



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 20220608 (P.20220608)

TravelPlan

Versione 2022-05-25

2

Specifiche dei Requisiti

Gli utenti di *TravelPlan* utilizzano il sistema per organizzare viaggi ed unirsi a viaggi organizzati da altri utenti.

Di ogni **utente** interessa conoscere il **nome, il cognome, l'indirizzo email, la città di provenienza e la data di iscrizione**.

Ogni **utente** può creare viaggi in qualità di **organizzatore**; dei **viaggi** interessa il **nome, il numero minimo e il numero massimo di partecipanti**. Gli utenti di *TravelPlan* possono quindi prendere parte ai viaggi organizzati dagli altri utenti.

Un **viaggio** è associato ad un certo numero di **attività** quali visite, pasti, tour, trasporti e pernottamenti. Di ogni **attività** interessa conoscere il **nome, l'istante di inizio, la durata, il prezzo, il luogo in cui si tiene, delle informazioni testuali ed eventualmente un insieme di codici identificativi di biglietti/prenotazioni relativi all'attività**.

Di ogni **luogo** interessa conoscere **l'indirizzo, la città, la regione e la nazione**.

Per facilitare la pianificazione, il sistema deve permettere di creare **attività composte**, cioè **che raggruppano più attività semplici**. Ad esempio, l'attività "tour gastronomico del centro storico" includerà la visita di due piazze, un percorso a piedi, un pasto al ristorante e la visita di una cantina.

Il sistema deve poter rappresentare gli **spostamenti** (**con qualsiasi mezzo**: treno, aereo, nave, bicicletta, a piedi, ecc.) come attività, delle quali è di interesse **conoscere sia il luogo di partenza che quello di arrivo**.

I **pernottamenti** sono attività particolari, **poiché il sistema non deve poter rappresentare più di un pernottamento al giorno per ogni utente che partecipa a un viaggio**.

Poiché può succedere che gli utenti che partecipano a uno stesso viaggio non prendano tutti parte alle stesse attività (p.es., due sotto-gruppi possono alloggiare in hotel diversi), **il sistema deve permettere di specificare, per ogni attività inclusa in un viaggio, gli utenti che vi partecipano**. Se per una certa attività non è specificato nessun utente, si assume che tutti i partecipanti al viaggio vi prendano parte.



I partecipanti ad un viaggio possono dare un **feedback** all'organizzazione del viaggio, assegnando **un voto da 1 a 5**.

TravelPlan prevede un sistema di punteggi per i propri utenti, in base alle valutazioni ricevute sui viaggi da loro organizzati. Il **punteggio p di un utente è così calcolato: p è uguale a 0 se la media aritmetica dei voti ricevuti è ≤ 3 , altrimenti p è uguale alla parte intera inferiore del 10% del numero dei feedback ricevuti con voto ≥ 4 .**

Il sistema deve permettere agli utenti registrati di partecipare a viaggi e di creare viaggi in qualità di organizzatori, aggiungendo a essi attività di vario tipo.

Infine, il sistema deve offrire le seguenti funzionalità avanzate:

1. Gli utenti registrati devono poter:
 - 1.1. creare viaggi in qualità di organizzatori e aggiungere a essi attività di vario tipo;
 - 1.2. trovare tutti i viaggi che includono una certa destinazione in un certo intervallo di tempo;
 - 1.3. trovare la/e città toccata/e dal maggior numero di viaggi in un certo intervallo di tempo;
 - 1.4. calcolare, per ogni regione di una data nazione, il numero di viaggi organizzati in un dato periodo di tempo che toccano quella regione;
 - 1.5. dato un budget minimo, un budget massimo, un insieme di regioni, un periodo di tempo e un punteggio, calcolare l'insieme dei viaggi che hanno un budget nell'intervallo richiesto, toccano almeno una delle regioni date, si svolgono completamente all'interno del periodo dato e sono organizzati da un utente con un punteggio pari o superiore a quello dato.
2. I responsabili del sistema devono poter calcolare, data una città di destinazione, per ognuno dei 12 mesi dell'anno, il numero di viaggi organizzati in quel mese nell'ultimo anno solare.

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: Stringa
- cognome: Stringa
- email: Email
- uttsc: DatiOra
- associato ai Viaggi
e cui partecipa
- associato ai Viaggi
che crea
- associato ai FeedBack

Viaggio:

- nome: Stringa
- min: Intero > 0
- max: Intero > 0
- associato alle Attività

Attività:

- nome: Stringa
- initia: DatiOra
- durata: Durata
- prezzo: Valuta
- info: Stringa
- cod: Stringa
- associato al luogo
- specificazione:
 - Pernottamenti
 - Spostamenti
- associato ai luoghi di
partenza e arrivo
- associato al Prezzo

Attività composte:

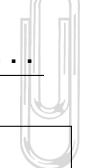
- associale alle Attività

luogo:

- individuato: Indirizzo
- associato alla città

Feed Back:

