



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 20220506 (P.20220506)

RistoBook

Versione 2022-05-06

2

Specifiche dei Requisiti

Il sistema deve permettere ai **clienti** di effettuare prenotazioni presso i ristoranti iscritti, usufruendo, eventualmente, di promozioni. I ristoratori, invece, possono iscriversi per registrare i propri ristoranti e gestire le loro prenotazioni e le loro promozioni.

Dei **clienti** interessa conoscere il **nome** e l'**indirizzo e-mail**, mentre dei **ristoranti** interessa il **nome**, la **partita IVA** (una stringa numerica), **l'indirizzo**, **la città** e **l'insieme** di **tipologie di cucina offerte** (scelte da una lista tenuta sotto controllo dallo staff di RistoBook).

I **clienti** possono prenotare presso un ristorante specificando il **giorno**, **l'ora** e **il numero** di **commensali**. Le **prenotazioni dei clienti devono essere confermate** (o rifiutate) dal personale incaricato dei rispettivi ristoranti, che devono poter accedere ad RistoBook tramite una interfaccia dedicata.

Uno dei punti di forza del modello di business di RistoBook è la possibilità per i ristoranti di offrire e pubblicizzare scontistiche (dette **promozioni**). In particolare, i ristoratori devono poter definire una **promozione** specificando una **percentuale di sconto** sulle **prenotazioni consumate in un certo periodo di tempo**. Tali promozioni sono valide per al massimo un certo **numero di coperti al giorno**.

Ad esempio, un ristorante potrebbe definire la seguente promozione: sconto del 20% sulle prenotazioni dalle 20 alle 22 da martedì 15 giugno 2022 a venerdì 18 giugno 2022, valido per al massimo 10 coperti al giorno. Un altro ristorante potrebbe definire invece una promozione del tipo: sconto del 25% sulle prenotazioni dalle 18 alle 19 di tutti i martedì e giovedì dal 1 ottobre al 31 dicembre 2022.

Al momento della prenotazione, il cliente può scegliere una delle promozioni ancora disponibili (anche nessuna, se lo desidera).

Il sistema deve permettere ai ristoratori e ai clienti di gestire lo stato delle prenotazioni. In particolare, quando una **prenotazione** viene creata, questa è nello stato "**pendente**". Il ristoratore può scegliere se accettarla o rifiutarla. Quando il cliente usufruisce effettivamente della prenotazione, questa viene contrassegnata come "**completata**". Se, invece, il



cliente non dovesse presentarsi al ristorante, il ristoratore contrasseggerà la prenotazione come "non utilizzata". Inoltre, in ogni momento (prima del giorno ed ora prenotati) i clienti possono annullare le proprie prenotazioni, anche se già accettate. Infine, in caso di tutto esaurito (o per altre ragioni, ad es. giorni di chiusura), il responsabile di un ristorante deve poter chiudere le prenotazioni per un certo lasso di tempo (ad es., una certa data e fascia oraria, o un'intera settimana): da quel momento in poi, RistoBook non consentirà più ai clienti di richiedere prenotazioni in quel lasso di tempo (a meno che il ristoratore non le riapra).

Il sistema deve offrire, oltre quelle già descritte, le seguenti funzionalità ai suoi attori:

- Il sistema deve permettere ristoratori di RistoBook di calcolare alcune statistiche di utilizzo delle loro promozioni. In particolare, dato un periodo di tempo, RistoBook deve permettere calcolare, per ogni promozione del ristorante considerato, il numero medio di clienti al giorno (dove la media è calcolata sui giorni di validità della promozione) che ha utilizzato quella promozione in una prenotazione.
- Data una città x , un insieme di tipologie di cucina C , un tasso di sconto minimo s ed una data d , i clienti devono poter trovare quali sono i ristoranti nella città x che offrono almeno una delle tipologie di cucina in C e prevedono promozioni con sconti di tasso almeno s nella data d ancora utilizzabili per il numero di coperti che sono interessati a prenotare.

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

Ci sono:

- nome; String
- email; Email

- associato alle Prenotazioni:

Ristorante:

- nome; String
- P.IVA; IVA
- indirizzo; Indirizzo
- associato alla Città
- associato alle Cucine
- associato alle Promozioni
- associato alle Chiusure

Chiusura:

- orario; Delta Data ora

Prenotazione:

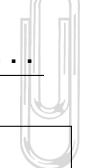
- giorno; Data
- ora; Ora
- nPersone; Intero > 0
- Specializzazione;

 - Rifiutata
 - Confermata;

 - Completa
 - Non Utilizzata

Promozione:

