



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: Matricola:

Data:

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

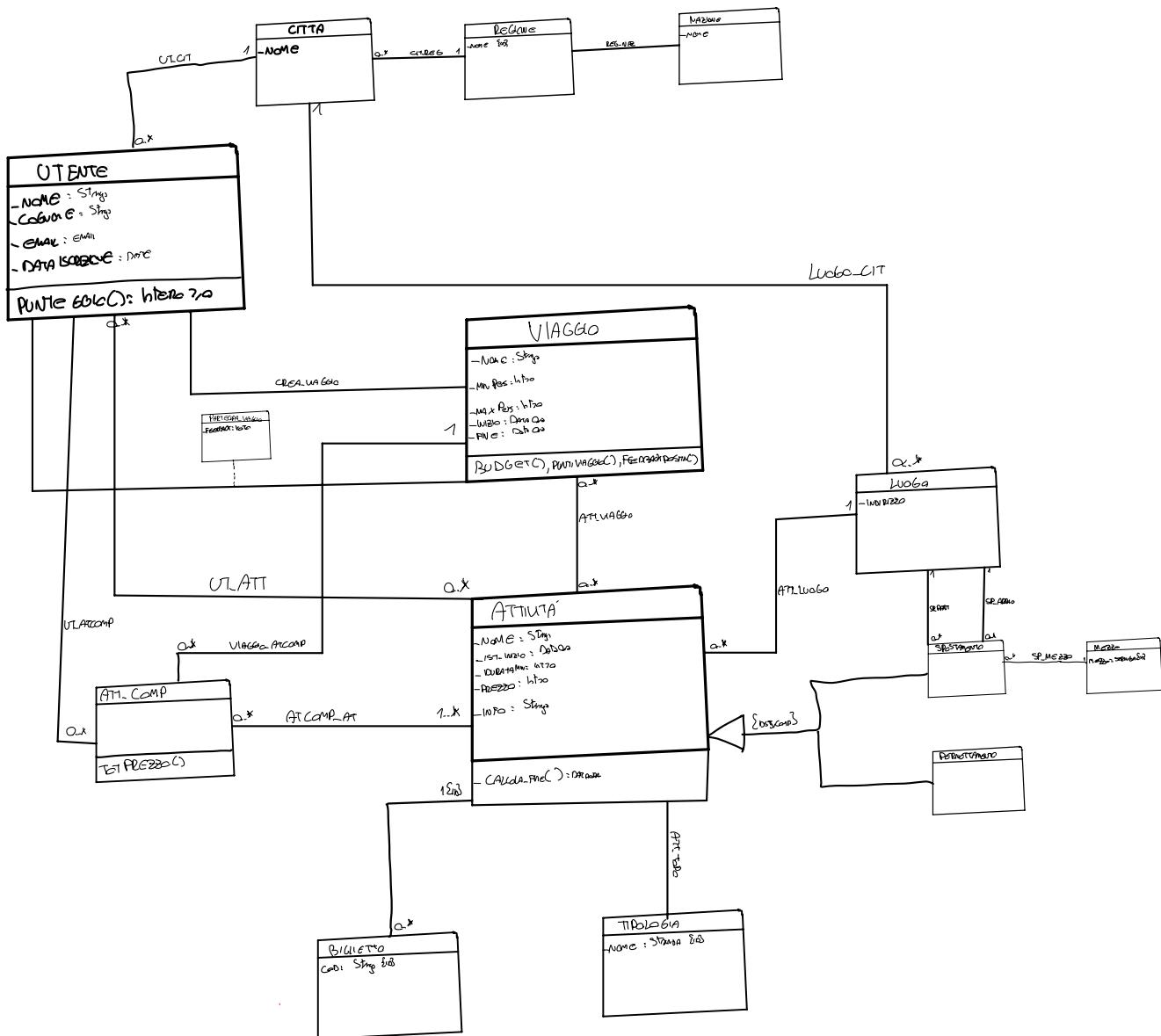
Risposta alla Domanda 1 (segue)

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>VIA G60</u>.....</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left;">attributo</th><th style="text-align: left;">dominio</th><th style="text-align: left;">moltep. (*)</th></tr> </thead> <tbody> <tr> <td></td><td></td><td></td></tr> </tbody> </table> <p>(*) solo se diversa da (1,1)</p> <p>Vincoli: $\exists p, mp, max_p$ $\exists v, ut, mp$ $\exists c, v_c, max_c$</p> $\left(\forall p, mp, max_p \right) \wedge \left(\forall v, ut, mp \right) \wedge \left(\forall c, v_c, max_c \right) \rightarrow mp = max_p \wedge mp = max_c$ $\left(\forall p, mp, max_p \right) \wedge \left(\forall v, ut, mp \right) \wedge \left(\forall c, v_c, max_c \right) \rightarrow ut = max_p \wedge ut = max_c$ $\left(\forall p, mp, max_p \right) \wedge \left(\forall v, ut, mp \right) \wedge \left(\forall c, v_c, max_c \right) \rightarrow max_p = max_c$	attributo	dominio	moltep. (*)				<p>2 Tipo: Entità Relationship (cerchiare)</p> <p>Nome: <u>VIA G60</u>.....</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left;">attributo</th><th style="text-align: left;">dominio</th><th style="text-align: left;">moltep. (*)</th></tr> </thead> <tbody> <tr> <td></td><td></td><td></td></tr> </tbody> </table> <p>(*) solo se diversa da (1,1)</p> <p>Vincoli: $\exists v, ut, mp$ $\exists c, v_c, max_c$</p> $\left(\forall v, ut, mp \right) \wedge \left(\forall c, v_c, max_c \right) \rightarrow ut = max_c$ $\left(\forall v, ut, mp \right) \wedge \left(\forall c, v_c, max_c \right) \rightarrow max_c = max_p$ $\left(\forall v, ut, mp \right) \wedge \left(\forall c, v_c, max_c \right) \rightarrow max_p = max_c$	attributo	dominio	moltep. (*)			
attributo	dominio	moltep. (*)											
attributo	dominio	moltep. (*)											