- valo: 1..5



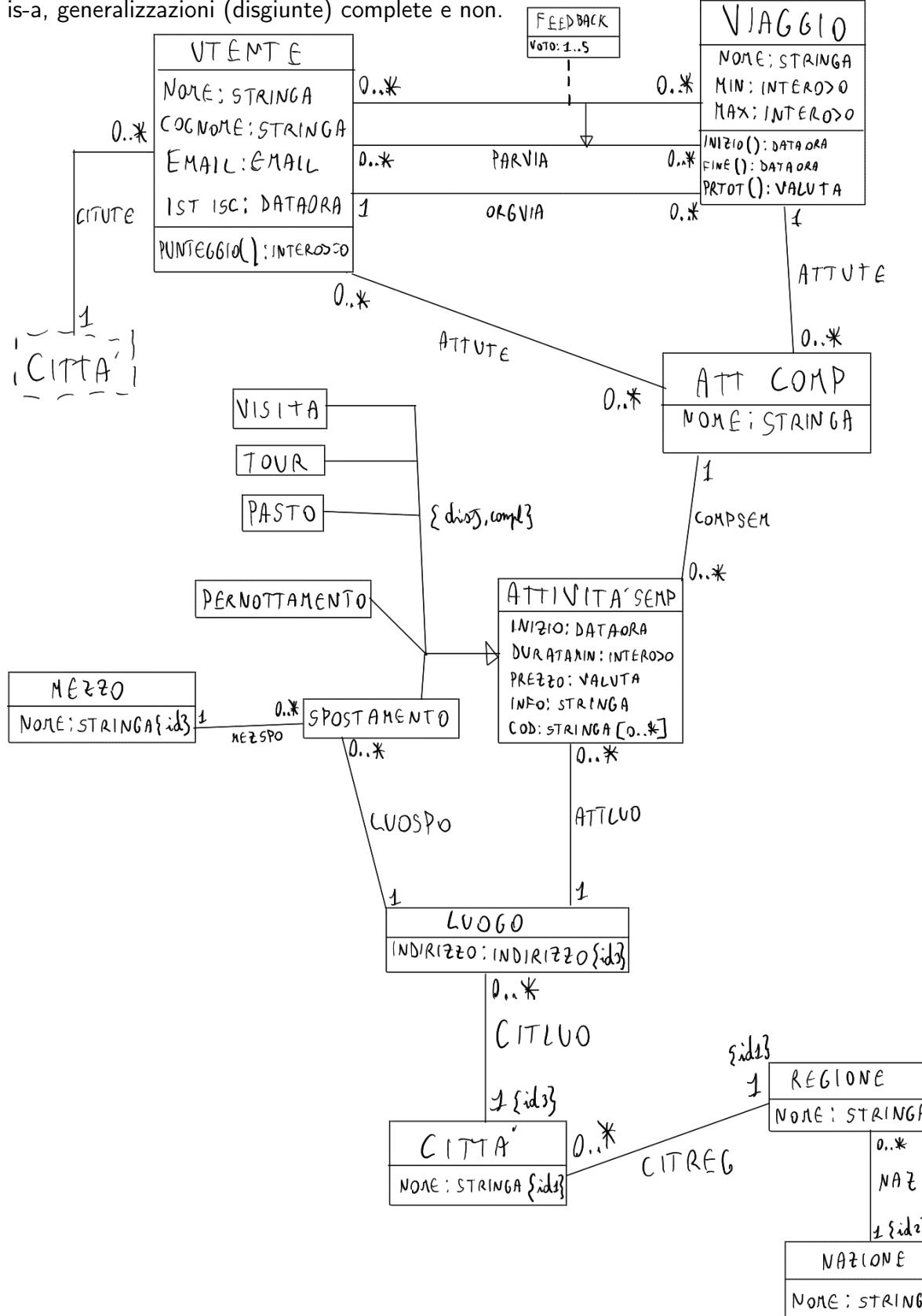
Risposta alla Domanda 1 (segue)

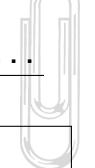
Domanda 2 (45 minuti; 75 minuti al massimo) Proseguire la fase di Analisi Concettuale dei requisiti, producendo un diagramma ER concettuale per l'applicazione, il dizionario dei dati ed eventuali vincoli esterni.