- giorni; Delta Data
- ore; Delta Ora
- NOperativi; Intero > 0
- Sconto; Perce



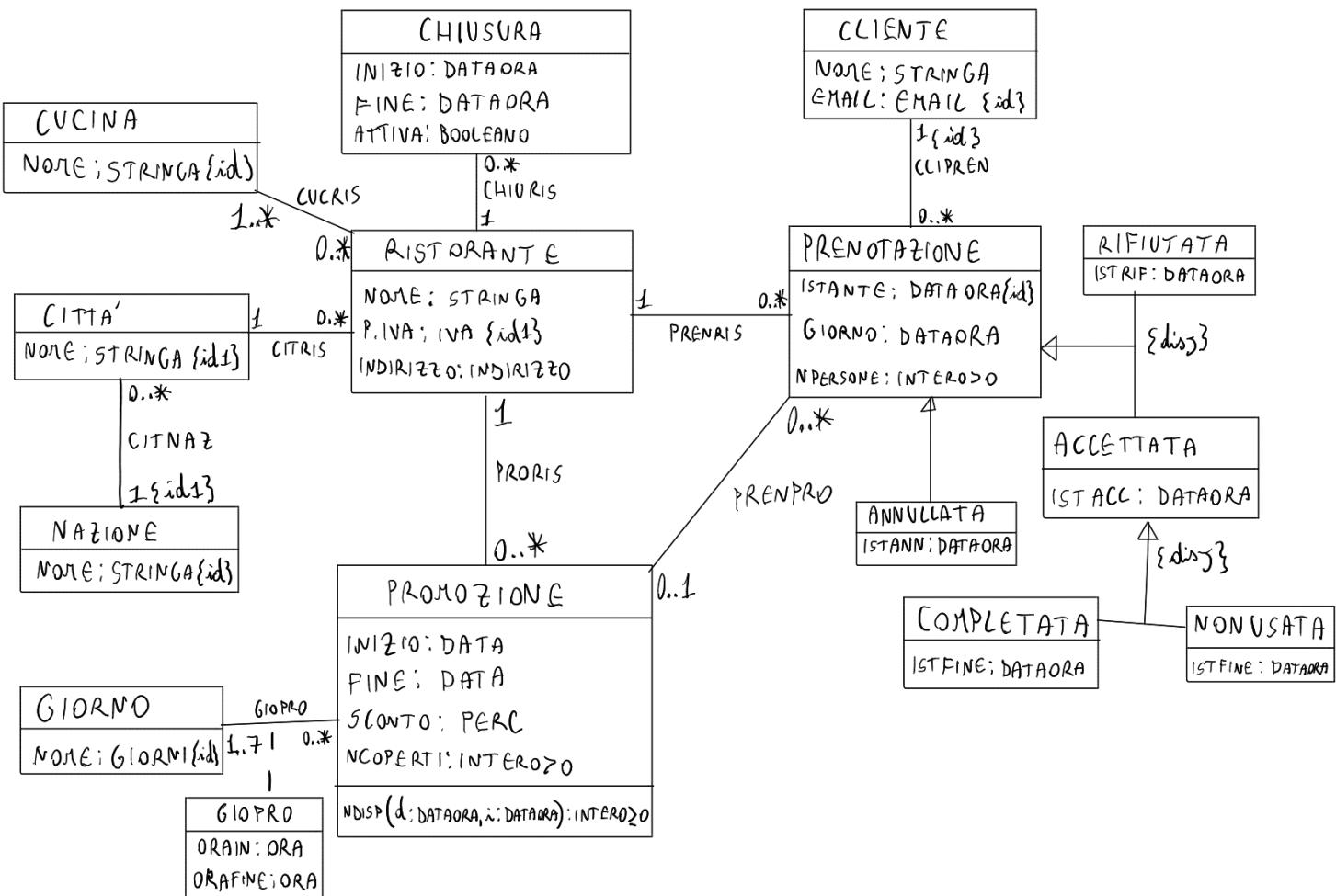
Risposta alla Domanda 1 (segue)

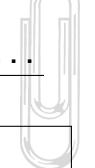
Domanda 2 (45 minuti; 75 minuti al massimo) Proseguire la fase di Analisi Concettuale dei requisiti, producendo un diagramma ER concettuale per l'applicazione, il dizionario dei dati ed eventuali vincoli esterni.

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>PRENOTAZIONE</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:</p> <p>$\forall p. \text{PRENOTAZIONE}(p) \wedge \text{ISTANTE}(p, i) \wedge \text{GIORNO}(p, g) \rightarrow i < g$</p> <p>$\forall p. \text{PRENOTAZIONE}(p) \rightarrow \exists i. \text{ISTANTE}(p, i)$</p> <p>$\forall p, p'. \text{PRENOTAZIONE}(p) \wedge \text{PRENOTAZIONE}(p') \wedge i. \text{ISTANTE}(p, i) \wedge i. \text{ISTANTE}(p', i) \rightarrow p \neq p'$</p>	attributo	dominio	moltep. (*)				<p>[2] Tipo: Entità Relationship (cerchiare)</p> <p>Nome: <u>ANNULLATA</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:</p> <p>$\forall a. \text{ANNULLATA}(a) \rightarrow \neg \text{COMPLETATA}(a) \wedge \neg \text{RIFUTATA}(a) \wedge \neg \text{NONUSATA}(a)$</p> <p>$\forall a, i, ia, d. \text{ANNULLATA}(a) \wedge \text{ISTANTE}(i, a) \wedge \text{ISTANTE}(ia, a) \wedge \text{GIORNO}(d, a) \rightarrow i < ia \wedge ia < d$</p>	attributo	dominio	moltep. (*)			
attributo	dominio	moltep. (*)											
attributo	dominio	moltep. (*)											