<p>3 Tipo: Entità Relationship (cerchiare)</p> <p>Nome: <i>VIA GLO</i></p> <table border="1"> <thead> <tr> <th>attributo</th> <th>dominio</th> <th>moltep. (*)</th> </tr> </thead> </table>	attributo	dominio	moltep. (*)	<p>5 Tipo: Entità Relationship (cerchiare)</p> <p>Nome: <i>VIA GLO</i></p> <table border="1"> <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><i>[V. VIA GLO. INZOPEN]</i></p> <p>$\forall v_1, v_2 [V_{GLO}(v_1) \wedge V_{GLO}(v_2) \wedge \text{INZOPEN}(v_1, v_2)] \rightarrow \text{IEP}$</p>	<p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p> <p><i>[V. VIA GLO. ATTIVITA]</i></p> <p>$\forall v_1, a_1, v_2, v_3, v_4 [V_{GLO}(v_1) \wedge \text{ATTIVITA}(a_1) \wedge \text{APT_VIA GLO}(a_1, v_1) \wedge \text{INZOPEN}(v_1, v_2) \wedge \text{INZOPEN}(v_1, v_3) \wedge \text{INZOPEN}(v_2, v_3) \wedge \text{ATTIVITA}(v_2) \wedge \text{ATTIVITA}(a_2) \wedge \text{APT_VIA GLO}(a_2, v_2) \wedge \text{INZOPEN}(v_3, v_4) \wedge \text{INZOPEN}(v_2, v_4) \wedge \text{INZOPEN}(v_3, v_4) \wedge \text{ISCRIZIONE}(a_1, a_2)] \rightarrow \text{VIALE_FV}$</p>						

<p>4 Tipo: Entità Relationship (cerchiare)</p> <p>Nome: <i>UTENTE</i></p> <table border="1"> <thead> <tr> <th>attributo</th> <th>dominio</th> <th>moltep. (*)</th> </tr> </thead> </table>	attributo	dominio	moltep. (*)	<p>6 Tipo: Entità Relationship (cerchiare)</p> <p>Nome: <i>ATTIVITA</i></p> <table border="1"> <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><i>[V. UTENTE. NO_BORN_CREATE]</i></p> <p>$\forall u_1, u_2 [U_{TEN}(u_1) \wedge U_{TEN}(u_2) \wedge \text{NO_BORN_CREATE}(u_1, u_2)] \rightarrow \text{PARTICIPAZIONE}(u_1, u_2)$</p> <p>$\sum U_{TEN}(u) \wedge V_{GLO}(u) \wedge \text{CREAZIONE}(u) \rightarrow \text{PARTICIPAZIONE}(u, u)$</p>	<p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p> <p><i>[V. ATTIVITA. UTENTE]</i></p> <p>UN UTENTE NON PUÒ CREA UNA ATTIVITÀ CON DATA MIGLIORE DELLA SUA ISCRIZIONE</p> <p>$\forall a_1, u_1, d_1, i_1, v_1 [U_{TEN}(u_1) \wedge V_{GLO}(a_1) \wedge \text{CREAZIONE}(a_1, u_1) \wedge \text{ATTIVITA}(a_1) \wedge \text{UTENTE}(u_1) \wedge \text{VIALE}(v_1) \wedge \text{CREAZIONE}(u_1, v_1) \wedge \text{ATTIVITA}(a_1, v_1) \wedge [\text{ATTIVITA}(a_1) \wedge \text{UTENTE}(u_1) \wedge \text{VIALE}(v_1) \wedge \text{CREAZIONE}(u_1, v_1) \wedge \text{ATTIVITA}(a_1)] \rightarrow \text{DATA_ISCRIZIONE}(u_1, v_1) \wedge \text{ISCRIZIONE}(a_1, v_1)] \rightarrow \text{DO_LA}$</p> <p>UGUALE PER UTENTE PARTECIPA E STESSA COSA PER VIALE (SIA CREAZIONE CHE PARTI) + 3V</p>						

7 Tipo: Entità | Relationship (cerchiare)

Nome: ...SPOSTAMENTO.....

attributo dominio moltep. (*)

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

Vincoli:

[U. SPOSTAMENTO. NO PARTIRE]

$$\forall s, l$$

$$[\text{Spostamento}(s) \wedge \text{Locato}(l) \wedge \text{Sp_Part}(s, l)] \rightarrow \exists \text{Salvo}(s)$$

9 Tipo: Entità | Relationship (cerchiare)

Nome: ...ATTI CITA.....

attributo dominio moltep. (*)

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

Vincoli:

[U. ATTILA. NO SM]

$$\forall A_1, A_2, IA_1, IA_2, F_1, F_2, V$$

$$[\text{ATTILA}(A_1) \wedge \text{ISINIZIO}(A_1, IA_1) \wedge \text{CALC_FINE}(A_1, F_1) \wedge \text{VIA_GEO}(V) \wedge \text{ATTI_VIAGGIO}(A_1, V) \wedge \text{ATTI_UTRA}(A_2) \wedge \text{INSERITO}(A_2, IA_2) \wedge \text{CALC_FINE}(A_2, F_2) \wedge \text{ATTI_VIAGGIO}(A_2, V) \wedge A_1 \neq A_2] \rightarrow (F_1 \neq F_2 \vee F_2 \neq F_1)$$