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

3] Tipo: **Entità** | Relationship (cerchiare)

Nome: ...VIAGGIO.....

attributo	dominio	moltep. (*)

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

Vincoli:

$$\forall v. \text{VIAGGIO}. \text{MIN} \leq \text{MAX}$$

$$\forall v, \text{min}, \text{max} \quad \text{VIAGGIO}(v) \wedge \text{MIN}(v, \text{min}) \wedge \text{MAX}(v, \text{max}) \rightarrow \text{min} \leq \text{max}$$

$$\forall v. \text{VIAGGIO}. \text{PERSONE} \leq \text{MAX}$$

$$\forall v, m \quad \text{VIAGGIO}(v) \wedge \text{MAX}(m, v) \rightarrow |\{u \mid \text{PARVIA}(u, v)\}| \leq m$$

5] Tipo: **Entità** | Relationship (cerchiare)

Nome: ATT COMP

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

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

Vincoli:

$$\begin{array}{l} \forall a \exists as \exists i \exists d \exists t \\ \text{ATTCOMP}(a) \rightarrow \neg \exists as' \exists i' \exists d' \exists t' \\ \text{CORPSEN}(a, as) \wedge \text{CORPSEN}(a, as') \wedge \text{INIZIO}(as, i) \wedge \text{INIZIO}(as', i') \\ \wedge \text{DURATA}(as, d) \wedge \text{DURATA}(as', d') \wedge \text{DATAORA}(t) \wedge t \geq i \wedge t \geq i' \wedge \\ t \leq i+d \wedge t \leq i'+d' \end{array}$$

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

Nome:

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

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

Vincoli:

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

Nome:

attributo	dominio	moltepl. (*)

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

Vincoli:

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

Nome:

attributo	dominio	moltepl. (*)

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

Vincoli:

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

Nome:

attributo	dominio	moltepl. (*)

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

Vincoli:

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

Nome:

attributo	dominio	moltepl. (*)

(*) 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

EMAIL: STRINGA SECONDO STANDARD

INDIRIZZO: (VIA=STRINGA, CIVICO=STRINGA[0..1], CAP=[0-9]{5})

VALUTA: (PRE=[A-Z]{3}, IMPORTO=REALE ≥ 0)

VOTE. PUNTEGGIO(); INTERO ≥ 0

-pre: $\exists v \text{ ORGVIA}(\text{this}, v)$

-post: VTOT = $\{(u, via, v) \mid \text{ORGVIA}(\text{this}, via) \wedge \text{FEEDBACK}(u, via) \wedge \text{VOTO}(u, via, v)\}$

$V_4 = \{(u, via, v) \mid \text{ORGVIA}(\text{this}, via) \wedge \text{FEEDBACK}(u, via) \wedge \text{VOTO}(u, via, v) \wedge v \geq 4\}$

$\left(\frac{\sum(v)}{(\text{UNION}_N) \text{ VTOT}} \leq 3 \rightarrow \text{RESULT} = 0 \right) \wedge \left(\frac{\sum(v)}{(\text{UNION}_N) \text{ VTOT}} > 3 \rightarrow \text{RESULT} = \lfloor 0.1 |V_4| \rfloor \right)$

VIAGGIO.INIZIO(); DATAORA

-pre: $\exists ac, a \text{ ATTVIA}(\text{this}, ac) \wedge \text{COMPSEM}(ac, a)$

-post: $A = \{(a, i) \mid \text{ATTSEM}(a) \wedge \exists ac \text{ COMPSEM}(a, ac) \wedge \text{ATTVIA}(ac, \text{this})\}$

$$\text{RETURN} = \min_{(a, i) \in A} (i)$$

VIAGGIO.FINE(); DATAORA

-pre: $\exists ac, a \text{ ATTVIA}(\text{this}, ac) \wedge \text{COMPSEM}(ac, a)$

-post: $A = \{(a, f) \mid \text{ATTSEM}(a) \wedge \exists ac, i, d \text{ COMPSEM}(a, ac) \wedge \text{ATTVIA}(ac, \text{this})\}$

$$\text{RETURN} = \max_{(a, f) \in A} (f)$$

Risposta alla Domanda 2 (segue)

VIAGGIO.PRTOT(): VALUTA

- pre: $\exists ac, a \text{ ATTVA}(this, ac) \wedge \text{COMPSEM}(ac, a)$

- post: $A = \left\{ (a, p) \mid \begin{array}{l} \text{ATTSEM}(a) \wedge \exists ac \text{ COMPSEM}(a, ac) \wedge \text{ATTVA}(ac, this) \\ \wedge \text{PREZZO}(a, p) \end{array} \right\}$

$$\text{RETURN} = \sum_{(a, p) \in A} p$$

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:

NEW_UTENTE(n ; STRINGA); UTENTE

2 Specifica use-case: ORGANIZZA VIAGGI (nome use-case)

Operazioni dello use-case:

NEW_VIAGGIO(u ; UTENTE, n ; STRINGA, min:INTERO>0, max:INTERO>0); VIAGGIO

NEW_COMP(v ; VIAGGIO, n ; STRINGA, A; ATTSEMP [1..*]); ATTCOMP

NEW_SEMP(i ; DATAORA, d; INTERO>0, p; VALUTA, l; LUOGO[1..2], info; STRINGA, c; STRINGA[0..*]); ATTSEMP

3 Specifica use-case: PARTECIPA VIAGGI (nome use-case)

Operazioni dello use-case:

PARVIAGGIO(u ; UTENTE, v; VIAGGIO)

PARATT(u ; UTENTE, a; ATTCOMP)