<p><input type="checkbox"/> 3 Tipo: Entità Relationship (cerchiare)</p> <p>Nome: <u>ACCETTATA</u></p> <table border="1"> <thead> <tr> <th>attributo</th> <th>dominio</th> <th>moltep. (*)</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> </tr> </tbody> </table>	attributo	dominio	moltep. (*)				<p><input type="checkbox"/> 5 Tipo: Entità Relationship (cerchiare)</p> <p>Nome: <u>RIFUTATA</u></p> <table border="1"> <thead> <tr> <th>attributo</th> <th>dominio</th> <th>moltep. (*)</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> </tr> </tbody> </table>	attributo	dominio	moltep. (*)			
attributo	dominio	moltep. (*)											
attributo	dominio	moltep. (*)											
<p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p> <p>$\forall . \text{ACCETTATA}, \text{DOPOCREATA}$</p> <p>$\forall a, i, ia \text{ ACCETTATA}(a) \wedge \text{ISTANTE}(i, a) \wedge \text{INSTACC}(ia, a)$ $\wedge \text{GIORNO}(d, a) \rightarrow i < ia \wedge ia < d$</p> <p>$\forall . \text{ACCETTATA}, \text{NOANNULLATA}$</p> <p>$\forall a, i \text{ ACCETTATA}(a) \wedge \text{INSTACC}(a, i)$ $\rightarrow \exists ia \text{ ANNULLATA}(a) \wedge \text{INSTANN}(a, ia) \wedge ia < i$</p>	<p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p> <p>$\forall . \text{RIFUTATA}, \text{DOPOCREATA}$</p> <p>$\forall a, i, ia \text{ RIFUTATA}(a) \wedge \text{ISTANTE}(i, a) \wedge \text{ISTRIF}(ia, a)$ $\rightarrow i < ia$</p>												

<p><input type="checkbox"/> 4 Tipo: Entità Relationship (cerchiare)</p> <p>Nome: <u>COMPLETATA</u></p> <table border="1"> <thead> <tr> <th>attributo</th> <th>dominio</th> <th>moltep. (*)</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> </tr> </tbody> </table>	attributo	dominio	moltep. (*)				<p><input type="checkbox"/> 6 Tipo: Entità Relationship (cerchiare)</p> <p>Nome: <u>NONUSATA</u></p> <table border="1"> <thead> <tr> <th>attributo</th> <th>dominio</th> <th>moltep. (*)</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> </tr> </tbody> </table>	attributo	dominio	moltep. (*)			
attributo	dominio	moltep. (*)											
attributo	dominio	moltep. (*)											
<p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p> <p>$\forall . \text{COMPLETATA}, \text{DOPOACCETTATA}$</p> <p>$\forall c, i, if \text{ COMPLETATA}(a) \wedge \text{INSTACC}(i, a) \wedge \text{INSTFINE}(if, a)$ $\rightarrow i < if$</p>	<p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p> <p>$\forall . \text{NONUSATA}, \text{DOPOACCETTATA}$</p> <p>$\forall c, i, if \text{ NONUSATA}(a) \wedge \text{INSTACC}(i, a) \wedge \text{INSTFINE}(if, a)$ $\rightarrow i < if$</p>												

<input type="checkbox"/> 7	Tipo: Entità Relationship (cerchiare)	
Nome:	PROMOZIONE.....	
attributo	dominio	moltep. (*)

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

Vincoli:

 $\forall \text{PROMOZIONE}.\text{ATTIVA}$ $\forall p, \text{pre}, g, i, f, d, o$ $\text{PREPRO}(\text{pre}, p) \wedge \text{GIORNO}(g, \text{pre}) \wedge \text{INIZIO}(i, \text{pre})$ $\wedge \text{FINE}(f, \text{pre}) \wedge \text{DATA}(d, g) \wedge \text{ORA}(o, g) \rightarrow i \leq d \wedge d \leq f \wedge$ $\exists o_i, o_f, gg, gset, m \text{ GIORNO}(gset, gg) \wedge o_i \leq gg \wedge gg \leq o_f \wedge$ $\text{ORAFINE}(gg, o_f) \wedge \text{GIORNOD}(gset, gg) \wedge o_i \leq o \wedge o \leq o_f \wedge$ $\text{GSET}(d, gset) \wedge \text{NPERSONE}(n, \text{pre}) \wedge n \leq \text{NDISP}(d, i)$ $\forall \text{PRENOTAZIONE}.\text{FINE DOPONIZIO}$ $\forall p, i, f \text{ PRENOTAZIONE}(p) \wedge \text{INIZIO}(i, p) \wedge \text{FINE}(f, p) \rightarrow i < f$

<input type="checkbox"/> 9	Tipo: Entità Relationship (cerchiare)	
Nome:	G.I.O.P.R.O.....	
attributo	dominio	moltep. (*)

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

Vincoli:

 $\forall \text{G.I.O.P.R.O}.\text{FINE DOPONIZIO}$ $\forall g, p, o_i, o_f \text{ GIORNO}(g, p) \wedge \text{ORAIN}(o_i, g) \wedge \text{ORAFINE}(o_f, g) \rightarrow o_i < o_f$ $\wedge \text{ORAFINE}(o_f, g) \rightarrow o_i < o_f$

