

Esame Es.20230130 – Prova scritta del 30 gennaio 2023

Si vuole progettare Slimmy, una nuova applicazione che permetta agli utenti di monitorare la loro alimentazione, tracciando i cibi che ingeriscono giornalmente (con relative quantità e nutrienti) per tenere sotto controllo apporto calorico ed equilibrio della loro dieta.

A seguito di interviste con il committente, è stata stilata la seguente specifica dei requisiti.

Slimmy deve consentire agli **utenti** di **registrarsi** al servizio fornendo **nome, cognome, nazionalità, email e password**.

Slimmy deve fornire un archivio di **schede alimenti** (**cibi e bevande**). **Tale archivio viene continuamente esteso, oltre che dalla redazione di Slimmy, anche dagli stessi utenti**. Per ogni **alimento** presente nell'archivio, la relativa scheda deve permettere di rappresentare il **nome, la marca (se disponibile), la grandezza della confezione (se disponibile), il codice identificativo (una stringa) codificato nel codice a barre della confezione (se disponibile), il relativo apporto calorico (in kCal) per unità di alimento, il valore di tale unità** (ad es., per i cibi liquidi: litri; per i solidi, grammi o etogrammi o chilogrammi; per alimenti come le uova, unità), **la quantità (in unità di alimento) di una porzione standard, oltre che la quantità (questa volta sempre in grammi, per garantire uniformità) di ogni nutrienti per unità di alimento**.

La **lista dei nutrienti** è **tipicamente stabile ed uguale per tutti gli alimenti** (ad es., carboidrati totali, zuccheri, proteine, grassi totali, grassi saturi, grassi insaturi, sodio, potassio, fibre, alcol, etc.), **ma Slimmy deve fare in modo che tale lista sia estendibile dall'autore dell'inserimento di una scheda alimento**. Viceversa, Slimmy deve permettere l'**inserimento di schede alimenti che non hanno valori per i nutrienti definiti 'opzionali'**. La lista dei **nutrienti opzionali** è **definita dalla redazione di Slimmy**. Inoltre, **tutti i nutrienti aggiunti dagli utenti alla lista dei nutrienti sono, per definizione, opzionali**. Si mostrano di seguito alcuni esempi di schede alimento:

Fusilli Di Chicco (formato n. 34)

- Codice identificativo: "8033829843"
- Unità di alimento: 100 grammi
- kCal per unità di alimento (ovvero per 100 grammi): 351
- Porzione standard: 0.7 unità di alimento (ovvero 70 grammi)
- Nutrienti (in grammi per 100 grammi di prodotto; *: nutrienti obbligatori):
 - Grassi totali*: 1.5 g
 - Zuccheri*: 3.4 g
 - Sale: 0 g
 - Grassi saturi*: 0.3 g
 - Fibre*: 2.9 g
 - Proteine*: 14 g
 - Carboidrati*: 69 g

Buzz Cola

- Codice identificativo: "11928322032"
- Unità di alimento: 100 ml
- kCal per unità di alimento (ovvero per 100 ml): 42
- Porzione standard: 3.3 unità di alimento (ovvero 330 ml)
- Nutrienti (in grammi per 100 ml di prodotto; *: nutrienti obbligatori):

- Grassi totali*: 0 g
- Grassi saturi*: 0 g
- Carboidrati*: 11 g
- Zuccheri*: 11 g

- Fibre*: 0 g
- Proteine*: 0 g
- Caffeina: 0.008 g
- Calcio: 0.002 g

- Fosforo: 0.01 g
- Potassio: 0.002 g
- Ferro: 0.00011 g
- Sale: 0.001 g

Come detto, schede relative a nuovi alimenti possono essere aggiunte in ogni momento dalla redazione di Slimmy, ma anche da parte degli utenti, che quindi possono contribuire in prima persona all'ampliamento dell'archivio degli alimenti di Slimmy. Mentre le schede di alimenti inserite dalla redazione diventano subito pubbliche e visibili a tutti gli utenti, le schede aggiunte da utenti devono attraversare un processo di validazione. Una volta validate diventano pubbliche esattamente come quelle inserite dalla redazione.

Una scheda alimento inserita da un utente viene resa pubblica quando almeno 10 utenti l'hanno dichiarata valida. Fintantoché una scheda non supera il processo di validazione, può ancora essere ottenuta come risultato di una ricerca, ma può essere utilizzata (ad es., inserita in un diario alimentare) solo da parte dell'utente che l'ha compilata e di quelli che l'hanno dichiarata valida fino a quel momento (in questo modo si vogliono incentivare gli utenti interessati ad inserire quell'alimento a contribuire alla validazione della relativa scheda).

Nel tempo, le schede alimenti (sia quelle inserite dalla redazione che quelle inserite dagli utenti e diventate pubbliche a seguito di validazione) possono essere invalidate (ad es., perché obsolete, si pensi al caso in cui un alimento confezionato cambi profilo nutrizionale perché prodotto con una nuova ricetta). Questo avviene quando 100 utenti le hanno dichiarate non più valide. Le schede invalidate diventano inaccessibili a tutti gli utenti (tranne che, per le schede inizialmente inserite da un utente, all'utente che le ha inserite).

Gli utenti, durante la loro giornata, inseriscono in Slimmy la composizione dei loro pasti (diario alimentare), in termini di alimenti ingeriti e relative quantità. L'insieme dei pasti è attualmente definito dalla redazione di Slimmy come: colazione, pranzo, snack e cena, ma potrebbe essere oggetto di estensioni e perfino di personalizzazioni in futuro.

Il sistema deve permettere di calcolare una serie di statistiche riguardo il diario alimentare di un utente in un certo giorno. In particolare: (i) Numero totale di kCal ingerite; (ii) grammi totali ingeriti di ogni nutriente.

Gli utenti di Slimmy possono acquistare abbonamenti Premium (la cui durata dipende dalla tipologia di abbonamento scelta tra quelle previste), durante i quali avranno accesso a servizi non offerti ai clienti Free. Il servizio principale offerto da Slimmy ai clienti Premium è relativo alla assistenza di coloro che desiderano seguire una dieta.

Le diete sono definite dai singoli utenti e sono private. Una dieta definisce delle regole, in termini di soglie minime e massime per alcune delle statistiche giornaliere di cui sopra. Ad esempio, un utente Premium che vuole perdere peso può definire una dieta, che decide di chiamare "ipocalorica ed iperproteica", con le seguenti regole circa le kCal totali e le quantità di nutrienti ingeriti al giorno:

Dieta: "ipocalorica ed iperproteica"

1. kCal totali: tra 1300 e 1500
2. Carboidrati totali: tra 150 e 180 g
3. Zuccheri: ≤ 30 g
4. Proteine totali: tra 80 e 110 g
5. Grassi saturi: ≤ 10 g
6. Sale: ≤ 3 g
7. Fibre: tra 35 e 40 g

Dato un pasto p , Slimmy deve permettere agli utenti di ottenere gli alimenti più comunemente aggiunti al diario per il pasto p da tutti gli altri utenti. Per ognuno, si vuole anche conoscere la quantità media con la quale viene aggiunta al diario.

Infine, Slimmy deve restituire, all'utente Premium che sta seguendo una dieta, le quantità di ognuno dei nutrienti che può ancora assumere (in base alle regole della dieta) in un certo giorno.