4 Specifica use-case: ...CERCA VIAGGI..... (nome use-case)

Operazioni dello use-case:

 $CERCADEST(c:CITTÀ, i: DATAORA, f: DATAORA); VIAGGIO[0..*]$ $MAXCITTÀ(i: DATAORA, f: DATAORA); CITTÀ[0..*]$ $VIAREC(n: NAZIONE, i: DATAORA, f: DATAORA); (REGIONE, INTERO)[0..*]$ $CERCA BUDGET(bmin: VALUTA, bmax: VALUTA, R: REGIONE[0..*], i: DATAORA, f: DATAORA, p: INTERO \geq 0); VIAGGIO[0..*]$ **5 Specifica use-case: ...STATISTICHE..... (nome use-case)**

Operazioni dello use-case:

 $CITMSE(c:CITTÀ); INTERO \geq 0 [12]$ **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

CERCA DEST (c: CITTÀ', i; DATAORA, f; DATAORA) : VIAGGIO [0..*]

- pre: $i \leq f$

- post:

$$\text{RESULT} = \left\{ v \mid \begin{array}{l} \exists ac, as, l, ia, d \text{ ATTRIA}(ac, v) \wedge \text{COMPSEM}(ac, as) \\ \wedge \text{ATTRIO}(as, l) \wedge \text{CITLVO}(l, c) \wedge \text{INIZIO}(as, ia) \wedge \text{DURATA}(as, d) \wedge \\ ia \geq i \wedge ia + d \leq f \end{array} \right\}$$

MAX CITTÀ' (i; DATAORA, f; DATAORA) : CITTÀ' [0..*]

- pre: $i \leq f$

- post:

$$C = \left\{ (c, n) \mid \begin{array}{l} \exists ac, as, l \text{ ATTRIA}(ac, v) \wedge \text{COMPSEM}(ac, as) \wedge \\ \text{ATTRIO}(as, l) \wedge \text{CITLVO}(c, l) \end{array} \right\}$$

$$\text{RESULT} = \left\{ c \mid \begin{array}{l} \exists m \ (c, m) \in C \wedge \exists c', m' \ (c', m') \in C \\ c' > c \end{array} \right\}$$

VIAREC (n: NAZIONE, i; DATAORA, f; DATAORA) : (REGIONE, INTERO>=0) [0..*]

- pre: $i \leq f$

- post:

$$\text{RESULT} = \left\{ (n, m) \mid \begin{array}{l} \exists ac, as, l, c \text{ ATTRIA}(ac, v) \wedge \text{COMPSEM}(ac, as) \wedge \\ \text{ATTRIO}(as, l) \wedge \text{CITLVO}(c, l) \wedge \text{CITREG}(c, n) \end{array} \right\}$$

CITMESE (c: CITTÀ') : (1..12, INTERO>=0) [12]

- pre:

- post:

$$\text{RESULT} = \left\{ (m, n) \mid \begin{array}{l} m \geq 1 \wedge m \leq 12 \wedge \\ \exists as, ac, l, i, f \ \text{ATTRIA}(ac, v) \wedge \text{COMPSEM}(ac, as) \wedge \text{LVOATT}(l, as) \wedge \text{INIZIO}(as, i) \\ \wedge \text{MESE}(i, m) \wedge \text{ANNO}(i, \text{ADESSO-1ANNO}) \end{array} \right\}$$

Risposta alla Domanda 5 (segue)

$\text{CERCA_BUDGET}(\text{lnr}; \text{VALUTA}, \text{lmax}; \text{VALUTA}, \text{R}; \text{REGIONE}[0..*], \text{i} : \text{DATAORA}, \text{f} : \text{DATAORA}, \text{t} : \text{INTERO} \geq 0) : \text{VIAGGIO}[0..*]$

- pre: $i \leq f \wedge l_{\min} \leq l_{\max}$

- post:

$$\text{RESULT} = \left\{ \begin{array}{l} V \\ | \quad \left| \begin{array}{l} \text{VIAGGIO}(v) \wedge \exists p_r \text{ PRTOT}(v, p_r) \wedge p_r \geq l_{\min} \wedge p_r \leq l_{\max} \wedge \\ \exists i_r, f_r \text{ ITZIO}(v, i_r) \wedge \text{FINE}(v, f_r) \wedge i_r \geq i \wedge f_r \leq f \wedge \\ \exists ac, as, l, c, m \text{ ATTVIA}(ac, v) \wedge \text{COMPSEM}(ac, as) \wedge \text{LUOATT}(l, as) \\ \wedge \text{CITLUO}(c, l) \wedge \text{CITREG}(c, m) \wedge m \in R \wedge \\ \exists u, p_{\min} \text{ ORGRIA}(u, v) \wedge \text{PUNTEGGIO}(u, p_{\min}) \wedge p_r \leq p_{\min} \end{array} \right. \end{array} \right\}$$