<input type="checkbox"/> 8	Tipo: Entità Relationship (cerchiare)	
Nome:	CHIUSURA.....	
attributo	dominio	moltep. (*)

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

Vincoli:

 $\forall \text{CHIUSURA}.\text{FINE DOPONIZIO}$ $\forall c, i, f \text{ CHIUSURA}(c) \wedge \text{INIZIO}(i, c) \wedge$ $\text{FINE}(c, f) \rightarrow i < f$

<input type="checkbox"/> 10	Tipo: Entità Relationship (cerchiare)	
Nome:	
attributo	dominio	moltep. (*)

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

Vincoli:

 $\forall \text{CHIUSURA}.\text{FINE DOPONIZIO}$ $\forall c, i, f$

<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

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

EMAIL: (STRINGA SECONDO STANDARD)

IVA: (STRINGA[0..9])

PERC: (REALE IN [0,1])

GIORNO: {LUN, MAR, MER, GIO, VEN, SAB, DOM}

PROMOZIONE.NDISPLAY(d:DATA, i:DATAORA); INTERO ≥ 0

-pre:

-post:

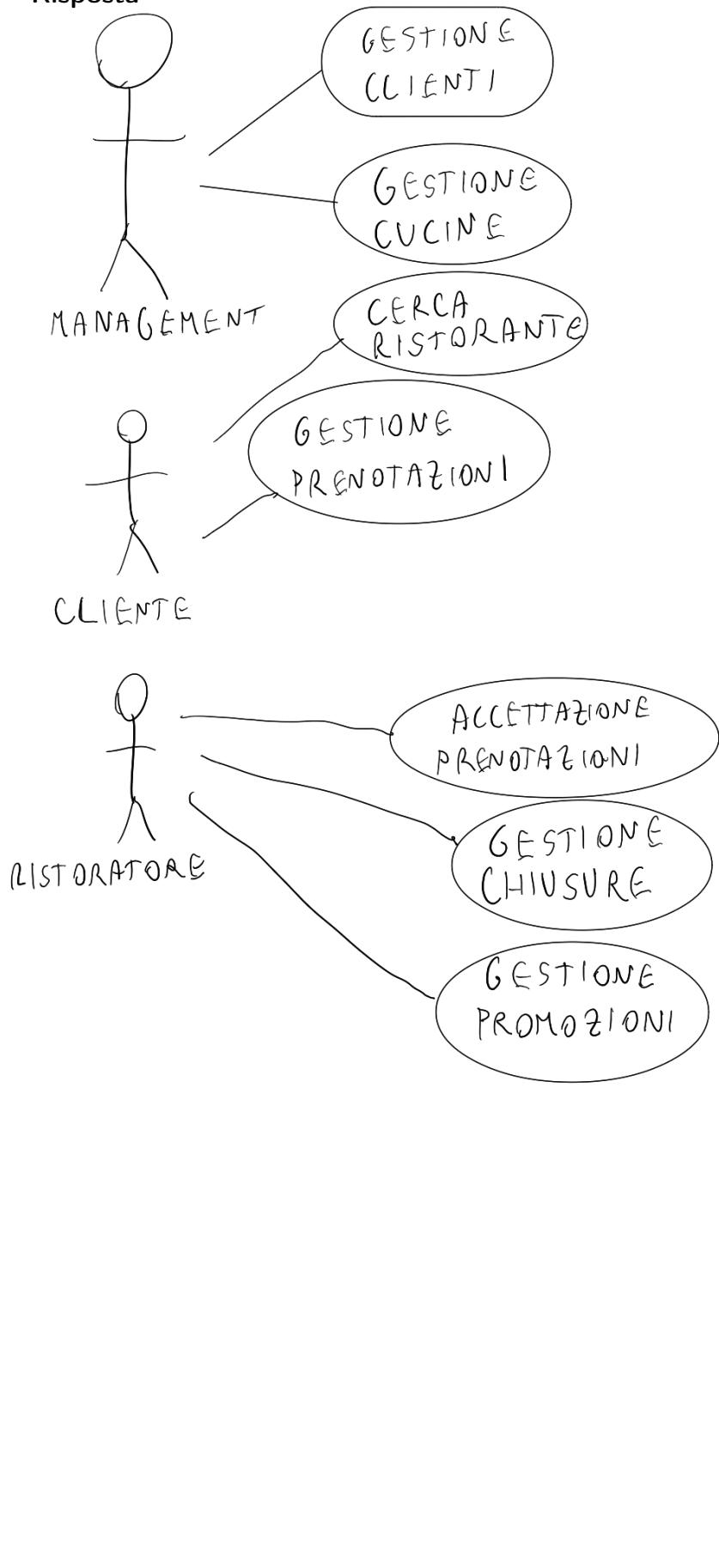
$$P = \left\{ (p, m_p) \mid \begin{array}{l} \text{PRENPROD}(p, \text{this}) \wedge \text{PERSONE}(p, m_p) \wedge \exists ip \text{ ISTANTE}(ip, p) \wedge \\ ip < i \wedge \exists d \text{ GIORNO}(d, ip) \wedge \text{ACCETTATA}(p) \wedge \neg \text{ANNULLATA}(p) \end{array} \right\}$$