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

Esame BD2.Esame.Risposte – Modulo risposte prova scritta

Dati dello studente e dell'esame

Cognome e nome: BIANCO SIMONE Matricola: 11

Data: 21/08/23

Corso di laurea e canale di appartenenza:

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

Firma di un membro della Commissione per
avvenuta identificazione:

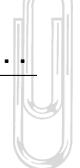
.....

Rinuncia alla prova

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



Questo modulo è ottimizzato per la stampa fronte-retro



Istruzioni e regole d'esame

Prima dell'esame

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

Durante l'esame

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

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

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

Al momento della consegna

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

In caso di rinuncia

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

Sommario delle domande

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

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

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

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

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

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

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

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

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



Questa pagina è stata intenzionalmente lasciata vuota

1 Analisi concettuale

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

Risposta

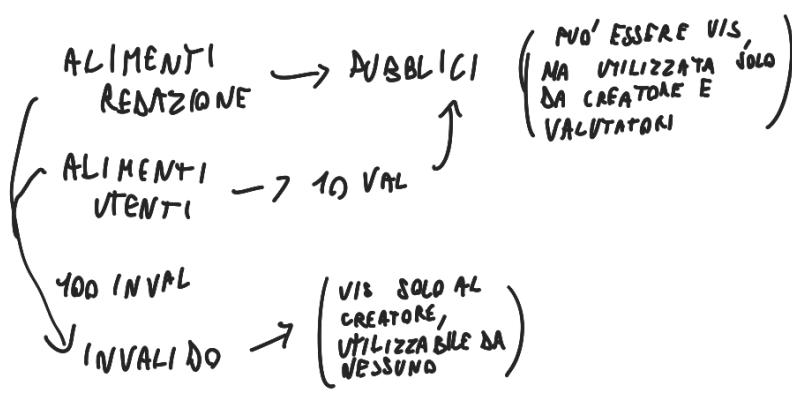
UTENTE

- NOME, COGNOME,
- NAZIONALITÀ
- EMAIL
- PASSWORD

- ALIMENTO
- CIBO O BEVANDA
- GLI UTENTI POSSONO AGGIUNDERLI
- NOME
- MARCA (0,1)
- CONFEZIONE (0,1)
- KCAL / UNITÀ'
- VAL UNITÀ'
- PORZIONE
- NUTRIENTI (GRAMMI)

- CONFEZIONE
- GRANDEZZA CONFEZIONE
- CODICE A BARRE

- OBBLIGATORIO
O OPZIONALE
- GRAMMI
- QUELLI
AGGIUNTI DA
UTENTI SONO
OPZIONALI
- ANCHE LA
RELAZIONE
AGGIUNGE GLI
OPZIONALI



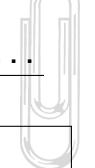
- TIPICO PASTO
- COLAZIONE
- PRANZO
- CENA
- SNACK
- ALTRO (NORE)

UTENTI PREMIUM
• DIETA CON
REGOLE

AIETA

REGOLA

- NUTRIMENTE
- MIN (0,1)
- MAX (0,9)



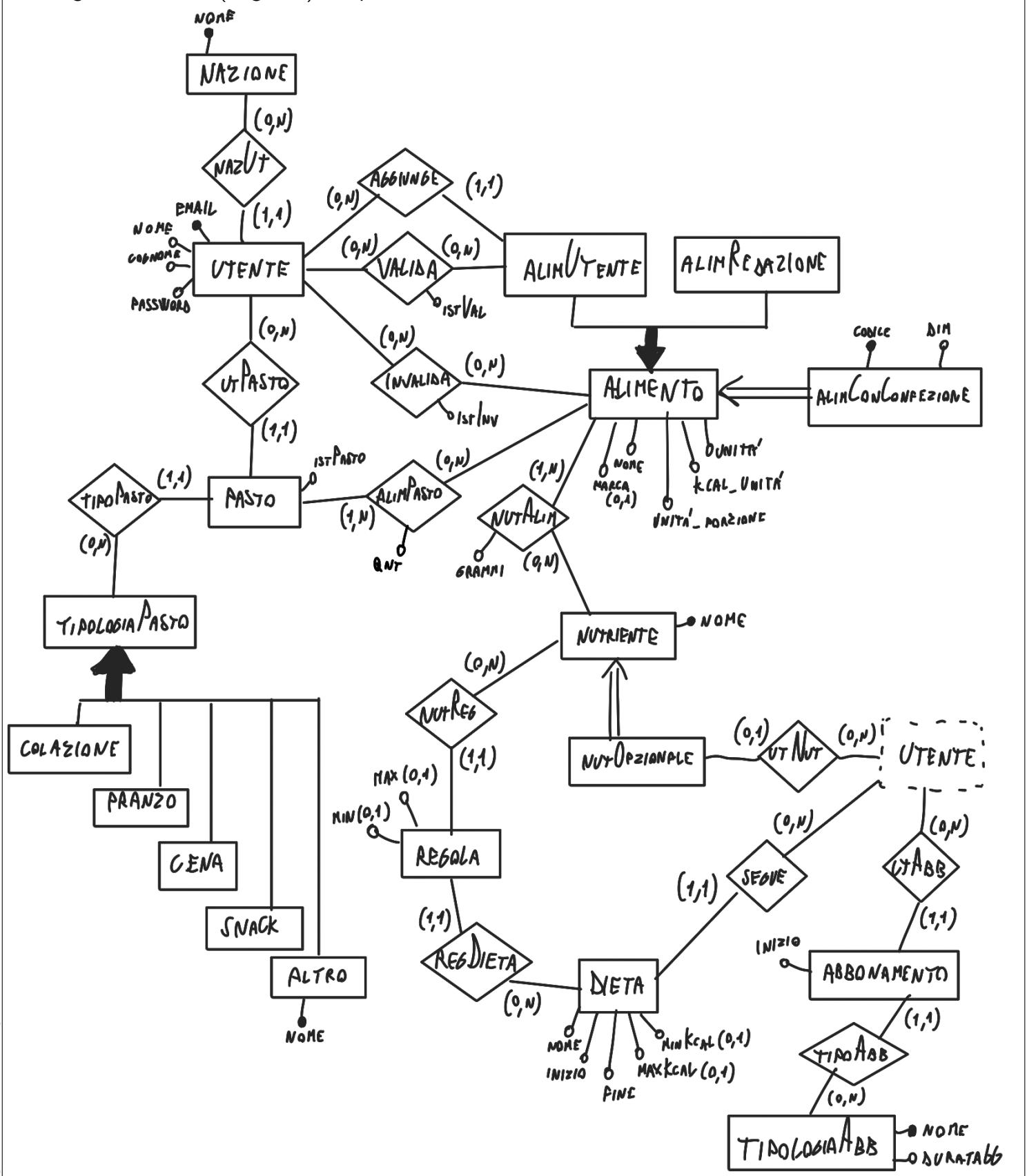
Risposta alla Domanda 1 (segue)

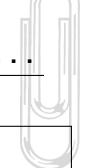
Domanda 2 (45 minuti; 75 minuti al massimo) Proseguire la fase di Analisi Concettuale dei requisiti, producendo un diagramma ER concettuale per l'applicazione, il dizionario dei dati ed eventuali vincoli esterni.