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

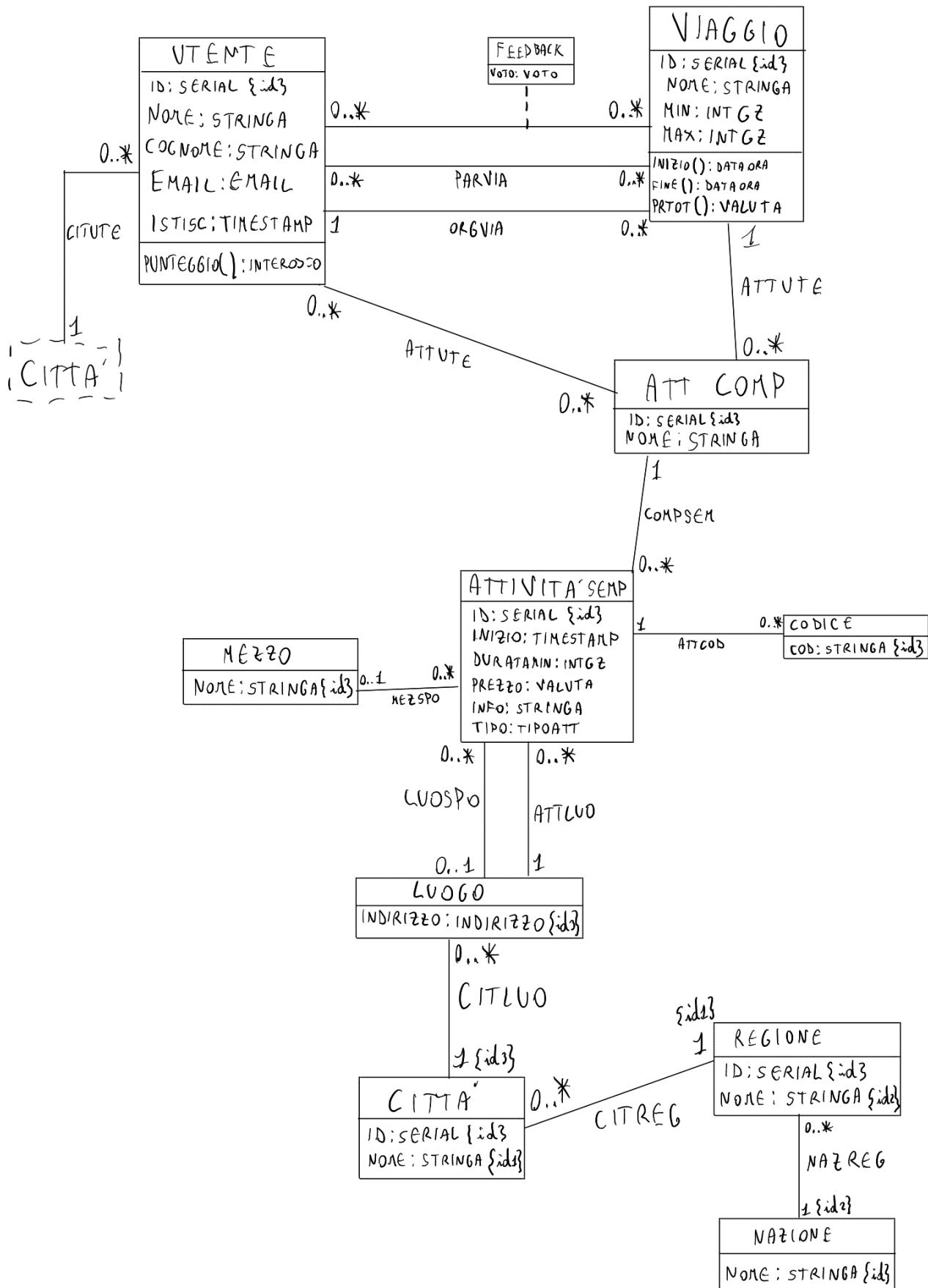
Corrispondenza tra domini concettuali e domini supportati dal DBMS

```

CREATE DOMAIN STRINGA AS VARCHAR NOT NULL;
CREATE DOMAIN EMAIL AS VARCHAR;
CREATE DOMAIN REALGEZ AS REAL CHECK(VALUE>=0);
CREATE TYPE VALUTA AS(
    PRE VARCHAR ILIKE '[A-Z]{3}',
    IMPORTO REALGEZ);
CREATE TYPE INDIRIZZO AS(
    VIA STRINGA,
    CIVICO VARCHAR,
    CAP VARCHAR ILIKE '[0-9]{5}');
CREATE DOMAIN INTGEZ AS INTEGER CHECK(VALUE>=0);
CREATE DOMAIN VOTO AS INTEGER CHECK(VALUE>=1 AND VALUE<=5);
CREATE TYPE TIPOATT AS ENUM('VIS','TOUR','PAS','SPO','PERN');
CREATE DOMAIN NMESSE AS INTEGER CHECK(VALUE>=1 AND VALUE<=12);

```

Diagramma ER ristrutturato



Breve descrizione delle scelte effettuate durante la ristrutturazione

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)