Sia n tale che soddisfi: NCOPERTI(this, n)

$$\text{RESULT} = n - \left(\sum_{(p, m_p) \in P} m_p \right)$$

Risposta alla Domanda 2 (segue)

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

Risposta

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

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

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

Operazioni dello use-case:

NEWCLIENTE(N: STRINGA, E: EMAIL): CLIENTE

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

Operazioni dello use-case:

NEWCUCINA(N: STRINGA): CUCINA

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

Operazioni dello use-case:

NEW PREN(g: DATAORA, n: INTERO>0, r: RISTORANTE, p: PROMOZIONE[0..1]): PRENOTAZIONE

ANNULLA PREN(p: PRENOTAZIONE): ANNULLATA

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

Operazioni dello use-case:

CERCA PROM($a_1: \text{CITTÀ}$, $c: \text{CUCINA}[1..*]$, $d: \text{PERC}$, $d: \text{DATA}$, $n: \text{INTERO} > 0$) ; RISTORANTE[0..*]

5 Specifica use-case: ... ACCETTA PRENOTAZIONI (nome use-case)

Operazioni dello use-case:

ACCETTA PREN($p: \text{PRENOTAZIONE}$): ACCETTATA

COMPLETA PREN($p: \text{ACCETTATA}$): COMPLETATA

NON USA PREN($p: \text{ACCETTATA}$): NON USATA

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

Operazioni dello use-case:

NEW CHIUSURA($i: \text{DATAORA}$, $f: \text{DATAORA}$): CHIUSURA

ATTIVA CHIUSURA($c: \text{CHIUSURA}$): CHIUSURA

DISATTIVA CHIUSURA($c: \text{CHIUSURA}$): CHIUSURA

7 Specifica use-case: ... GESTIONE PROMOZIONI (nome use-case)

Operazioni dello use-case:

NEW PROM($i: \text{DATAORA}$, $f: \text{DATAORA}$, $d: \text{PERC}$, $n: \text{INTERO} > 0$, $g: \text{GIORNO}[0..7]$, $o_i: \text{ORA}[0..7]$, $o_f[0..7]$): PROMOZIONE

MEDIA PROM($m: \text{RISTORANTE}$, $i: \text{DATA}$, $f: \text{DATA}$): (PROMOZIONE, REALE > 0)[0..*]

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 PROM($ct : CITTA', c : CUCINA[0..*], p : PERC, d : DATA, n : INTERO > 0$) ; RISTORANTE[0..*]

- pre: $d \geq address$

- post:

$$\text{RESULT} = \left\{ \begin{array}{l} M \\ \quad \left| \begin{array}{l} \text{CITRIS}(m, ct) \wedge CUCRIS(m, cuc) \wedge cuc \in c \wedge (\exists pro, s) \text{PROMRIS}(pro, m) \wedge \\ SCONTOS(s, pro) \wedge s' > s \wedge (\exists di, df) \text{INIZIO}(pro, di) \wedge \text{FINE}(pro, df) \wedge \\ di \leq d \wedge df \geq d \wedge (\exists gg, gg') \text{GIOPROM}(gg, pro) \wedge \text{NOME}(gg, gg') \wedge \text{GSET}(di, gg') \wedge \\ NDISP(pro, d, address) \geq n \end{array} \right. \end{array} \right\}$$

Risposta alla Domanda 5 (segue)

MEDIA PROM ($m: RISTORANTE$, $i: DATA$, $f: DATA$): ($PROMOZIONE$, $REALE \geq 0$) [0..*]

- pre: $\exists p \text{ PROMRIS}(p, m) \wedge i \leq f$

- post:

$$\left\{ \begin{array}{l}
 \text{RESULT} \\
 \left(p, m \right)
 \end{array} \right| \left\{ \begin{array}{l}
 \text{PRORIS}(p, m) \wedge \\
 gg = \left\{ d \mid \begin{array}{l} \text{DATA}(d) \wedge \exists g, gret, ip, if \\ i \leq d \wedge ip \leq d \wedge f \geq d \wedge if \geq d \wedge GIOPROM(gg, p) \wedge \text{NONE}(gret, gg) \end{array} \right\} \wedge \\
 mg = |gg| \wedge \\
 \text{GSET}(d, gg) \\
 npers = \left\{ (pre, n) \mid \begin{array}{l} \text{PREPRO}(pre, p) \wedge \text{NPERSONE}(pre, n) \wedge \\ \exists g, d \text{ GIORNO}(g, pre) \wedge \text{DATA}(d, g) \wedge d \in gg \end{array} \right\} \wedge \\
 mp = \sum_{(pre, n) \in npers} n \quad \wedge \quad m = mp / mg
 \end{array} \right\}$$