8 Tipo: Entità | Relationship (cerchiare)

Nome: ...PERCORSAMENTO.....

attributo dominio moltep. (*)

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

Vincoli:

[U. PERCORSAMENTO. NO 2 PERU GOLUB]

$$\forall p_1, p_2, v, l_1, l_2, g_1, g_2$$

$$[\text{Percorsamento}(p_1) \wedge \text{PERU_GOLUB}(p_1) \wedge \text{UTENZE}(v) \wedge \text{VZ_ATT}(v, p_1) \wedge \text{GEO}(v, g_1) \wedge \text{Percorsamento}(p_2) \wedge \text{PERU_GOLUB}(p_2) \wedge \text{UTENZE}(l_2) \wedge \text{VZ_ATT}(l_2, p_2) \wedge \text{GEO}(l_2, g_2)] \rightarrow (\neg g_1 = g_2)$$

10 Tipo: Entità | Relationship (cerchiare)

Nome: ...ATTComp.....

attributo dominio moltep. (*)

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

Vincoli:

[U. ATTComp. ATT IN VIAGGIO]

$$\forall A, AC, V$$

$$[\text{ATTComp}(AC) \wedge \text{ATTILA}(A) \wedge \text{ATTComp_ATT}(AC, A) \wedge \text{VIA_GEO}(V) \wedge \text{ATTComp}(AC) \wedge \text{ATTILA}(A) \wedge \text{ATTComp_ATT}(AC, A) \wedge \text{VIA_GEO}(V) \wedge \text{ATT_VIAGGIO}(A, V) \wedge \text{VIA_GEO_ATTComp}(V, AC) \rightarrow \text{ATT_VIAGGIO}(A, V)]$$

11 Tipo: Entità Relationship (cerchiare)	13 Tipo: Entità Relationship (cerchiare)
Nome: PARTECIPA.....	Nome: URGENTI.....
attributo	attributo
dominio	dominio
moltep. (*)	moltep. (*)
(*) solo se diversa da (1,1)	(*) solo se diversa da (1,1)
Vincoli:	Vincoli:
$\forall U_1 V_1 A_1 I_1 F_1 P_1$ $(U_1 \cap V_1 \cap A_1 \cap I_1 \cap F_1 \cap P_1)$ $(U_1 \cap V_1 \cap A_1 \cap I_1 \cap F_1 \cap P_1) \wedge$ $(U_1 \cap V_1 \cap A_1 \cap I_1 \cap F_1 \cap P_1) \rightarrow \exists F_2$ $F_2 = \text{FEEDBACK}(P_1, F_1)$ $\neg A_1 \wedge \neg I_1 \wedge \neg F_1 \wedge \neg P_1$	
	IN TUTTO 16 VINCOLI

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

14	Tipo: Entità Relationship (cerchiare)		
Nome:			
attributo	dominio	moltep. (*)	
<hr/>			
(*) 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

OPERAZIONE CLASSE ATTRUTTA'

- CALCOLA_FINE(A: ATTRUTTA'): DATA ORA

PRE = Nessuna

POST: Non modifica il livello est. $M_{in} = M_{out}$

Nessuna TUTTA AGGIUNTA C REMOSSA

$$\text{RESULT} = \left(\sum_{\text{ATT}} \left| \begin{array}{l} \text{ATTRUTTA}(\text{this}) \wedge \text{LISTA_DURATA}(\text{this}, \text{A}) \\ \wedge \text{DURATA}(\text{this}, \text{D}) \rightarrow \text{F} = \text{IATD} \end{array} \right| \right)$$

OPERAZIONE CLASSE ATTCOMP

- tot_PREZZO(): REALE

- PRE = Nessuna

- POST: $M_{in} = M_{out}$
 $\hookrightarrow_{\text{A}} \text{ATT} = \left\{ (\text{A}, \text{P}) \mid \text{ATTRUTTA}(\text{A}) \wedge \text{ATTCOMP_AT}(\text{this}, \text{A}) \wedge \text{PREZZO}(\text{A}, \text{P}) \right\}$

$$\text{RESULT} = \sum_{\text{ATT}} \text{P}$$

OPERAZIONE CLASSE VIAGGIO

- BUDGET(): REALE

- PRE = Nessuna

- POST: $M_{in} = M_{out}$

$\hookrightarrow_{\text{A}} \text{ATT} = \left\{ (\text{A}, \text{P}) \mid \text{ATTRUTTA}(\text{A}) \wedge \text{ATT_VIAGGIO}(\text{A}, \text{this}) \wedge \text{PREZZO}(\text{A}, \text{P}) \right\}$

$$\hookrightarrow_{\text{A}} \text{TOT_AT} = \sum_{\text{ATT}} \text{P}$$

$\hookrightarrow_{\text{AC}} \text{ATT_COMP} = \left\{ (\text{AC}) \mid \text{ATTCOMP}(\text{AC}) \wedge \text{VIAGGIO_ATCOMP}(\text{this}, \text{AC}) \right\}$

$$\hookrightarrow_{\text{AC}} \text{TOT_AC} = \sum_{\text{AC} \in \text{ATT_COMP}} \text{tot_PREZZO}(\text{AC})$$

$$\text{RESULT} = \text{TOT_AT} + \text{TOT_AC}$$

Risposta alla Domanda 2 (segue)**PUNTI VIA GIO(C) = REALE**

DSE: $\exists v \in \text{UTENTE}(v) \wedge \text{PARTICIP-VAGGIO}(v, \text{this}) \wedge \text{FINE}(\text{this}, f) \wedge \text{ADDESSO}(f) \rightarrow f \in A$