$\forall u \text{ SPOSTAMENTO_LUGO} \text{ MEZZO}$

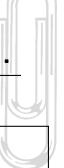
$$\forall a \text{ ATTSEMP}(a) \wedge \text{TIPOATT}(a, 'SPO') \leftrightarrow \exists l, m \text{ LUOSPOL}(l, a) \wedge \text{MEZZO}(m)$$

$\forall u \text{ UTENTE_PERNODISI}$

$$\begin{aligned} & \forall u \text{ UTENTE}(u) \rightarrow \exists ac, ac', as, as', i, i', di, di' \\ & \text{ATTUTE}(u, ac) \wedge \text{ATTUTE}(u, ac') \wedge \text{COMPSEM}(ac, as) \wedge \\ & \text{COMPSEM}(ac', as') \wedge as \neq as' \wedge \text{INIZIO}(i, as) \wedge \text{INIZIO}(i', as') \wedge \\ & \text{TIPOATT}(as, 'PERN') \wedge \text{TIPOATT}(as', 'PERN') \wedge \text{DATA}(di, i) \wedge \text{DATA}(di', i') \wedge di = di' \end{aligned}$$

$\forall u, v \text{ FEEDBACK_SEPARTECIPA}$

$$\forall u, v \text{ FEEDBACK}(u, v) \rightarrow \text{PARVIA}(u, v)$$



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	<u>UTENTE</u> ... (nome)	Derivante da: entità relationship (cerchiare)
Attributi	<u>ID</u> NOME COGNOME EMAIL TIMESTAMP CITTÀ	
Domini	SERIAL STRINGA STRINGA EMAIL TIMESTAMP 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):		
$\text{FK}(\text{CITTÀ}) \text{ REF } \text{CITTÀ}(ID)$		
La relazione accorda le relazioni che implementano le seguenti relationship: <u>CITUTE</u>		

2 Relazione	<u>ORGANIZZAZIONE</u> ... (nome)	Derivante da: entità relationship (cerchiare)
Attributi	<u>ID</u> NOME MIN MAX ORG	
Domini	SERIAL STRINGA INTGZ INTGZ 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):		
$\text{FK}(\text{ORG}) \text{ REF } \text{UTENTE}(ID)$		
$\text{CHECK}(\text{MIN} \leq \text{MAX})$		
La relazione accorda le relazioni che implementano le seguenti relationship: <u>ORGVIA</u>		

3 Relazione	<u>PARVIA</u> ... (nome)	Derivante da: entità relationship (cerchiare)
Attributi	<u>UTENTE</u> VIAGGIO VOTO	
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):		
$\text{FK}(\text{UTENTE}) \text{ REF } \text{UTENTE}(ID)$		
$\text{FK}(\text{VIAGGIO}) \text{ REF } \text{VIAGGIO}(ID)$		
La relazione accorda le relazioni che implementano le seguenti relationship:		

4 Relazione	<u>PFEEDBACK</u> .. (nome)	Derivante da: entità relationship (cerchiare)
Attributi	<u>UTENTE</u> VIAGGIO VOTO	
Domini	INTEGER INTEGER 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):		
$\text{FK}(\text{UTENTE}) \text{ REF } \text{UTENTE}(ID)$ $(\text{UTENTE}, \text{VIAGGIO}) \subseteq \text{PARVIA}(\text{UTENTE}, \text{VIAGGIO})$		
$\text{FK}(\text{VIAGGIO}) \text{ REF } \text{VIAGGIO}(ID)$		
La relazione accorda le relazioni che implementano le seguenti relationship:		

5 Relazione	<u>ATTIVITA</u> ... (nome)	Derivante da: entità relationship (cerchiare)
Attributi	<u>ID</u> NOME VIAGGIO	
Domini	SERIAL STRINGA 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):		
$\text{FK}(\text{VIAGGIO}) \text{ REF } \text{VIAGGIO}(ID)$		
La relazione accorda le relazioni che implementano le seguenti relationship: <u>ATTIVIA</u>		

6 Relazione ...ATT.VT.E... (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>VTENTE</u> ATT	
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):

FK(VTENTE) REF VTENTE(ID)

FK(ATT) REF ATT COMP(ID)

La relazione accorda le relazioni che implementano le seguenti relationship:

7 Relazione .ATT.SEMP. (nome) 1	Derivante da: entità relationship (cerchiare)
Attributi ID INIZIO DURATAMIN PREZZO INFO TIPO ATTIVO ATTCITTÀ	
Domini SERIAL TIMESTAMP INT62 VALUTA STRINGA TIPOATT INDIRIZZO 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(ATTIVO, ATTCITTÀ) REF LUOGO(INDIRIZZO, CITTÀ) FK(LUOGO, CITSP0) REF LUOGO(INDIRIZZO, CITTÀ)

FK(ATTCOMP) REF ATT COMP(ID) FK(MEZZO) REF MEZZO(NOME)