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 VARCHAR NOT NULL;`

`CREATE TYPE INDIRIZZO(`
`VIA STRINGA,`
`CIVICO VARCHAR,`
`CAP CAP)`

`CREATE DOMAIN CAP AS VARCHAR CHECK(VALUE ~ [0-9]{5});`

`CREATE DOMAIN EMAIL AS STRINGA;`

`CREATE DOMAIN IVA AS VARCHAR CHECK(VALUE ~ [0-9]);`

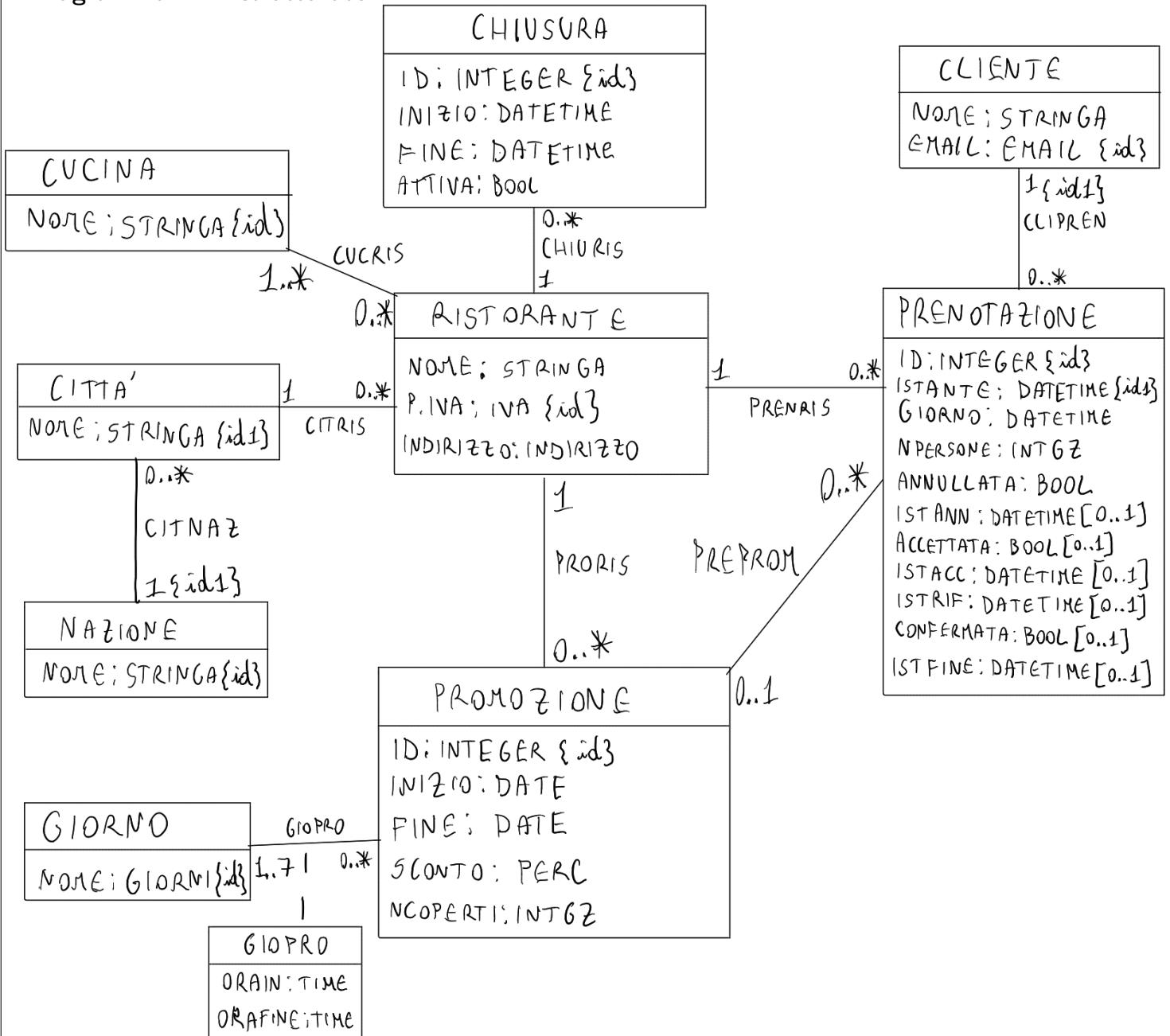
`CREATE DOMAIN PERC AS REAL CHECK(VALUE ≥ 0 AND VALUE ≤ 1);`

`CREATE TYPE GIORNI AS ENUM (LUN, MAR, MER, GIO, VEN, SAB, DOM);`

`CREATE DOMAIN INTGEZ AS INTEGER CHECK(VALUE > 0);`

`CREATE DOMAIN REALGEZ AS REAL CHECK(VALUE > 0);`

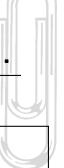
Diagramma ER ristrutturato



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)

