInfo (m)

fk: Indispibilita references Indisp (1o)

La relazione accorda le relazioni che implementano le seguenti relationship:

7 Relazione .Postalello..... (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>ID</u> <u>Tipologia</u> <u>CantInv</u> <u>Allegato</u>	
Domini <u>Serial</u> <u>Tipolog</u> <u>Real</u> <u>Integ</u>	

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: Allegato references AllInfo (1o)

La relazione accorda le relazioni che implementano le seguenti relationship: .All-.Posta

8 Relazione Prenotazione..... (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>ID</u> <u>DataI</u> <u>DataF</u> <u>NroPasseg</u> <u>UT_Pren</u> <u>UT_asp</u> ?	
Domini <u>Serial</u> <u>Date</u> <u>Date</u> <u>Integ</u> <u>Integ</u> <u>Integ</u>	

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: UT_Pren references UT_Cogn (1o) check (DataI < DataF)

fk: UT_Cogn references Utente (1o) Unique (Cogn)

Unique (Cogn)

Unique (Cogn)

La relazione accorda le relazioni che implementano le seguenti relationship: .UT_Pren, .UT_asp

9 Relazione Postalello_Pren..... (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>Postalello</u> <u>Prenotazione</u>	
Domini <u>Integ</u> <u>Integ</u>	

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: Postalello references Postalello (1o)

fk: Prenotazione references Prenotazione (1o)

La relazione accorda le relazioni che implementano le seguenti relationship:

10 Relazione Prenotazione..... (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>Prenotazione</u> <u>Motivazion</u> <u>ID</u>	
Domini <u>Integ</u> <u>Stringa</u> <u>Serial</u>	

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: Prenotazione references Prenotazione (1o)

Unique (1o)

La relazione accorda le relazioni che implementano le seguenti relationship: .LSA_Pren

11 Relazione <u>AreeVoti</u>(nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>PuntiMax</u> ID <u>Tecnologia</u> <u>VotoMax</u> * <u>VotoQsp</u> *	
Domini Integ serial Bool Voto Voto	

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 : PuntiMax VotoMax VotoQsp(n)
unique (10)

La relazione accorda le relazioni che implementano le seguenti relationship: ...IS-A Men-A.....

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	

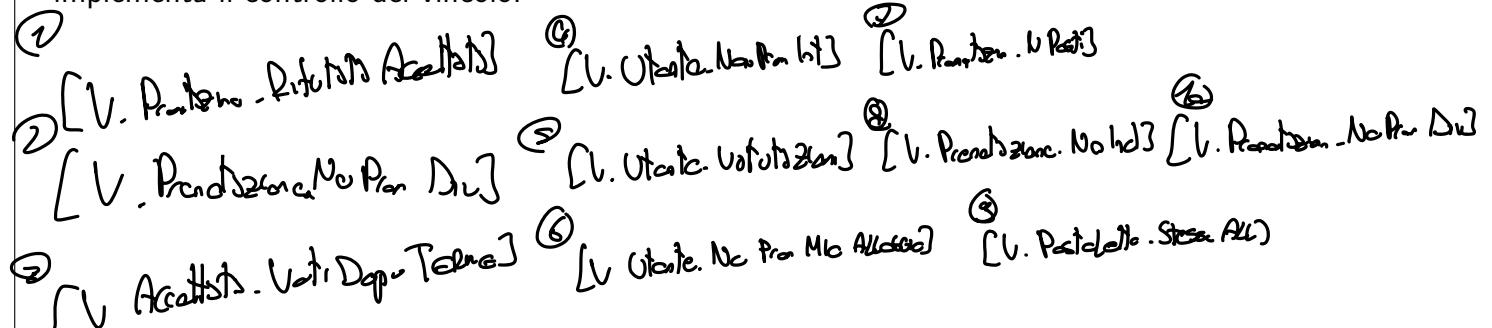
Gli attributi chiave primaria sono sottolineati, quelli i cui valori possono essere NULL sono contrassegnati con *

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 ennupla); (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.