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: NAZIONE</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 style="text-align: left;">NAME</td><td style="text-align: left;">STRINGA</td><td></td></tr> </tbody> </table> <p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p>	attributo	dominio	moltep. (*)	NAME	STRINGA		<p>[2] Tipo: Entità Relationship (cerchiare)</p> <p>Nome: UTENTE</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 style="text-align: left;">EMAIL</td><td style="text-align: left;">EMAIL</td><td></td></tr> <tr> <td style="text-align: left;">NAME</td><td style="text-align: left;">STRINGA</td><td></td></tr> <tr> <td style="text-align: left;">CODNAME</td><td style="text-align: left;">STRINGA</td><td></td></tr> <tr> <td style="text-align: left;">PASSWORD</td><td style="text-align: left;">STRINGA</td><td></td></tr> </tbody> </table> <p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p>	attributo	dominio	moltep. (*)	EMAIL	EMAIL		NAME	STRINGA		CODNAME	STRINGA		PASSWORD	STRINGA	
attributo	dominio	moltep. (*)																				
NAME	STRINGA																					
attributo	dominio	moltep. (*)																				
EMAIL	EMAIL																					
NAME	STRINGA																					
CODNAME	STRINGA																					
PASSWORD	STRINGA																					

<input type="checkbox"/> 3	Tipo: Entità	Relationship (cerchiare)
Nome:	PASTO	
attributo	dominio	moltep. (*)
<i>IST_PASTO</i>	<i>DATAORA</i>	
(*) solo se diversa da (1,1)		
Vincoli:		
<ul style="list-style-type: none"> • V. PASTO. ALIM/VALIDO $\forall a, p, i, u \text{ ALIMENTO}(a) \wedge \text{PASTO}(a) \wedge \text{ALIM_PASTO}(p, a) \wedge \\ \text{IST_PASTO}(a, i) \wedge \text{UT_PASTO}(a, u) \rightarrow \text{CREA}(u, a) \vee \text{VALIDA}(u, a) \\ \vee \left\{ \begin{array}{l} u' \mid \text{UTENTE}(u') \wedge \text{VALIDA}(u', a) \wedge \\ (\exists i' \text{ ISTVAL}(a, i') \wedge i' \leq i) \end{array} \right\} \geq 10$ <ul style="list-style-type: none"> • V. PASTO. ALIM/INVALIDO $\forall a, p, i, u \text{ ALIMENTO}(a) \wedge \text{PASTO}(a) \wedge \text{ALIM_PASTO}(p, a) \wedge \\ \text{IST_PASTO}(a, i) \wedge \text{UT_PASTO}(a, u) \rightarrow \text{CREA}(u, a) \vee \\ \left\{ \begin{array}{l} u' \mid \text{UTENTE}(u') \wedge \text{IN_VALIDA}(u', a) \wedge \\ (\exists i' \text{ ISTINV}(a, i') \wedge i' \leq i) \end{array} \right\} < 100$		

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

<input type="checkbox"/> 4	Tipo: Entità	Relationship (cerchiare)
Nome:	ALIMENTO	
attributo	dominio	moltep. (*)
NAME	STRINGA	
MARCA	STRINGA	(a, i)
UNITA'	UNITA'	
KCAL_UNITA'	REALIZZO	
UNITA'_PORZIONE	INTEROIZZO	
(*) solo se diversa da (1,1)		
Vincoli:		

<input type="checkbox"/> 6	Tipo: Entità	Relationship (cerchiare)
Nome:	VALIDA	
attributo	dominio	moltep. (*)
<i>IST_VAL</i>	<i>DATAORA</i>	
(*) solo se diversa da (1,1)		
Vincoli:		
<ul style="list-style-type: none"> • V. VALIDA. CREATORE R $\forall a, u \text{ ALIMENTO}(a) \wedge \text{VALIDA}(u, a) \rightarrow \\ \exists \text{AGGIUNTE}(u, a)$		

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

Nome: INVALIDA

attributo	dominio	moltep. (*)
<u>IST/IV</u>	<u>DATI/MORA</u>	

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

Vincoli:

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

Nome: ALIM CON CONFEZIONE

attributo	dominio	moltep. (*)
<u>CODICE</u>	<u>STRINGA</u>	
<u>BN</u>	<u>DIMENSIONI</u>	

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

Vincoli:

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

Nome: NUTRIENTE

attributo	dominio	moltep. (*)
<u>NAME</u>	<u>STRINGA</u>	

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

Vincoli:

- V. NUTRIENTE OBBLIGATORIO A

H_n, n ALIMENTO(z) \ NUTRIENTE(z) \
\ NUTRIZIONALE(n) -> NUTALIN(z,n)

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

Nome: NUTALIM

attributo	dominio	moltep. (*)
<u>GRAMMI</u>	<u>REALIZZO</u>	

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

Vincoli:

<p>11 Tipo: Entità Relationship (cerchiare)</p> <p>Nome: ... <u>REGOLA</u></p> <table border="1" data-bbox="95 235 786 303"> <thead> <tr> <th>attributo</th><th>dominio</th><th>moltep. (*)</th></tr> </thead> <tbody> <tr> <td><i>MIN</i></td><td>REALE$\exists\forall$</td><td>(0,1)</td></tr> <tr> <td><i>MAX</i></td><td>REALE$\exists\forall$</td><td>(0,1)</td></tr> </tbody> </table>	attributo	dominio	moltep. (*)	<i>MIN</i>	REALE $\exists\forall$	(0,1)	<i>MAX</i>	REALE $\exists\forall$	(0,1)	<p>13 Tipo: Entità Relationship (cerchiare)</p> <p>Nome: ... <u>DIETA</u></p> <table border="1" data-bbox="786 235 1511 303"> <thead> <tr> <th>attributo</th><th>dominio</th><th>moltep. (*)</th></tr> </thead> <tbody> <tr> <td><i>NOME</i></td><td>STRINGA</td><td></td></tr> <tr> <td><i>MINKCAL</i></td><td>REALE$\exists\forall$</td><td>(0,1)</td></tr> <tr> <td><i>MAXKCAL</i></td><td>REALE$\exists\forall$</td><td>(0,1)</td></tr> <tr> <td><i>INIZIO</i></td><td>DATAORA</td><td></td></tr> <tr> <td><i>FINE</i></td><td>DATAORA</td><td></td></tr> </tbody> </table>	attributo	dominio	moltep. (*)	<i>NOME</i>	STRINGA		<i>MINKCAL</i>	REALE $\exists\forall$	(0,1)	<i>MAXKCAL</i>	REALE $\exists\forall$	(0,1)	<i>INIZIO</i>	DATAORA		<i>FINE</i>	DATAORA	
attributo	dominio	moltep. (*)																										
<i>MIN</i>	REALE $\exists\forall$	(0,1)																										
<i>MAX</i>	REALE $\exists\forall$	(0,1)																										
attributo	dominio	moltep. (*)																										
<i>NOME</i>	STRINGA																											
<i>MINKCAL</i>	REALE $\exists\forall$	(0,1)																										
<i>MAXKCAL</i>	REALE $\exists\forall$	(0,1)																										
<i>INIZIO</i>	DATAORA																											
<i>FINE</i>	DATAORA																											
<p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p> <ul style="list-style-type: none"> ✗ • V. REGOLA. INTERVALLO $\forall \tau \text{ REGOLA}(\tau) \rightarrow (\forall m \text{ MIN}(\tau, m) \rightarrow (\forall n' \text{ MAX}(\tau, n') \rightarrow m < n'))$ • V. REGOLA. NUTRIENTE UNICO $\forall d, n, x, x' \text{ DIETA}(d) \wedge \text{REGDIETA}(d, x) \wedge \text{REGDIETA}(d, x') \wedge \text{NUTREG}(x, n) \wedge \text{NUTREG}(x', n) \rightarrow x = x'$ 	<p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p> <ul style="list-style-type: none"> ✗ • V. DIETA. INTERVALLO $\forall d \text{ DIETA}(d) \rightarrow (\forall m \text{ MINKCAL}(d, m) \rightarrow (\forall n' \text{ MAXKCAL}(d, n') \rightarrow m < n'))$ ✗ • V. DIETA. CONTINUITÀ $\forall d, i, f \text{ DIETA}(d) \wedge \text{INIZIO}(d, i) \wedge \text{FINE}(d, f) \rightarrow i < f$ 																											