Post: $M_{in} = M_{out}$

$$S_A \text{ VotiVAGGIO} = \left\{ (f) \mid \text{UTENTE}(v) \wedge \text{PARTICIP-VAGGIO}(v, \text{this}, p_v) \wedge \text{feedback}(p_v, f) \right\}$$

$$\text{Result} = \sum_{f \in \text{VotiVAGGIO}} f$$

FEEDBACK PESRIV(C) = INTENSITÀ

PES: Nessuna

Post: $M_{in} = M_{out}$

$$S_A \text{ TOT PES} = \left\{ (f) \mid \text{UTENTE}(v) \wedge \text{PARTICIP-VAGGIO}(v, \text{this}, p_v) \wedge \text{feedback}(p_v, f) \wedge f \geq 4 \right\}$$

$$\text{Result} = |\text{TOT PES}|$$

OPERAZIONE CLASSE UTENTE**PUNTI GIO(C) = REALE**

$$PNC = \exists v \wedge \text{CREA-VAGGIO}(\text{this}, v)$$

Post: $M_{in} = M_{out}$

$$S_A \text{ VAGGIO} = \left\{ (v) \mid \text{VAGGIO}(v) \wedge \text{CREA-VAGGIO}(\text{this}, v) \right\}$$

$$S_A \text{ TOT F} = \sum_{v \in \text{VAGGIO}} \text{PUNTI VIA GIO}(v)$$

$$S_A P = \frac{\text{TOT F}}{|\text{VAGGIO}|}$$

$$S_A \text{ TOT PES} = \sum_{v \in \text{VAGGIO}} \text{Feedback PES}(v)$$

$$\text{Result} = \begin{cases} \text{SE } P \leq 3 \\ \text{ALTRIMENTI } \text{Result} \in [\text{TOT PES}] \end{cases}$$

TIPI DI DATI

- VOL = WORD [o...s]

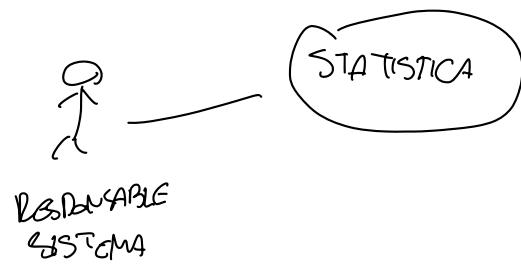
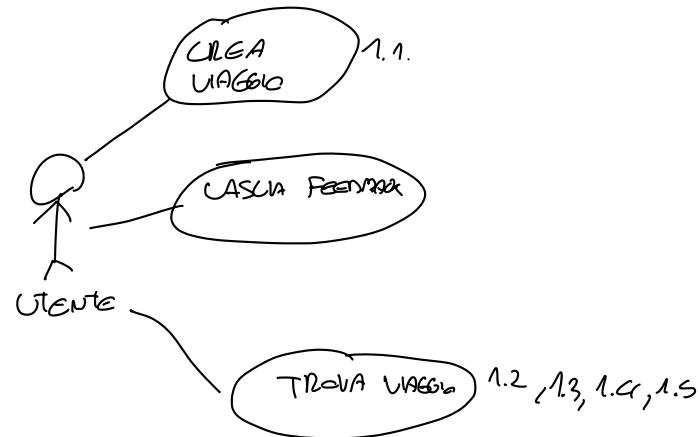
- EMAIL = SPRNG SECONDO REGEX

- INPI REZ = { VIA: SPRNG, ALTRUO: Intenz, CNP: CAP }

- CAP = SPRNG SECONDO REGEX

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: CERCA_VIAGGIO (nome use-case)

Operazioni dello use-case:

CERCA_VIAGGIO(A: Città \rightarrow {I: DATA, F: DATA, N: STRING, MIN: INTERO, MAX: INTERO}) : VIAGGIO

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

Operazioni dello use-case:

FEEDBACK(v: VIAGGIO; v: Voto) : REALE

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

Operazioni dello use-case:

- RICERCA_SEMPLE(w: DATA, f: DATA, c: STRINGA) : [VIAGGIO]{0..}

- CITTA_VIAGGIO(w: DATA, f: DATA) : CITTA

- VIAGGI_REG(r: STRINGA, w: DATA, f: DATA) : INTERO

- RICERCA_COMPLETA(b_min: REALE, b_max: REALE, r: {Espresso, L'Espresso, L'Espresso, L'Espresso}, w: DATA, f: DATA, p: REALE) : [VIAGGIO]{0..}

4 Specifica use-case: STATISTICA (nome use-case)

Operazioni dello use-case:

- STATISTICA_TOT (C: STRANGA) [INTERO] {0..18}
- STATISTICA_MES (C: STRANGA, M: INT 1..12): Inter

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

Operazioni dello use-case:

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

Operazioni dello use-case:

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 **SADANNO IMPLEMENTAZIONI SOLO GLI USE CASE PIÙ COMPLESSI**

- RICERCA COMPLETA ($P_{MIN}: REALE, P_{MAX}: REALE, R = \{Legge, a, b\}, W: DATA, F_W: DATA, P: REALE\} \models [VAGGIO]_{[a..b]}$)