① TRIGGER PER IL vincolo: [V_Prenotazione_Rifatti_Accettati]

OPERAZIONI: INSERIMENTO / MODIFICA SO Prenotazione, Accettati, Rifatti

ISTANTE DI INVOCAZIONE: PRIMA DELL'OPERAZIONE

Funzione:

Se IsError = FALSE

Se now la tuta che sto inserendo

IsError := exist (

SELECT *
from Prenotazione P
Join Rifatti R on P.ID = R.prenotazione
Join Accettati A on P.ID = A.prenotazione

)

Se IsError = TRUE Blocca l'operazione

ALTRIMENTI Procedi

Risposta alla Domanda 7 (segue)

[V. Pronto2loc No Prn Dv]

① TRI GOER PER IL VNUCOLO: *[V. Pronto2loc No Prn Dv]*
 OPERAZIONI: INSERIMENTO / MODIFICA SO Accettata
 ISTANTE DI VNUCAZIONE: PRIMA DELL'OPERAZIONE

Funzione:

Se $IS_ERROR = FALSE$ Si \exists NEW LA TUTTA CHE STO INSERENDO
 $IS_ERROR := \exists ($ *Select **
From Accettata A1, Accettata A2,
(Pronto2loc P JOIN PostoLetto Pn ON PLP ON PLD. Pronto2loc = P.ID
JOIN PostoLetto PL ON PLP. PostoLetto = PL.ID) P1,
(Pronto2loc P JOIN PostoLetto Pn ON PLP ON PLD. Pronto2loc = P.ID
JOIN PostoLetto PL ON PLP. PostoLetto = PL.ID) P2

where $A1.ID = NEW.ID$ AND $A1.TERMINATA = FALSE$ AND $A1.ID \neq A2.ID$ AND
 $A1.PRENOTAZIONE = P1.ID$ AND $A2.PRENOTAZIONE = P2.ID$
 $)$

Se $IS_ERROR = TRUE$ Blocca L'OPERAZIONE

ALTRIMENTI PROSEGHI

② TRI GOER PER IL VNUCOLO: *[V. Accettata, Voti Dopo Termina]*

OPERAZIONI: INSERIMENTO / MODIFICA SO Accettata
 ISTANTE DI VNUCAZIONE: PRIMA DELL'OPERAZIONE

Funzione:

Se $IS_ERROR = FALSE$ Si \exists NEW LA TUTTA CHE STO INSERENDO
 $IS_ERROR := \exists ($ *SELECT **
From Accettata A
Where A.ID = NEW.ID AND A.TERMINATA = TRUE
AND A.votilettast = NULL

)

Se $IS_ERROR = TRUE$ Blocca L'OPERAZIONE

ALTRIMENTI PROSEGHI

Risposta alla Domanda 7 (segue)

④ TRI GOER PER IL UNICO: [V. Produttore. No Pro D]

OPERAZIONI: INSERIMENTO / MODIFICA SO PRODUTTORE, ACQUAJO

ISTANTE DI INOCULAZIONE: PRIMA DELL' OPERAZIONE

Funzione:

$S_2 \text{ IS_Error} = \text{FALSE}$

$S_2 \text{ Now } \rightarrow \text{ tutti che sto ricevendo}$

$\text{IS_Error} := \text{exists} (\text{select} *$

Fran (Produttore P Jan Acquajo A1 on Ar.prontidato = P1.ID Jon Postoletto PPL on PPL.Prontidato = P1.ID)

Jan Postoletto PL1 on PL1.ID PPL.Postoletto = P1,

(Produttore P2 Jan Acquajo A2 on Ar.prontidato = P2.ID Jon Postoletto PPL on PPL.Prontidato = P2.ID)