<p>12 Tipo: Entità Relationship (cerchiare)</p> <p>Nome: ... <u>ABBONAMENTO</u></p> <table border="1" data-bbox="85 1291 786 1358"> <thead> <tr> <th>attributo</th><th>dominio</th><th>moltep. (*)</th></tr> </thead> <tbody> <tr> <td><i>INIZIO</i></td><td>DATAORA</td><td></td></tr> </tbody> </table>	attributo	dominio	moltep. (*)	<i>INIZIO</i>	DATAORA		<p>14 Tipo: Entità Relationship (cerchiare)</p> <p>Nome: ... <u>TIPOLOGIA ABB</u></p> <table border="1" data-bbox="786 1291 1511 1358"> <thead> <tr> <th>attributo</th><th>dominio</th><th>moltep. (*)</th></tr> </thead> <tbody> <tr> <td><i>NOME</i></td><td>STRINGA</td><td></td></tr> <tr> <td><i>DURATAABB</i></td><td>INTERO$\exists\forall$</td><td></td></tr> </tbody> </table>	attributo	dominio	moltep. (*)	<i>NOME</i>	STRINGA		<i>DURATAABB</i>	INTERO $\exists\forall$	
attributo	dominio	moltep. (*)														
<i>INIZIO</i>	DATAORA															
attributo	dominio	moltep. (*)														
<i>NOME</i>	STRINGA															
<i>DURATAABB</i>	INTERO $\exists\forall$															
<p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p> <ul style="list-style-type: none"> • V. ABBONAMENTO. DISBUONTO $\forall u, a, a', i, d, d' \text{ UTENTE}(u) \wedge \text{UTABO}(a, u) \wedge \text{UTABO}(a', u) \wedge a \neq a' \wedge \text{INIZIO}(a, i) \wedge \text{INIZIO}(a', i') \wedge \text{DURATAABB}(a, d) \wedge \text{DURATAABB}(a', d') \rightarrow \nexists t \text{ DATAORA}(t) \wedge (i \leq t \wedge t \leq i' \wedge a \neq a') \wedge (i \leq t \wedge t \leq i' \wedge a = a')$ 	<p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p>															

<p>15 Tipo: <input checked="" type="checkbox"/> Entità Relationship (cerchiare)</p> <p>Nome: <u>ALIMENTO</u></p> <table border="1" data-bbox="103 235 790 303"> <thead> <tr> <th>attributo</th> <th>dominio</th> <th>moltep. (*)</th> </tr> </thead> <tbody> <tr> <td><u>QNT</u></td> <td><u>INTERO/P</u></td> <td></td> </tr> </tbody> </table> <hr/> <p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p>	attributo	dominio	moltep. (*)	<u>QNT</u>	<u>INTERO/P</u>		<p>17 Tipo: <input checked="" type="checkbox"/> Entità Relationship (cerchiare)</p> <p>Nome:</p> <table border="1" data-bbox="817 235 1518 303"> <thead> <tr> <th>attributo</th> <th>dominio</th> <th>moltep. (*)</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> </tr> </tbody> </table> <hr/> <p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p>	attributo	dominio	moltep. (*)			
attributo	dominio	moltep. (*)											
<u>QNT</u>	<u>INTERO/P</u>												
attributo	dominio	moltep. (*)											
<p>16 Tipo: <input checked="" type="checkbox"/> Entità Relationship (cerchiare)</p> <p>Nome:</p> <table border="1" data-bbox="103 1224 790 1291"> <thead> <tr> <th>attributo</th> <th>dominio</th> <th>moltep. (*)</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> </tr> </tbody> </table> <hr/> <p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p>	attributo	dominio	moltep. (*)				<p>18 Tipo: <input checked="" type="checkbox"/> Entità Relationship (cerchiare)</p> <p>Nome:</p> <table border="1" data-bbox="817 1224 1518 1291"> <thead> <tr> <th>attributo</th> <th>dominio</th> <th>moltep. (*)</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> </tr> </tbody> </table> <hr/> <p>(*) solo se diversa da (1,1)</p> <p>Vincoli:</p>	attributo	dominio	moltep. (*)			
attributo	dominio	moltep. (*)											
attributo	dominio	moltep. (*)											

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

- DOMINIO EMAIL : STRINGA SECONDO STANDARD

- DOMINIO UNITÀ:

Tipo: STRINGA

QNT: REALE > 0

- DOMINIO DIMENSIONI:

LUNG: REALE > 0

LANG: REALE > 0

ALT: REALE > 0

- V. DIETA. DISGIVINTA

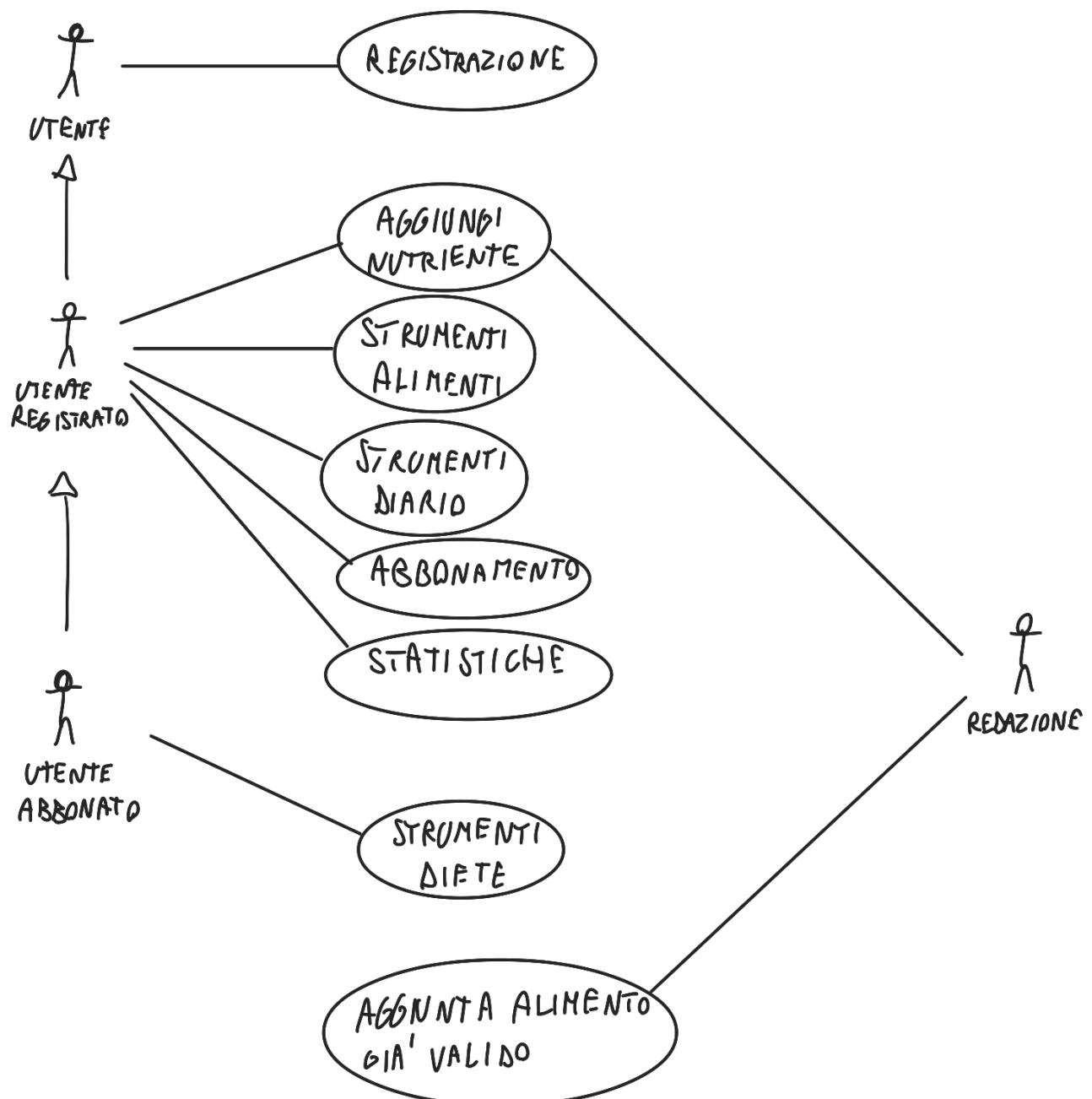
$$\forall u, d, d', i, i', f, f' \quad \text{UTENTE}(u) \wedge \text{SEGUE}(u, d) \wedge \text{SEGUE}(u, d') \wedge d \neq d' \wedge \text{INIZIO}(d, i) \wedge \text{INIZIO}(d', i') \\ \wedge \text{FINE}(d, f) \wedge \text{FINE}(d', f') \rightarrow \exists t \quad \text{DATAORA}(t) \wedge (i \leq t \wedge t \leq f) \wedge (i' \leq t \wedge t \leq f')$$

- V. DIETA. ABDONAMENTO

$$\forall u, d, a, i, i', f, f' \quad \text{UTENTE}(u) \wedge \text{SEGUE}(u, d) \wedge \text{UTABA}(a, u) \wedge \text{INIZIO}(d, i) \wedge \text{INIZIO}(d', i') \wedge \text{FINE}(d, f) \wedge \\ \text{DURATA66}(a, d') \rightarrow i \leq i' \wedge f \leq (i + d' - 1) \text{ GIORNO}'$$

Risposta alla Domanda 2 (segue)

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

Risposta

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

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

1 Specifica use-case:REGISTRAZIONE..... (nome use-case)

Operazioni dello use-case:

REGISTRA UTENTE ($n: \text{STRINGA}$, $c: \text{STRINGA}$, $e: \text{EMAIL}$, $p: \text{STRINGA}$, $n': \text{NAZIONE}$): UTENTE

2 Specifica use-case: ...AGGIUNGI... NUTRIENTE..... (nome use-case)

Operazioni dello use-case:

INSERISCI NUTRIENTE ($n: \text{STRINGA}$): NUTRIENTE OPZIONALE

3 Specifica use-case:STRUMENTI... ALIMENTI..... (nome use-case)

Operazioni dello use-case:

INSERISCI ALIMENTO DA VALIDARE ($u: \text{UTENTE}$, $N: \text{NUTRIENTE} (1, N)$, $n: \text{STRINGA}$, $m: \text{MARCA} (0, +)$, $u': \text{UNITA}'$, $k: \text{REALIZZO}$, $p: \text{REALIZZO}$, $c: \text{STRINGA} (0, 1)$, $d: \text{DIMENSIONI}$): ALIMENTO UTENTE

VALIDA ALIMENTO ($u: \text{UTENTE}$, $a: \text{ALIMENTO}$)

INVALIDA ALIMENTO ($u: \text{UTENTE}$, $a: \text{ALIMENTO}$)

RICERCA ALIMENTO ($n: \text{STRINGA}$, $m: \text{STRINGA} (0, 1)$): ALIMENTO ($0, N$)

4 Specifica use-case:STRUMENTI DIARIO..... (nome use-case)

Operazioni dello use-case:

INSERISCI PASTO ($u: \text{UTENTE}, t: \text{TIPOLOGIA PASTO}, A: \text{ALIMENTO } (1, N), t': \text{DATA ORA} \rangle: \text{PASTO}$)

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

Operazioni dello use-case:

ABBONATA ($u: \text{UTENTE}, t: \text{TIPOLOGIA ABBONAMENTO} \rangle: \text{ABBONAMENTO}$)

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

Operazioni dello use-case:

CALCOLATOKCAL ($u: \text{UTENTE}, d: \text{DATA} \rangle: \text{REAL } 2,0$)
 X CALCOLTOTGAMMINUTRIENTE ($u: \text{UTENTE}, n: \text{NUTRIENTE}, d: \text{DATA} \rangle: \text{REAL } 2,0$)
 ALIMENTI PIU' CONSUMATI ALTRI UTENTI ($u: \text{UTENTE}, t: \text{TIPOLOGIA PASTO} \rangle: \text{ALIMENTO}$)
 QUANTITA' MEDIA ALIMENTO ($a: \text{ALIMENTO} \rangle: \text{REAL } 2,0$)

7 Specifica use-case:STRUMENTI DIETE..... (nome use-case)

Operazioni dello use-case:

CREAREGOLA ($n: \text{NUTRIENTE}, \min: \text{REAL } 2,0 (0,1), \max: \text{REAL } 2,0 (0,1) \rangle: \text{REGOLA}$)
 CREA DIETA ($u: \text{UTENTE}, R: \text{REGOLA } (1, N) \rangle: \text{DIETA}$)
 NUTRIENTI ASSORBIBILI ($u: \text{UTENTE}, d: \text{DATA} \rangle: (n: \text{NUTRIENTE}, g: \text{REAL } 2,0) (0, N)$)

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

ALIMENTI PIU' CONSUMATI ALTRI UTENTI ($u: \text{UTENTE}, t: \text{TIPOLOGIA PASTO}$): $\text{AUMENTO}(0, n)$

PRE:

POST:

$$A = \left\{ (a, n) \mid \begin{array}{l} \text{ALIMENTO}(a) \wedge \\ n = \left| \left\{ p \mid \text{MASTRO}(p) \wedge \text{TIPOLOGIA PASTO}(p, t) \wedge \text{ALIMENTO}(p, a) \wedge \right. \right. \\ \left. \left. (\exists u' \text{ UTENTI}(p, u') \wedge u \neq u') \right\} \right| \end{array} \right\}$$

$$A_{\max} = \underset{(a, n) \in A}{\text{ARGMAX}}(n)$$

$$\text{RESULT} = \left\{ a' \mid \exists n' (a', n') \in A_{\max} \right\}$$

NUTRIENTI ASSUMIBILI ($u: \text{UTENTE}, d: \text{DATA}$): $(n: \text{NUTRIENTE}, g: \text{REALIZZO}) (0, n)$