La relazione accorda le relazioni che implementano le seguenti relationship:

8 Relazione .ATT.SEMP. (nome) 2	Derivante da: entità relationship (cerchiare)
Attributi LUOSP0* CITSP0* MEZZO* ATT COMP	
Domini INDIRIZZO INTEGER STRINGA 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):

CHECK((TIPOC='SPO' AND LUOSP0>NULL AND CITSP0>NULL AND MEZZO>NULL) OR

(TIPOC='SPO' AND LUOSP0= NULL AND CITSP0= NULL AND MEZZO= NULL))

La relazione accorda le relazioni che implementano le seguenti relationship: COMP, MEZZO, LUOSP0, ATTIVO

9 Relazione .MEZZO..... (nome)	Derivante da: entità relationship (cerchiare)
Attributi NOME	
Domini STRINGA	

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:

10 Relazione ..CODICE.... (nome)	Derivante da: entità relationship (cerchiare)
Attributi COD ATT	
Domini STRINGA 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(ATT) REF ATT SEMP(ID)

La relazione accorda le relazioni che implementano le seguenti relationship:ATT COD

11 Relazione ...LV.0G.0.... (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>INDIRIZZO</u> CITTÀ'	
Domini <u>INDIRIZZO</u> 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(CITTÀ') REF CITTÀ'(ID)

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

12 Relazione ...CITTÀ'.... (nome)	Derivante da: entità relationship (cerchiare)
Attributi ID NOME REGIONE	
Domini SERIAL STRINGA 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(REGIONE) REF REGIONE(ID)

UNIQUE(NOME, REGIONE)

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

13 Relazione ...REGIONE.... (nome)	Derivante da: entità relationship (cerchiare)
Attributi ID NOME NAZIONE	
Domini SERIAL STRINGA STRINGA	

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) REF NAZIONE(NOME)

UNIQUE(NOME, NAZIONE)

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

14 Relazione ...NAZIONE.... (nome)	Derivante da: entità relationship (cerchiare)
Attributi NOME	
Domini STRINGA	

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 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.

~~REVOKE UPDATE ON ATTUTE
T, UTENTE, ATTIVITA' IN VIAGGIO~~

- INSERT ATTUTE

- OK = EXIST (SELECT * FROM ATTCOMP a, PARVIA pr
WHERE pr.UTENTE = NEW.UTENTE AND a.ID = NEW.ATT
AND pr.VIAGGIO = a.VIAGGIO)

T, UTENTE, ATT PERIODIS

- INSERT ATTUTE

- ERROR = EXIST (SELECT * FROM ATTUTE au, ATTCOMP ac1, ATTCOMP ac2,
ATTSEMP as1, ATTSEMP as2
WHERE ac1.ID = NEW.ATT AND au.UTENTE = NEW.UTENTE AND
au.ATT = ac2.ID AND as1.ATTCOMP = ac1.ID AND as2.ATTCOMP = ac2.ID
AND TSRANGE(as1.INIZIO, as1.INIZIO + as1.DURATA) * TSRANGE(as1.INIZIO, as1.INIZIO + as1.DURATA) <> 'EMPTY')

OR EXIST (SELECT * FROM ATTUTE au, ATTCOMP ac1, ATTCOMP ac2,
ATTSEMP as1, ATTSEMP as2
WHERE ac1.ID = NEW.ATT AND au.UTENTE = NEW.UTENTE AND
au.ATT = ac2.ID AND as1.ATTCOMP = ac1.ID AND as2.ATTCOMP = ac2.ID
AND as1.TIPO = 'PERN' AND as2.TIPO = 'PERN'
AND CAST(as1.INIZIO AS DATE) = CAST(as2.INIZIO AS DATE))

T, ATTCOMP, SEMPDIS

- INSERT OR UPDATE ATTSEMP

- ERROR = EXIST (SELECT * FROM ATTSEMP
WHERE ATTCOMP = NEW.ATT COMP
AND TSRANGE(INIZIO, INIZIO + DURATA) * TSRANGE(NEW.INIZIO, NEW.INIZIO + NEW.DURATA) <> 'EMPTY')

Risposta alla Domanda 7 (segue)

T, UTENTE, VIAGGIO DOPO ISC

- INSERT PARVIA

- ERROR=EXIST (SELECT * FROM UTENTE u, VIAGGIO v
WHERE u.ID=NEW.UTENTE AND
v.ID=NEW.VIAGGIO AND u.ISTISC>v.INIZIO())

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

CERCA DEST(c : INTEGER, λ : TIMESTAMP, f : TIMESTAMP); INTEGER[0..*]

ERROR IF $f < \lambda$

$R = \left(\begin{array}{l} \text{SELECT ac.VIAGGIO FROM ATTCOMP ac, ATTSEMP as} \\ \text{AS. ATTCOMP = ac.ID AND AS. ATTCITTA' = c} \\ \text{AND AS. INIZIO} \geq \lambda \text{ AND AS. INIZIO + AS. DURATA} \leq f \end{array} \right)$