Jan Postoletto PL2 on PL2.ID PPL.Postoletto = P2.ID AND A1.Terminata = FALSE AND A2.Terminata = FALSE

WHERE PL1.ID = P1.ID AND P1.ID = P2.ID AND PL2.ID = P2.ID AND A1.Terminata = FALSE AND A2.Terminata = FALSE

)

Se $\text{IS_Error} = \text{TRUE}$ BLOCCA L'OPERAZIONE

ALTRIMENTI PROSEGHI

⑤ TRI GOER PER IL UNICO: [V. Utente. No Pro ML Allegato]

OPERAZIONI: INSERIMENTO / MODIFICA SO UTENTE, PRODUZIONE

ISTANTE DI INOCULAZIONE: PRIMA DELL' OPERAZIONE

Funzione:

$S_2 \text{ IS_Error} = \text{FALSE}$

$S_2 \text{ Now } \rightarrow \text{ tutti che sto ricevendo}$

$\text{IS_Error} := \text{exists} (\text{select} *$

fran Utente U1, Produzione P Jan Postoletto PPL on PPL.Prontidato = P1.ID Jon Postoletto PL on

U1.UtenteID = P1.ID AND P1.UtenteID = U1.ID AND P1.Prontidato = P1.ID AND P1.ID = PPL.Prontidato = P1.ID

PL.ID = PPL.Postoletto Jan Allegato A on PL.Allegato = A.ID

PL.ID = PPL.Postoletto Jan Allegato A on PL.Allegato = A.ID AND A.UtenteID = U1.ID AND A.UtenteID = U1.ID

WHERE Now.ID = U1.ID AND P1.UtenteID = U1.ID AND A.UtenteID = U1.ID

)

Se $\text{IS_Error} = \text{TRUE}$ BLOCCA L'OPERAZIONE

ALTRIMENTI PROSEGHI

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

Registrazione (*n: String*, *c: String*, *dn: Date*, *g: Genero*): Utente

Esegui il seguente comando SQL

Insert into Utente (Nome, Cognome, DataNascita, Genero)

Value (Par_1, Par_2, Par_3, Par_4)

Dove Avete impostato Par_1 - Par_4 con i valori dei parametri attuali, rispettivamente: n, c, dn, g

CERCA_ALLEGATO (*dc:Date*, *dt:Date*, *c: STRINGA*, *np: Intero > 0*): ALLEGATO [--*]

Memorizza dentro Q il risultato della seguente query SQL:

```
SELECT *
From ALLEGATO A1 JOIN Utente U on Autore=A1.ID JOIN CITTA C on U.Città=C.ID
Where CITTA.NOME=C AND Città_Posti(A1.ID)>NP
Not Exist (SELECT *
From PROBABILIC P JOIN Personale PPL on PPL.posizione=P.ID JOIN ALLEGATO A2 on PL.Allegato=A2.ID
JOIN PostiLetto PL on PL.ID=PPL.posizione JOIN ALLEGATO A2 on PL.Allegato=A2.ID
Where A2.ID=A1.ID AND A2.Terminata=False)
```

Se Q è vuoto genera erede "Nessun Allegato"

Altrimenti ritorna Q

Risposta alla Domanda 8 (segue)

Orazione di classe Alloggio

COLLOCA_PESO(Ai Int): Int > 0

Memorizza dentro Q il risultato della se query SQL:

Select Count(*)

From Alloggio A Join Posidetto PL on PL-Alloggio = A.ID

Where A.ID = A1

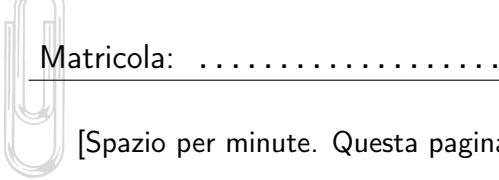
Se Q < 0

Allora ritorna Q

Genera erede "Nessun Posidetto per l'Alloggio"

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.]



Matricola:

Minute

[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.]