PRE: $\exists a, i, f, di, df$

$$\begin{aligned} &\text{ABBONAMENTO}(a) \wedge \text{UTENTE}(a, u) \wedge \text{INIZIO}(a, i') \wedge \text{FINE}(a, f') \wedge \\ &\text{DATA}(i, di) \wedge \text{DATA}(f, df) \wedge di \leq d \wedge d \leq df \end{aligned}$$

POST:

$$N = \left\{ (n, t) \mid \begin{array}{l} \text{NUTRIENTE}(n) \wedge (\exists dt, i, f, m, g \text{ DIETA}(dt) \wedge \text{INIZIO}(dt, i) \wedge \text{FINE}(dt, f)) \\ \wedge (i \leq d \wedge d \leq f) \wedge \text{RES DIETA}(dt, n) \wedge \text{NUTREG}(n, n) \wedge \text{MAX}(n, m) \wedge \\ g = \text{CALCOLATORGRAMMI NUTRIENTE}(n, n, d) \wedge t = m - g \wedge t \geq 0 \end{array} \right\}$$

$$\text{RESULT} = N$$

CALCOLATORGRAMMI NUTRIENTE ($u: \text{UTENTE}, n: \text{NUTRIENTE}, d: \text{DATA}$): REALIZZO

PRE:

POST:

$$A = \left\{ (p, a, g) \mid \begin{array}{l} \text{MASTRO}(p) \wedge \text{UTENTI}(p, a) \wedge \text{ALIMENTO}(a) \wedge \text{ALIMENTO}(p, a) \wedge \text{NUTRIMENTO}(a, n) \\ \wedge (\exists g', q \text{ GRAMMI}(a, n, g') \wedge \text{QNT}(p, a, q) \wedge g = g' \cdot q) \end{array} \right\}$$

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

Risposta alla Domanda 5 (segue)

QUANTITA' MEDIA ALIMENTO (2: ALIMENTO) : REALE>0

PREF:

POST:

$$P = \left\{ (p, q) \mid \begin{array}{l} \text{PASTO}(p) \wedge \text{ALIM}(p, a) \\ \wedge \text{QNT}(p, a, q) \end{array} \right\}$$

$$(P = \emptyset \rightarrow \text{RESULT} = 0)$$

$$\left(P \neq \emptyset \rightarrow \text{RESULT} = \frac{\sum q}{|P|} \right)$$

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.

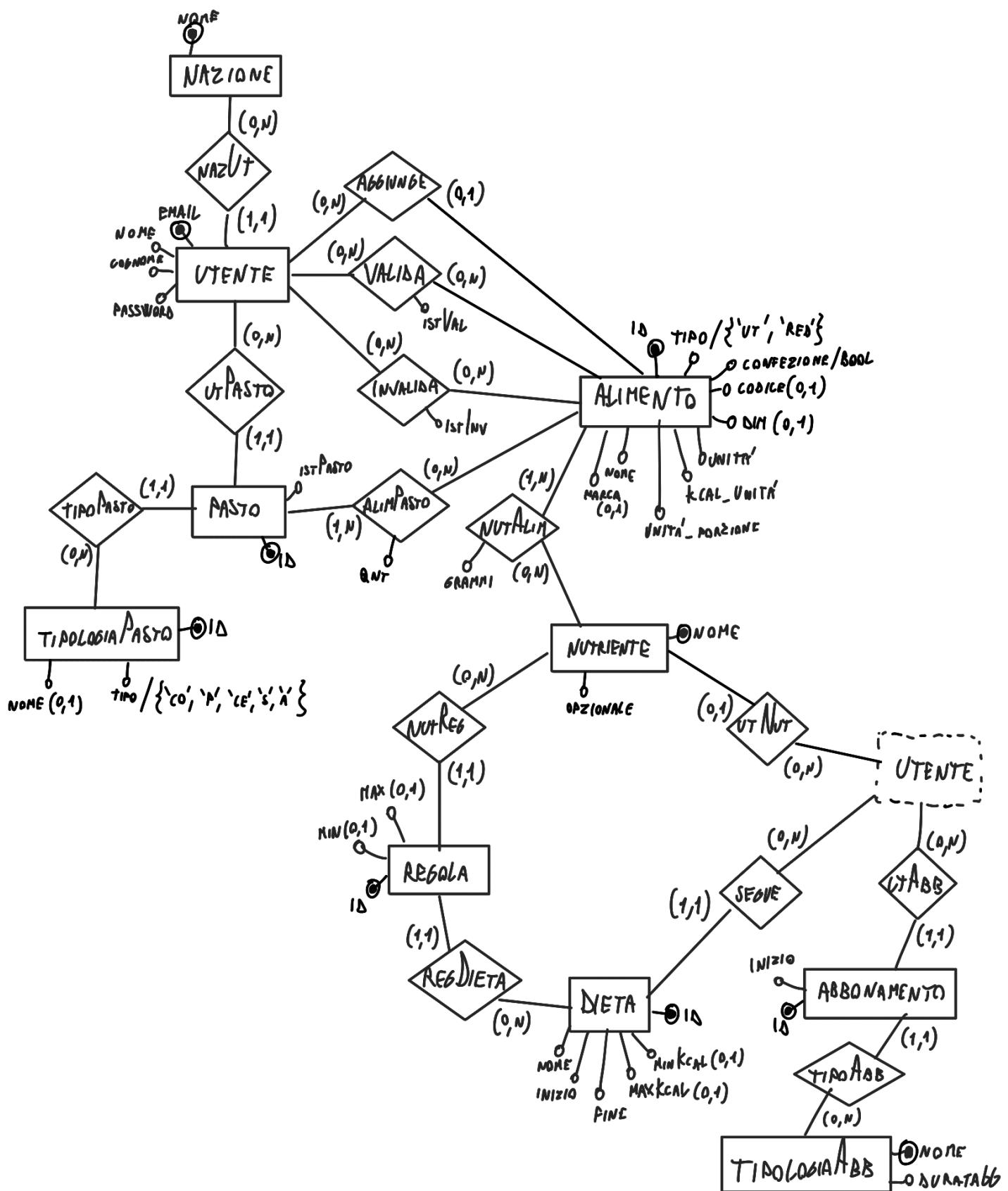
Descrivere brevemente le principali scelte effettuate.