PRE CONDIZIONI: $B_{MIN} \leq B_{MAX} \wedge W \leq F_W$

POST: $\forall V \models \left| \begin{array}{l} VAGGIO(V) \wedge B_{MIN} \leq B \leq B_{MAX} \wedge \text{INIZIO}(V, V) \wedge \text{FINE}(V, F_W) \\ IN \leq V \leq F_W \wedge UTENTE(V) \wedge \text{CREA_VIAGGIO}(V, V) \wedge \text{PUNTEGGIO}(V, P) \\ \exists A, LA, CIT \text{ ATTURA}(A) \wedge \text{ATTI_VIAGGIO}(V, A) \wedge \text{LUGO}(A) \wedge \text{ATTI_LUGO}(A, LA) \\ \text{CITA}(C) \wedge \text{LUGO_CIT}(C, A) \wedge \text{REGOLE}(C) \wedge \text{CIT_REG}(C, P) \wedge P \in R \end{array} \right\}$

$$\text{RESULT} = V$$

- STATISTICA_MESI ($C: STREGA, M: INT 1..12$): $Intero$

PRE: NESSUNA

POST: SIA $ADDESSO(A) \wedge ANB(A, AN)$

$$\text{SIA } MIBO = AN - 1$$

$\exists A, LA, CIT \models \left| \begin{array}{l} VAGGIO(V) \wedge \text{INIZIO}(V, V) \wedge \text{IV2INIZIO} \wedge \text{MESI}(V, MES) \wedge MES = M \\ \text{ATTURA}(A) \wedge \text{ATTI_VIAGGIO}(A, V) \wedge \text{LUGO}(A) \wedge \text{ATTI_LUGO}(A, LA) \wedge \text{LUGO_CIT}(LA, CIT) \wedge CIT = C \end{array} \right\}$

$$\text{RESULT} = 12$$

- STATISTICA_TOT ($C: STREGA$): $[1..12, Intero]$

PRE: NESSUNA

POST: $\exists A \models Q = \left\{ \left(m, n \right) \middle| \text{Intero}(m) \wedge 1 \leq m \leq 12 \wedge \text{STATISTICA_MESI}(C, m, n) \right\}$

$$\text{RESULT} = Q$$

Risposta alla Domanda 5 (segue)

VIA GLI-REG ($R = \text{STRN6, IN:DATA, FIN:DATA}$) \cap INTER>0

PRE: $IN \leq FN$

POST: $SIA R_2 \left\{ \begin{array}{l} \text{VIA GLI}(U) \wedge [IN, FN \in \text{INT}(U, IN) \wedge \text{PRE}(U, FN) \wedge IN \leq IN \leq FN] \wedge \\ \wedge \exists A, L_C, REG \text{ APPLICAB}(A, IN) \wedge \text{LOC}(C, A) \wedge \text{ATT-MOD}(A, IN) \wedge \text{CHTA}(C) \wedge \text{LUCER-OR}(C, IN) \\ \wedge \text{REGIONE}(C, IN) \wedge \text{ULREG}(C, REG) \wedge REG = R_2 \end{array} \right\}$

RESULT = $|R|$

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.

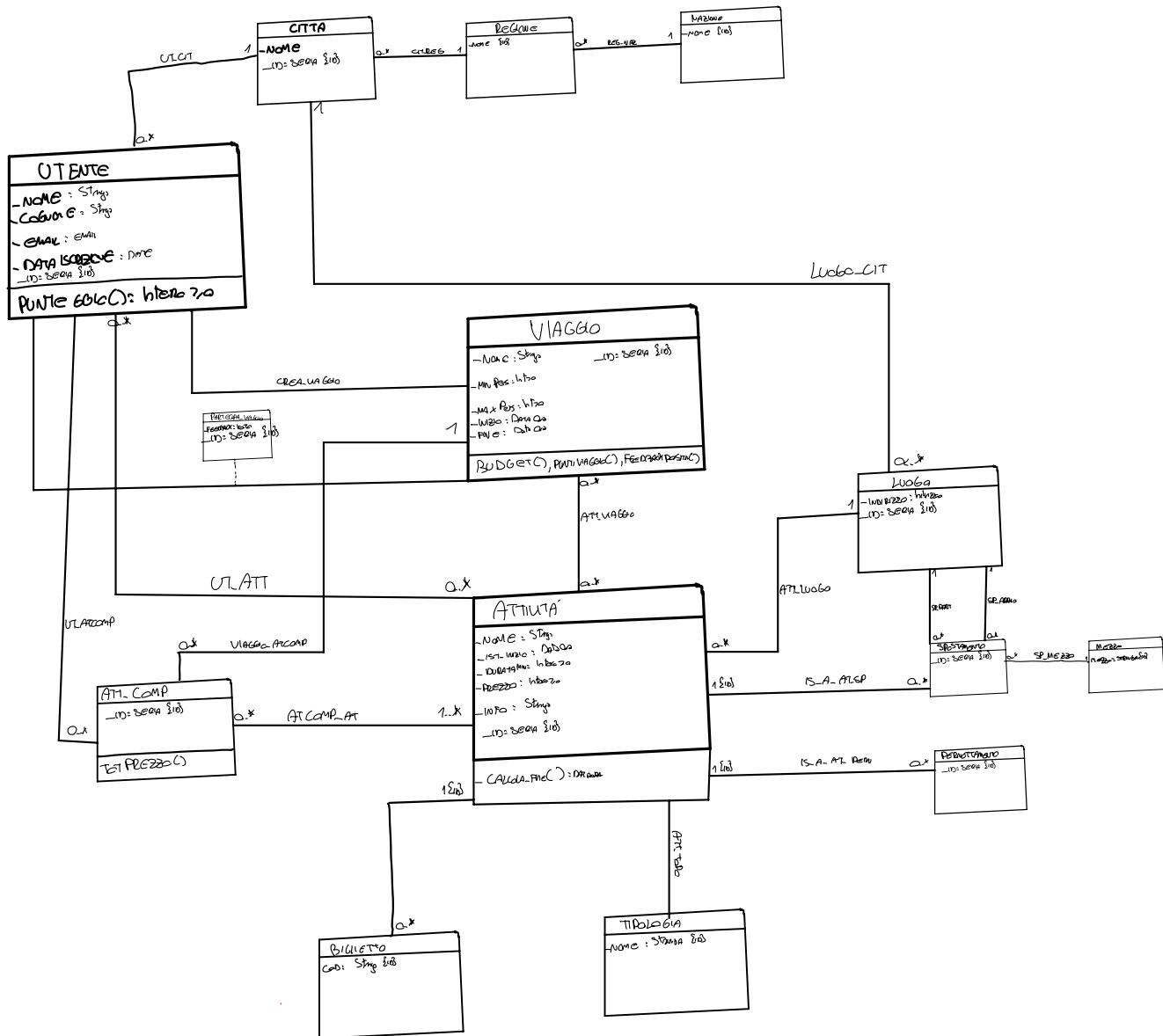
Descrivere brevemente le principali scelte effettuate.