- ✓ $\forall p \text{ PRENOTAZIONE}(p) \wedge \text{ANNULLATA}(p, \text{true}) \rightarrow \neg (\exists c, a \text{ CONFIRMED}(p, c) \vee \text{ACCEPTED}(p, \text{false}))$
- ✓ $\forall p \text{ PRENOTAZIONE}(p) \wedge \text{ANNULLATA}(p, \text{true}) \leftrightarrow \exists i, ia \text{ ISTANTE}(i, p) \wedge \text{ISTANN}(ia, p) \wedge i < ia$
- ✓ $\forall p \text{ PRENOTAZIONE}(p) \wedge \text{ACCEPTED}(p, \text{true}) \leftrightarrow \exists i, ia \text{ ISTANTE}(i, p) \wedge \text{ISTACC}(ia, p) \wedge i < ia$
- ✓ $\forall p \text{ PRENOTAZIONE}(p) \wedge \text{ACCEPTED}(p, \text{false}) \leftrightarrow \exists i, ir \text{ ISTANTE}(i, p) \wedge \text{ISTRIF}(ir, p) \wedge i < ir$
- ✓ $\forall p \text{ PRENOTAZIONE}(p) \wedge \exists c \text{ CONFIRMED}(p, c) \rightarrow \exists i, if \text{ ISTANTE}(i, p) \wedge \text{ISTFINE}(if, p) \wedge i < if$
- ✓ $\forall p \text{ PRENOTAZIONE}(p) \wedge \exists c \text{ CONFIRMED}(p, c) \rightarrow \text{ACCEPTED}(p, \text{true})$



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 RISTORANTE... (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>NOME</u> <u>P.IVA</u> <u>INDIRIZZO</u> <u>CITTÀ</u> <u>NAZIONE</u>	
Domini <u>STRINGA</u> <u>IVA</u> <u>INDIRIZZO</u> <u>STRINGA</u> <u>STRINGA</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):

FOREIGN KEY (CITTÀ,NAZIONE) REFERENCES CITTÀ(NOME,NAZIONE)

La relazione accorda le relazioni che implementano le seguenti relationship: .CIT.RIS.....

2 Relazione CITTÀ..... (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>NOME</u> <u>NAZIONE</u>	
Domini <u>STRINGA</u> <u>STRINGA</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):

FOREIGN KEY (NAZIONE) REFERENCES NAZIONE(NOME)

La relazione accorda le relazioni che implementano le seguenti relationship: .CIT.NAZ.....

3 Relazione NAZIONE.... (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>NOME</u>	
Domini <u>STRINGA</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:

4 Relazione CUCINA..... (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>NOME</u>	
Domini <u>STRINGA</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:

5 Relazione CUCRIS..... (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>CUCINA</u> <u>RIST</u>	
Domini <u>STRINGA</u> <u>IVA</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):

FOREIGN KEY (CUCINA) REFERENCES CUCINA(NOME)

FOREIGN KEY (RIST) REFERENCES RISTORANTE(P.IVA)

La relazione accorda le relazioni che implementano le seguenti relationship:

6 Relazione .CHIUSURA... (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>ID</u> INIZIO FINE ATTIVA RIST	
Domini INTEGER DATETIME DATETIME BOOL IVA	

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):

FOREIGN KEY (RIST) REFERENCES RISTORANTE(P.NA)

SERIAL(ID)

(CHECK(INIZIO < FINE))

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

7 Relazione PROMOZIONE (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>ID</u> INIZIO FINE SCONTI INCOPERTI RIST	
Domini INTEGER DATETIME DATETIME PERC INT62 IVA	

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):

FOREIGN KEY (RIST) REFERENCES RISTORANTE(P.NA)

SERIAL(ID)

(CHECK(INIZIO < FINE))

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

8 Relazione GIORNO.... (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>NOME</u>	
Domini GIORNI	

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:

9 Relazione GIORNO.PROM... (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>GIORNO</u> <u>PROM</u> ORAIN ORAFINE	
Domini GIORNI INTEGER TIME TIME	

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):

FOREIGN KEY (GIORNO) REFERENCES GIORNO(NOME)

(CHECK(ORAIN < ORAFINE))

FOREIGN KEY (PROM) REFERENCES PROMOZIONE(ID)

La relazione accorda le relazioni che implementano le seguenti relationship:

10 Relazione CLIENTE... (nome)	Derivante da: entità relationship (cerchiare)
Attributi NOME EMAIL	
Domini STRINGA EMAIL	

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

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

La relazione accorda le relazioni che implementano le seguenti relationship:

11 Relazione PRENOTAZIONE. (nome) 1

Derivante da: entità | relationship (cerchiare)

Attributi	<u>ID</u>	ISTANTE	GIORNO	N PERSONE	CLIENTE	RIST	PROM*	
Domini	INTEGER	DATETIME	DATETIME	INT62	EMAIL	IVA	INTEGER	

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

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

SERIAL (ID) FOREIGN KEY (CLIENTE) REFERENCES CLIENTE (EMAIL)

FOREIGN KEY (RIST) REFERENCES RISTORANTE (P.NA) FOREIGN KEY (PROM) REFERENCES PRO OZIONE (ID)

CHECK (CONFERMATA = NULL AND ISTANTE = NULL OR (CONFERMATA > NULL AND ISI FINE > ISTACC))

La relazione accorda le relazioni che implementano le seguenti relationship:

12 Relazione PRENOTAZIONE. (nome) 2

Derivante da: entità | relationship (cerchiare)

Attributi	ANNULLATA	ISTANN*	ACCETTATA*	ISTACC*	ISTRIF*	CONFIRMATA*	ISTFINE*	
Domini	BOOL	DATETIME	BOOL	DATETIME	DATETIME	BOOL	DATETIME	

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

AND ISTANN=NULL

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

CHECK (ANNULLATA = FALSE OR (ANNULLATA = TRUE AND ACCETTATA <> FALSE AND CONFIRMATA = NULL AND ISTANN > ISTANTE))

CHECK (ACCETTATA = NULL AND ISTACC = NULL OR (ACCETTATA = TRUE AND ISTACC > ISTANTE) OR (ACCETTATA = FALSE AND ISTRIF > ISTANTE))

CHECK (ISTANTE < GIORNO) AND ISTRIF = NULL

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.