DBMS da utilizzare **POSTGRESQL**

Corrispondenza tra domini concettuali e domini supportati dal DBMS

- `CREATE DOMAIN INT-GZ AS INTEGER CHECK(VALUE >= 0)`
- `CREATE DOMAIN INT-GEZ AS INTEGER CHECK(VALUE >= 0)`
- `CREATE DOMAIN REAL-GZ AS REAL CHECK(VALUE >= 0)`
- `CREATE DOMAIN REAL-GEZ AS REAL CHECK(VALUE >= 0)`
- `CREATE DOMAIN STR AS VARCHAR(200)`
- `CREATE DOMAIN EMAIL AS STR CHECK (isValidEmail(VALUE))`
- `CREATE TYPE UNITA AS (`
 `TIPO STR,`
 `QNT REAL-6Z`
`)`
- `CREATE TYPE DIMENSIONI (`
 `LUNG REAL-6E2`
 `LARG REAL-6E2,`
 `ALT REAL-6E2`
`)`
- `CREATE TYPE TIPO-PASTO AS ENUM('co','n','le','s','a')`
- `CREATE TYPE TIPO-ALIM AS ENUM('vt','red')`

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)

X • V. ALIMENTO. CONFEZIONE

$$\forall a \text{ ALIMENTO}(a) \rightarrow (\exists c, d \text{ codice}(a, c) \wedge \Delta IM(a, d) \Leftrightarrow \text{CONFEZIONE}(a, \text{TRUE}))$$

X • V. TIPOLOGIA PASTO, ALTRO

$$\forall t \text{ TIPOLOGIA PASTO}(t) \rightarrow (\exists n \text{ NONE}(t, n) \Leftrightarrow \text{TIPO}(t, 'n'))$$

X • V. ALIMENTO. AGGIUNGE

$$\forall a \text{ ALIMENTO}(a) \rightarrow (\exists u \text{ AGGIUNGE}(u, a) \Leftrightarrow \text{TIPO}(a, 'ut'))$$

• V. ALIMENTO. VALIDA

$$\forall u, a \text{ UTENTE}(u) \wedge \text{ALIMENTO}(a) \wedge \text{VALIDA}(u, a) \rightarrow \text{TIPO}(a, 'ut')$$

• V. NUTRIENTE. OBBLIGATORIO

$$\forall a, n \text{ ALIMENTO}(a) \wedge \text{NUTRIENTE}(n) \wedge \text{OPZIONALE}(n, \text{FALSE}) \rightarrow \text{NUTALIN}(a, n)$$

Risposta alla Domanda 6 (segue)

• V. PASTO. ALIM/VALIDO

$$\forall a, p, i, u \text{ ALIMENTO}(a) \wedge \text{TIPO}(a, 'UT') \wedge \text{PASTO}(p) \wedge \text{ALIM_PASTO}(p, a) \wedge \text{IST_PASTO}(p, i) \wedge \text{UT_PASTO}(p, u) \rightarrow$$

$$\text{CREA}(u, a) \vee \text{VALIDA}(u, a) \vee \left| \left\{ u' \mid \begin{array}{l} \text{UTENTE}(u') \wedge \text{VALIDA}(u', a) \\ (\exists i' \text{ ISVAL}(a, i') \wedge i' < i) \end{array} \right\} \right| \geq 10$$

• V. VALIDA_CREATORE

$$\forall a, u \text{ ALIMENTO}(a) \wedge \text{TIPO}(a, 'UT') \wedge \text{VALIDA}(u, a) \rightarrow \exists \text{AGGIUNDE}(u, a)$$

T • U. TIPOLOGIA_PASTO. NOME (HIAVE)

$$\forall t, t', n, n' \text{ TIPOLOGIA_PASTO}(t) \wedge \text{TIPOLOGIA_PASTO}(t') \wedge \text{TIPO}(t, 'A') \wedge \text{TIPO}(t', 'A') \wedge t \neq t' \wedge$$

$$\text{NOME}(t, n) \wedge \text{NOME}(t', n') \rightarrow n \neq n'$$

• U. ALIMENTO. CODICE (HIAVE)

$$\forall a, a', c, c' \text{ ALIMENTO}(a) \wedge \text{ALIMENTO}(a') \wedge \text{CONFEZIONE}(a, \text{TRUE}) \wedge \text{CONFEZIONE}(a', \text{TRUE}) \wedge a \neq a' \wedge$$

$$\text{CODICE}(a, c) \wedge \text{CODICE}(a', c') \rightarrow c \neq c'$$

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 STR	

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

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

La relazione accorda le relazioni che implementano le seguenti relationship:

2 Relazione ...UTENTE..... (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>EMAIL</u> NOME COGNOME PASSWORD NAZ	
Domini EMAIL STR STR STR STR	

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

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

FK: (NAZ) REF NAZIONE(NOME)

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

3 Relazione ...VALIDA..... (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>UTENTE</u> ALIM ISVAL	
Domini EMAIL INTEGER TIMESTAMP	

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

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

FK: (UTENTE) REF UTENTE(EMAIL)

FK: (ALIM) REF ALIMENTO(ID)

La relazione accorda le relazioni che implementano le seguenti relationship:

4 Relazione ...INVALIDA..... (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>UTENTE</u> ALIM ISVAL	
Domini EMAIL INTEGER TIMESTAMP	

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

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

FK: (UTENTE) REF UTENTE(EMAIL)

FK: (ALIM) REF ALIMENTO(ID)

La relazione accorda le relazioni che implementano le seguenti relationship:

5 Relazione ...ALIMENTO..... (nome)	Derivante da: entità relationship (cerchiare)
Attributi ID TIPO MANCA * UNITA KCAL-UNITA UNTA'-PORZIONE UTENTE * CONFEZIONE	
Domini INTEGER TIPO_ALIN STR UNITA REAL_6Z INT_6Z EMAIL BOOLEAN	

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

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

SERIAL: ID

FK: (UTENTE) REF UTENTE(EMAIL)

FUN: UTENTE ≠ NULL ⇔ TIPO = 'UT'

BN: CONFEZIONE = TRUE ⇔ ID ≠ NULL & CONFEZIONE ≠ NULL

CONFEZIONE * | CONFEZIONE *

DIMENSIONI | STR

La relazione accorda le relazioni che implementano le seguenti relationship: ... **ABBI_UTNCE**

6 Relazione ... PASTO (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>ID</u> <u>LISTPASTO</u> <u>TIPO</u> <u>UTENTE</u>	
Domini <u>INTEGER</u> <u>TIMESTAMPA</u> <u>INTEGER</u> <u>EMAIL</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):

SERIAL:ID
FK: (TIPO) REF TIPOLOGIA_PASTO(ID)
FK: (UTENTE) REF UTENTE(EMAIL)

La relazione accorda le relazioni che implementano le seguenti relationship: UTPASTO, TIPO_PASTO

7 Relazione ... TIPOLOGIA_PASTO (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>ID</u> <u>TIPO</u> <u>NOME</u> *	
Domini <u>INTEGER</u> <u>TIPOMASTO</u> <u>STR</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):

SERIAL:ID
ENN: TIPO = 'A' \Leftrightarrow NOME \neq NULL

La relazione accorda le relazioni che implementano le seguenti relationship:

8 Relazione ... NUT_ALIM (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>ALIM</u> <u>NUT</u> <u>GRAMMI</u>	
Domini <u>INTEGER</u> <u>STR</u> <u>REAL -62</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: (ALIM) REF ALIMENTO(ID)
FK: (NUT) REF NUTRIENTE(NOME)

La relazione accorda le relazioni che implementano le seguenti relationship:

9 Relazione ... NUTRIENTE (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>NOME</u> <u>OPZIONALE</u> <u>UTENTE</u> *	
Domini <u>STR</u> <u>BOOLEAN</u> <u>EMAIL</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) REF UTENTE(EMAIL)