RETURN R

MAXCITTA'(λ : TIMESTAMP, f : TIMESTAMP); INTEGER[0..*]

ERROR IF $f < \lambda$

$R = \text{WITH Q AS} \left(\begin{array}{l} \text{SELECT AS. ATTCITTA' C, COUNT(DISTINCT ac.VIAGGIO)m FROM ATTCOMP ac, ATTSEMP as} \\ \text{WHERE AS. ATTCOMP = ac.ID AND AS. INIZIO} \geq \lambda \\ \text{AND AS. INIZIO + AS. DURATA} \leq f \\ \text{GROUP BY AS. ATTCITTA'} \end{array} \right)$

SELECT C FROM Q

WHERE $m = (\text{SELECT MAX}(m) FROM Q)$

RETURN R

Risposta alla Domanda 8 (segue)

VIAREG(n ; INTEGER, i : TIMESTAMP, f ; TIMESTAMP): (INTEGER, INTEGER)[0..*]
 ERROR IF $f < i$

$R = \text{SELECT } M.ID, \text{COUNT}(\text{DISTINCT ac.VIAGGIO})N \text{ FROM ATTCOMP ac, ATTSEMP as, CITTÀ c, REGIONE m}$
 $\text{WHERE as.ATTCOMP=ac.ID AND as.INIZIO} \geq i$
 $\text{AND as.INIZIO + as.DURATA} \leq f \text{ AND as.ATTCITTA'} = c.ID \text{ AND}$
 $c.REGIONE = m.ID \text{ AND m.NAZIONE=}n$
 $\text{GROUP BY } M.ID$

RETURN R

CITMSE(c; INTEGER): (NMESE, INTEGER)[+2]

$R = \text{SELECT EXTRACT(MONTH FROM as.INIZIO), COUNT(DISTINCT ac.VIAGGIO)}$
 $\text{FROM ATTCOMP ac, ATTSEMP as}$
 $\text{WHERE as.ATTCOMP=ac.ID AND as.ATTCITTA'} = c \text{ AND}$
 $\text{EXTRACT(YEAR FROM as.INIZIO)} = \text{EXTRACT(YEAR FROM NOW(), INTERVAL '1 YEAR')}$
 $\text{GROUP BY EXTRACT(MONTH FROM as.INIZIO)}$

RETURN R

CERCA BUDGET(b_{\min} : VALUTA, b_{\max} : VALUTA, R : INTEGER[0..*], i : TIMESTAMP, f ; TIMESTAMP, p : INTEGER[0..*])

ERROR IF $b_{\min} > b_{\max}$ OR $i > f$

$R = \text{SELECT V.ID FROM VIAGGIO v, ATTCOMP ac, ATTSEMP as, CITTÀ c, UTENTE u}$
 $\text{WHERE as.ATTCOMP=ac.ID AND as.ATTCITTA'} = c.ID \text{ AND c.REGIONE IN R}$
 $\text{AND v.INIZIO}() \geq i \text{ AND v.FINE}() \leq f \text{ AND v.PRTOT}() > b_{\min} \text{ AND v.PRTOT}() < b_{\max}$
 $\text{AND v.ORGVIA} = u.ID \text{ AND u.PUNTEGGIO}() \geq p$

RETURN R

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.]

OPERAZIONI

VIAGGIO.INIZIO(): TIMESTAMP

```
RETURN = SELECT MIN(os.INIZIO) FROM ATTCOMP ac, ATTSEMPS os
WHERE ac.VIAGGIO = this.ID AND os.ATTCOMP = ac.ID
```

VIAGGIO.FINE(): TIMESTAMP

```
RETURN = SELECT MAX(os.INIZIO + os.DURATA) FROM ATTCOMP ac, ATTSEMPS os
WHERE ac.VIAGGIO = this.ID AND os.ATTCOMP = ac.ID
```

VIAGGIO.PREZZO(): VALUTA

```
RETURN = SELECT SUM(os.PREZZO) FROM ATTCOMP ac, ATTSEMPS os
WHERE ac.VIAGGIO = this.ID AND os.ATTCOMP = ac.ID
```

UTENTE.PUNTEGGIO(): INTEGER

```
RETURN = WITH VTOT AS (SELECT AVG(VOTO) m FROM FEEDBACK f, VIAGGIO v
WHERE f.VIAGGIO = v.ID AND v.ORGVIA = this.ID),
V4 AS (SELECT COUNT(VOTO) c FROM FEEDBACK f, VIAGGIO v
WHERE f.VIAGGIO = v.ID AND v.ORGVIA = this.ID AND f.VOTOS = 4)
```

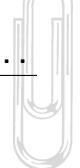
SELECT CASE

WHEN VTOT.m <= 3 THEN 0

ELSE FLOOR(0.1 * V4.c)

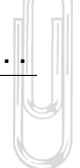
END

FROM VTOT, V4



[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.]