T. PRENOTAZIONE.RISTORANTE APERTO

- INSERT PRENOTAZIONE

- ERROR = \exists (SELECT * FROM PRENOTAZIONE p, CHIUSURA C
WHERE p.RIST = C.RIST
AND p.GIORNO > C.INIZIO AND p.GIORNO < C.FINE
AND C.ATTIVA = TRUE))

T. PRENOTAZIONE.PROMRIS

- INSERT PRENOTAZIONE

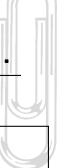
- OK = \exists (SELECT * FROM PRENOTAZIONE pre, PROMOZIONE pro
WHERE pre.RIST = pro.RIST
AND pre.PROM = pro.ID)

T. PROMOZIONE.ATTIVA

- INSERT PRENOTAZIONE

- OK = NEW.PROM = NULL OR

- \exists (SELECT * FROM PRENOTAZIONE pre, PROMOZIONE pro, GIOPRO gp
WHERE pre.PROM = pro.ID AND gp.PROM = pro.ID
AND TO_CHAR(pre.GIORNO, 'DY') = gp.GIORNO
AND CAST(pre.GIORNO AS DATE) >= pro.INIZIO AND CAST(pre.GIORNO AS DATE) <= pro.FINE
AND CAST(pre.GIORNO AS TIME) >= gp.DRAINI AND CAST(pre.GIORNO AS TIME) <= gp.DRAFINE
AND NDISP(pre.GIORNO) >= pre.NPERSONE)



Risposta alla Domanda 7 (segue)

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

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

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

Risposta

CERCA PROM (*cif: CITTÀ*; *c: CUCINA[1..*]*, *l: PERC*, *d: DATE*, *n: INT[2]*): (*IVA*) [0..*]

$R = \left(\begin{array}{l} \text{SELECT DISTINCT R.NA FROM RISTORANTE R, CUCRIS CR, PROMOZIONE P, GIOPROM GP, PRENOTAZIONE PR} \\ \text{WHERE CR.RIST = R.PIVA AND CR.CUC IN C AND P.RIST = R.PIVA AND GP.PROM = P.ID} \\ \text{AND P.SCONTI} \geq l \text{ AND } d \geq P.INIZIO \text{ AND } d \leq P.FINE} \\ \text{AND GP.GIORNO} \geq \text{TO_CHAR}(d, 'DD') \text{ AND PR.PROM} = P.ID} \\ \text{AND CAST(PR.GIORNO AS DATE)} = d \\ \text{GROUP BY P.ID HAVING (P.NCOPERTI - SUM(PR.NPERSONE))} \geq n \end{array} \right)$

RETURN R

MEDIA PROM (*m: RISTORANTE*, *i: DATE*, *f: DATE*): (*INTEGER, REAL[2]*) [0..*]

$OK = i \leq f \text{ AND EXISTS} \left(\begin{array}{l} \text{SELECT * FROM PROMOZIONE P} \\ \text{WHERE P.RIST} = m \end{array} \right)$

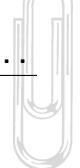
$R = \left(\begin{array}{l} \text{SELECT PRO.ID, AVG(SUM(P.NPERSONE)) AS AVG_PRM} \\ \text{FROM (SELECT PRO.ID, CAST(PR.GIORNO AS DATE) AS GP, SUM(PR.NPERSONE) AS SUM_PRM} \\ \text{FROM PROMOZIONE PRO, GIOPROM GP, PRENOTAZIONE PR} \\ \text{WHERE GP.PROM} = PRO.ID \text{ AND PR.PROM} = PRO.ID} \\ \text{AND CAST(PR.GIORNO AS DATE)} \leq f \text{ AND CAST(PR.GIORNO AS DATE)} \geq i} \\ \text{GROUP BY PRO.ID, GP} \\ \text{GROUP BY PRO.ID} \end{array} \right)$

RETURN R

Risposta alla Domanda 8 (segue)

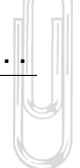
N DISP (p; INTEGER, d; DATE, i; DATETIME); INT67
R = SELECT pro.NCOPERTI - SUM(prc.NPERSONE)
FROM PROMOZIONE pro, PRENOTAZIONE prc
WHERE prc.PROID = pro.ID AND pro.ID = p
AND CAST(prc.GIORNO AS DATE) = d AND prc.ISTANTE < i
(GROUP BY pro.ID)

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



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

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



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