DBMS da utilizzare Post SQL

Corrispondenza tra domini concettuali e domini supportati dal DBMS

Crete Domn	String	as	varchar check value not null
Crete Domn	EMAL	as	varchar (dato value not null and value as regex)
Crete Domn	Interzo	as	integer check value not null and value > 0
Crete Domn	Interzo	as	integer check value not null and value >= 0
Crete Domn	Voto	as	integer (check value not null and value <= 100)
Crete type	Indirizzo	=	{ vedi: String, tecnico = Interzo, cap = Interzo }

Diagramma ER ristrutturato



Breve descrizione delle scelte effettuate durante la ristrutturazione

Ho diuso la generalizzazione in associazione B-A aggiungendo quindi un ulteriore 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)

Nc001:

[V. Spostamento. Nc Perù]

$\forall s, p, A$
 $Spostamento(s) \wedge Attivita(A) \wedge is_A_AT_SP(A, s) \wedge Pernottamento(p) \rightarrow \exists is_A_AT_PER(A, p)$

[V. Pernottamento. No Sport]

$\forall s, p, A$
 $Pernottamento(p) \wedge Attivita(A) \wedge is_A_AT_PER(A, p) \wedge Spostamento(s) \rightarrow \exists is_A_AT_SP(A, s)$

Modifica:

Nessun Vincolo da Modificare

Risposta alla Domanda 6 (segue)

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 NAZIONE..... (nome)	Derivante da: entità relationship (cerchiare)
--	--

Attributi	<u>Nome</u>							
-----------	-------------	--	--	--	--	--	--	--

Domini	<u>STRNGA</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 REGIONE..... (nome)	Derivante da: entità relationship (cerchiare)
--	--

Attributi	<u>Nome</u>	<u>NAZIONE</u>						
-----------	-------------	----------------	--	--	--	--	--	--

Domini	<u>STRNGA</u>	<u>STRNGA</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: NAZIONE detraes NAZIONE

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

3 Relazione CITTÀ..... (nome)	Derivante da: entità relationship (cerchiare)
--------------------------------------	--

Attributi	<u>ID</u>	<u>Regione</u>						
-----------	-----------	----------------	--	--	--	--	--	--

Domini	<u>SEGNAL</u>	<u>STRNGA</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: Regione detraes Regione (None)

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

4 Relazione UTENTE..... (nome)	Derivante da: entità relationship (cerchiare)
---------------------------------------	--

Attributi	<u>ID</u>	<u>Nome</u>	<u>Cognome</u>	<u>Email</u>	<u>DataNasc</u>	<u>Città</u>		
-----------	-----------	-------------	----------------	--------------	-----------------	--------------	--	--

Domini	<u>SEGNAL</u>	<u>STRNGA</u>	<u>STRNGA</u>	<u>EMAL</u>	<u>DATA</u>	<u>Integer</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: Città detraes Città (None)

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

5 Relazione VIAGGIO..... (nome)	Derivante da: entità relationship (cerchiare)
--	--

Attributi	<u>ID</u>	<u>Nome</u>	<u>Mn P.</u>	<u>Max P.</u>	<u>WizB6</u>	<u>Fare</u>	<u>Creare</u>	
-----------	-----------	-------------	--------------	---------------	--------------	-------------	---------------	--

Domini	<u>SEGNAL</u>	<u>STRNGA</u>	<u>Intero</u>	<u>Intero</u>	<u>Data</u>	<u>Data</u>	<u>Boolean</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: Creare detraes Utente (None)
check (MnP <= MaxP AND WizB6 >= Fare)

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

6 Relazione <u>PARTECIPA_VIA6</u> (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>UTENTE</u> <u>VIAGGIO</u> <u>FEEDBACK*</u> <u>ID</u>	
Domini <u>INTERGEN</u> <u>INTERGEN</u> <u>VOTO</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: UTENTE references ORGANIZZAZIONE (no)
fk: VIAGGIO references VIAGGIO (no)

La relazione accorda le relazioni che implementano le seguenti relationship:

7 Relazione <u>ATTIVITA7</u> (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>ID</u> <u>NOME</u> <u>ISTITUTO</u> <u>DIPARTIMENTO</u> <u>PROFESSORE</u> <u>VIA</u> <u>VIAGGIO</u> <u>LUGO</u> <u>TIPICO</u>	
Domini <u>SERIAL</u> <u>STRADA</u> <u>DATETIME</u> <u>INTERNA</u> <u>INTERNA</u> <u>SINGOLO</u> <u>INTERNA</u> <u>INTERNA</u> <u>STRADA</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: VIAGGIO references VIAGGIO (no) fk: TIPICO references TIPOLOGIA (no)
fk: LUGO references LUGO (no)

La relazione accorda le relazioni che implementano le seguenti relationship: ATTIVITA_VIAGGIO, ATTIVITA_LUGO, ATTIVITA_TIPICO

8 Relazione <u>OT.ATI</u> (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>UTENTE</u> <u>ATTIVITA</u>	
Domini <u>SERIAL</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: UTENTE references UTENTE (no)
ATTIVITA references ATTIVITA (no)

La relazione accorda le relazioni che implementano le seguenti relationship:

9 Relazione <u>ATT.Comp</u> (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>ID</u> <u>VIAGGIO</u>	
Domini <u>SERIAL</u> <u>INTERNA</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: VIAGGIO references VIAGGIO (no)

La relazione accorda le relazioni che implementano le seguenti relationship: VIA_GOK_AT.Comp

10 Relazione <u>AT.ATTComp</u> (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>ATT.Comp</u> <u>ATTIVITA</u>	
Domini <u>INTERGEN</u> <u>INTERGEN</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: ATT.Comp references ATT.Comp (no)

fk: ATTIVITA references ATTIVITA (no)

La relazione accorda le relazioni che implementano le seguenti relationship:

11 Relazione .UT_ATCComp.. (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>Utente</u> <u>ATCComp</u>	
Domini <u>Integr</u> <u>Integr</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 (1..)
fk: ATCComp references ATCComp (1..)

La relazione accorda le relazioni che implementano le seguenti relationship:

12 Relazione .Ludo..... (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>ID</u> <u>LUDOGO</u> <u>CITTÀ</u>	
Domini <u>Serie</u> <u>Integr</u> <u>Serie</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: CITTÀ references CITTÀ (1..)

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

13 Relazione .BIGLIETTe..... (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>Attività</u> <u>CON</u>	
Domini <u>Integr</u> <u>SINGA</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: Attività references ATTIVITA (1..)

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

14 Relazione .TIPDOLGA.... (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>Nona</u>	
Domini <u>STANZA</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:

15 Relazione .SPSSTAMENTO... (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>Id</u> <u>LPART</u> <u>LAPP</u> <u>Mozzo</u> <u>ATTIVITA</u>	
Domini <u>Serie</u> <u>Integr</u> <u>Integr</u> <u>SERIE</u> <u>Integr</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):

chk (LPART ≠ LAPP) fk: Mozzo references Mozzo (none)

fk: partita references ATTIVITA (1..)

La relazione accorda le relazioni che implementano le seguenti relationship: ...SP_PART, ...SP_APP, SP_Mozzo

(S_A_M_SP)

16 Relazione .MC22C.....(nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>Nmc</u>	
Domini <u>stamp</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):

17 Relazione .PERNOTTAMENTO(nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>ID</u>	
Domini <u>SEZIONE</u> <u>Integro</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: ATTINTA references ATTINTA (ID)

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):

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):

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:

*SARANNO IMPLEMENTATI SOLO 3 VINCI ESSENDO POCO QUESTO
TEMPORALI (VA PIÙ DI 3 VINCI ESSENDO FATTIBILE)*

[continua a pagina delle minute]

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 ennupple); (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) [U. PARTECIPAZIONE GOL. FEEDBACK]

2) [U. ATTCOMP. AT IN VIAGGIO]

3) [U. ATTURA. NO SIM]

1) T. PARTECIPAZIONE VIAGGIO FEEDBACK

- INSERIMENTO: PARTECIPAZIONE VIAGGIO

- PREOPERAZIONE:
IS VALID := (NOT EXISTS (

```
SELECT *
FROM PARTECIPAZIONE PV
JOIN UTENTE U ON U.ID = PV.UTENTE
JOIN VIAGGIO V ON V.ID = PV.VIAGGIO
WHERE MEET.ID = PV.ID AND V.FINE > NOW() AND
PV.FEEDBACK IS NOT NULL
```

)

if IS VALID:
COMMIT

else: GENERA ERRORE
ROLLBACK

2) T. ATTCOMP. AT IN VIAGGIO

- INSERIMENTO: ATTCOMP

- PREOPERAZIONE:
IS VALID := (NOT EXISTS (

```
SELECT *
FROM ATTCOMP AC
JOIN VIAGGIO V ON V.ID = AC.VIAGGIO
JOIN ATTCOMP ATAC ON ATAC.ATTCOMP = AC.ID
JOIN ATTURA A ON ATAC.ATTURA = A.ID
WHERE AC.VIAGGIO + A.VIAGGIO <= NOW().ID: AC.ID
```

)

if IS VALID:
COMMIT

else: GENERA ERRORE
ROLLBACK

Risposta alla Domanda 7 (segue)

T. ATTIVITA.NB SIM

INSCERIMENTO = ATTIVITA

PES OPERAZIONE:

$i \in V_{ATTIVITA} = \{ \text{not exist} \mid \text{select } \# \}$

From ATTIVITA A₁, ATTIVITA A₂, VIAGGIO V

where A₁.VIAGGIO = V.ID AND A₂.VIAGGIO = V.ID

AND A₁.ID ≠ A₂.ID AND A₁.ISTITUTO = A₂.ISTITUTO

)

If i ∈ V_{ATTIVITA}

Comit

else
genere eccezione
Riavvia

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

SANAVO SCRITTO TUTTE LE OPERAZIONI DI CLASSE E SOLAMENTE 2 OPERAZIONI

OPERAZIONI DI CLASSE

- ATIVITA'

- CALCO_UA_FINE (A:integer): DATAFOLIA
create function

```
Q=( SELECT A.WORK + WORKLOAD A.ATIVITAMIN:INTERVAL
    FROM ATIVITA A
    WHERE A.ID = A )
```

RESULT = Q

- ATT COMP

- PREZZO_TOTALE (AC:integer):REALC

CREATE FUNCTION

```
Q=( SELECT sum(A.PREZZO)
    FROM ATTCOMP AC JOIN ATTCOMP ATC ON MAC.ATTCOMP=AC.AC
    JOIN ATIVITA A ON MAC.ATIVITA=A.ID
    WHERE AC.AC=ATC )
```

RESULT = Q

- VIA GLO

- PUNTO_GLO (V1:integer): REALC

```
WHILE NOT ATCOMP = (SELECT sum(AC.PREZZO_TOTALE)
    FROM ATTCOMP AC JOIN VIAGGIO V ON AC.VIAGGIO=V.AC
    WHERE V.ID=V1)
```

```
WHILE NOT AT = (SELECT sum(A.PREZZO)
    FROM ATTIVITA A JOIN VIAGGIO V ON V.ID=A.VIAGGIO
    WHERE V.ID=V1)
```

Q = AT+ATCOMP

RESULT = Q

- PONTI_VIAGGIO (V1:integer): REALC

create func.

```
Q=( SELECT sum(PV.FEEBACK)
    FROM PUNTO_VIAGGIO PV JOIN VIAGGIO V ON PV.VIAGGIO=V.ID
    WHERE V.FINE>NOW() AND V.ID=V1)
```

RESULT = Q

Risposta alla Domanda 8 (segue)

- FEEDBACK POSITIVI (U.ID INTEG) = Intero

Credit Function

$Q = (\text{SELECT COUNT}(PU.FEEDBACK})$
 FROM PAPPERA VIAGGIO PV JOIN VIAGGIO V ON PV.VIAGGIO = V.ID
 WHERE V.FINE > NOW() AND V.ID = U.ID AND PU.FEEDBACK > 0)

RESULT = Q

UTENTE

- PUNTI GES (U.ID INTEG) = Reale

whit NViaggi := (SELECT COUNT(U.ID))
 FROM UTENTE U JOIN VIAGGIO V ON U.CREAZIONE = V.ID
 WHERE U.ID = U)

whit P := (SELECT SUM(PUNTI_VIAGGIO(V.ID))/NViaggi)

FROM UTENTE U JOIN VIAGGIO V ON U.ID = V.ID
 WHERE U.ID = U)

whit Positivi := (SELECT COUNT(FEEDBACK_POSITIVI(U.ID)) * 0.1)

FROM UTENTE U JOIN VIAGGIO V ON U.CREAZIONE = V.ID
 WHERE U.ID = U)

if P < 3

RESULT = 0

else RESULT = P +

USE CASE

- RICERCA COMPLETA (BMIN = Reale, BMAX = Reale, R = Intero [Integ], UDato, FN = Data, P = Reale)

MEMORIZZATA IN Q IL risultato della seguente query SQL

(SELECT U.ID
 FROM VIAGGIO V, ATTIVITA A, LUOGO L, CITTA C, REGGIO R, UTENTE U
 WHERE BMIN <= BUDGET(V.ID) <= BMAX AND
 V.FINE > FN AND V.PUNTO < P AND
 A.VIAGGIO = V.ID AND A.LUOGO = L.ID AND L.CITTA = C.ID AND C.REGGIO = R.NOME
 AND R.ID = R
 AND U.CREAZIONE = V.ID AND PUNTI_GES(U.ID) > P)
 AND V.CREAZIONE = U.ID AND PUNTI_GES(U.ID) > P)

so Q ≠ NULL => DICTANA Q

ALGORITMI GENERALI ERRORE

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.]

STATISTICA - 27 (c: MEGO)

MEMORIA: NEL Q IL RISULTATO DELLA SEGUENTE QUERY SERVIRÀ

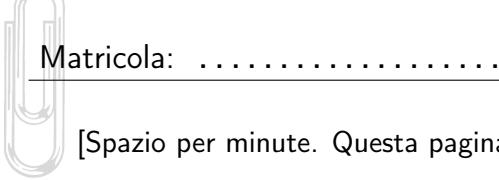
SELECT EXTRACT(MONTH FROM V.ID), COUNT(DISTINCT V.ID)
 FROM VIAGGIO V, ATTIVITA A, LUGO L, CITTÀ C
 WHERE A.VIAGGIO = V.ID AND A.LUGO = L.ID AND L.CITTÀ = C.ID
 AND C.ID = C AND (EXTRACT(YEAR FROM V.ID)) = EXTRACT(YEAR FROM NOW)

GRUPPARE PER EXTRACT(MONTH FROM V.ID)

SE Q ≠ NULL → RECORDA Q

ALTRIMENTI GENERA ERRORE

[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.]