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

10 Relazione ... REGOLA (nome)	Derivante da: entità relationship (cerchiare)
Attributi <u>ID</u> <u>NUT</u> <u>MIN</u> * <u>MAX</u> * <u>DIETA</u>	
Domini <u>INTEGER</u> <u>STR</u> <u>REAL -62</u> <u>REAL -62</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):

SERIAL:ID
ENN: MIN \neq NULL \wedge MAX \neq NULL \rightarrow MIN < MAX
PK: (NUT) REF NUTRIENTE(NOME) FK: (DIETA) REF DIETA(ID)

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

[11] Relazione ...DIETA.....(nome) Derivante da: **entità** | relationship (cerchiare)

Attributi	<u>ID</u>	NOME	INIZIO	FINE	MIN-KCAL*	MAX-KCAL*	UTENTE	
Domini	INTEGER	STR	TIMESTAMP	TIMESTAMP	REAL-GZ	REAL-GZ	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):

FK: (UTENTE) REF UTENTE(EMAIL)
 ENN: MIN-KCAL ≠ NULL ∧ MAX-KCAL ≠ NULL → MIN-KCAL < MAX-KCAL
 ENN: INIZIO < FINE SERIAL:ID

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

[12] Relazione .ABBRONAMENTO ..(nome) Derivante da: **entità** | relationship (cerchiare)

Attributi	<u>ID</u>	INIZIO	UTENTE	TIPO				
Domini	INTEGER	TIMESTAMP	EMAIL	STR				

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

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

FK: (UTENTE) REF UTENTE(EMAIL) SERIAL:ID
 FK: (TIPO) REF TIPOABB(NOME)

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

[13] Relazione TIPOABBRONAMENTO (nome) Derivante da: **entità** | relationship (cerchiare)

Attributi	<u>NOME</u>	DURATA bb						
Domini	STR	INT-GZ						

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 ..ALIMPASTO....(nome) Derivante da: **entità** | relationship (cerchiare)

Attributi	<u>PASTO</u>	ALIM	QNT					
Domini	INTEGER	INTEGER	INT-GZ					

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: (PASTO) REF PASTO(ID)
 FK: (ALIM) REF ALIMENTO(ID)

La relazione accorda le relazioni che implementano le seguenti relationship:

[15] Relazione(nome) Derivante da: **entità** | relationship (cerchiare)

Attributi								
Domini								

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

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

La relazione accorda le relazioni che implementano le seguenti relationship:

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

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

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

La relazione accorda le relazioni che implementano le seguenti relationship:

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

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

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

La relazione accorda le relazioni che implementano le seguenti relationship:

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

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

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

La relazione accorda le relazioni che implementano le seguenti relationship:

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

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

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

La relazione accorda le relazioni che implementano le seguenti relationship:

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

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

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

La relazione accorda le relazioni che implementano le seguenti relationship:

Ulteriori vincoli esterni

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

- T. DIETA.ABBONAMENTO

- INSERIMENTO IN DIETA
- PRE-OPERAZIONE

$$\text{ISVALID} = \left(\begin{array}{l} \text{SELECT *} \\ \text{FROM ABBONAMENTO } a, \text{TIPOTABB } t \\ \text{WHERE } a.\text{UTENTE} = \text{NEW.UTENTE} \text{ AND } a.\text{TIPO} = t.\text{ID} \\ a.\text{INIZIO} \leq \text{NEW.INIZIO} \text{ AND} \\ \text{NEW.FINE} \leq (a.\text{INIZIO} + t.\text{DURATABB} \cdot '1 DAY') \end{array} \right)$$

```

IF ISVALID
    COMMIT
ELSE
    GENERA ERRORE
    REVERT

```

- T. TIPOLOGIA_PASTO. NONE (HIAVE)

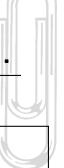
- INSERIMENTO IN TIPOLOGIA_PASTO
- PRE-OPERAZIONE

$$\text{ISVALID} = \left(\begin{array}{l} \text{NOT EXISTS} \left(\begin{array}{l} \text{SELECT *} \\ \text{FROM TIPOLOGIA_PASTO } t \\ \text{WHERE } t.\text{TIPO} = 'A' \text{ AND NEW.TIPO} = 'A' \\ \text{AND } t.\text{NONE} = \text{NEW.NONE} \end{array} \right) \end{array} \right)$$

```

IF ISVALID
    COMMIT
ELSE
    GENERA ERRORE
    REVERT

```



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

ALIMENTI PIU' CONSUMATI ALTRI UTENTI, ($u: \text{EMAIL}$, $t: \text{INTEGER}$): INSIEME ($<2: \text{INTEGER}>$)

TMP TABLE ($a: \text{INTEGER}$, $n: \text{INT_6EZ}$)

PK: (a) REF ALIMENTO (ID)

INSERT INTO TMP TABLE (a, n)

(SELECT $a.ID, COUNT(p.ID)$

FROM ALIMENTO a, PASTO p, ALIMPASTO ap

WHERE $a.ID = ap.ID$ AND $ap.ALIN = a.ID$

AND $p.TIPO = :t$ AND $p.UTENTE <?> u$

GROUP BY $a.ID$)

$Q = (\text{SELECT MAX}(n) \text{ AS MAX FROM TMP TABLE})$

$Q' = (\text{SELECT } a \text{ FROM TMP TABLE WHERE } n = Q.\text{MAX})$

RETURN Q

QUANTITA' MEDIA ALIMENTO ($a: \text{INTEGER}$): REAL-6EZ

$Q = (\text{SELECT AVG}(ap.QNT) \text{ AS M}$

FROM ALIMPASTO ap

WHERE ap.ALIN = :a)

RETURN $Q.M$

Risposta alla Domanda 8 (segue)

NUTRIENTI ASSUMIBILI ($u:EMAIL, d: DATE$): INSIEME($\langle n:INTEGER, g:REAL-6EZ \rangle$)

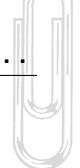
$IS_ERROR = (\text{NOT EXISTS}(\text{SELECT } *$
 $\text{FROM } ABONAMENTO } a$
 $\text{WHERE } a.\text{UTENTE} = :u$
 $\text{AND } \text{EXTRACT}(\text{DATE FROM } a.\text{INIZIO}) \leq d$
 $\text{AND } \text{EXTRACT}(\text{DATE FROM } a.\text{FINE}) \geq d))$

IF IS_ERROR
 GENERA ERRORE

$Q = (\text{SELECT } n.ID, (r.MAX - \text{SUM}(ap.QNT} \cdot \text{nd.GRAMMI})) \text{ AS } t$
 $\text{FROM } NUTRIENTE } n, PASTO } p, ALIM_PASTO } ap, NutAlim } nd, DIETA } d, REGOLA } r$
 $\text{WHERE } t > 0 \text{ AND } p.\text{UTENTE} = :u \text{ AND } ap.PASTO} = p.ID \text{ AND}$
 $ap.ALIM} = nd.ALIM \text{ AND } nd.NUT} = r.NUT \text{ AND } r.DIETA} = d.ID$
 $\text{AND } d.\text{UTENTE} = :u \text{ AND } \text{EXTRACT}(\text{DATE FROM } d.\text{INIZIO}) \leq d$
 $\text{AND } \text{EXTRACT}(\text{DATE FROM } d.\text{FINE}) \geq d$
 $\text{GROUP BY } n.ID)$

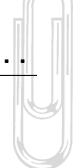
RETURN Q

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.]