

Basi di Dati 2021-22 – Progetto “Social Market” (12 CFU)

Le modifiche rispetto alla versione precedente sono indicate in rosso

Dominio applicativo – Specifiche

Si richiede di progettare e realizzare (una parte di) una base di dati a supporto di un social market. Un social market è un punto di recupero e distribuzione di alimenti, prodotti per l'igiene della persona e la pulizia della casa a favore di persone in stato di disagio socioeconomico.

Il social market reperisce e distribuisce risorse alimentari e beni di consumo “a costo zero”, utili a rispondere ai bisogni di cittadini in situazioni di fragilità sociale ed economica residenti nel territorio. In questo modo viene inoltre perseguita la finalità di lotta allo spreco, promuovendo lo scambio, l'utilizzo ed il recupero di beni invenduti dismessi, nell'osservanza delle norme di igiene e sicurezza, attraverso l'attivazione della comunità locale e delle Associazioni del territorio.

Chi è in difficoltà riceve prima di tutto potere d'acquisto: può fare la spesa scegliendo liberamente i prodotti che più servono alla famiglia, pagando non in euro ma con punti da un budget che è assegnato sulla base del reddito e del numero dei componenti famigliari (da un minimo di 30 ad un massimo di 60 punti mensili). L'autorizzazione ad accedere al social market è concessa dai Servizi Sociali, e dai Centri di Ascolto, previa verifica dei requisiti.

Il social market ha quindi un insieme di clienti, per ognuno dei quali viene registrato un codice, codice fiscale e dati anagrafici del titolare dell'autorizzazione, dati di contatto (telefono, ecc), quale ente/centro lo ha autorizzato, data di inizio autorizzazione (l'autorizzazione ha validità di sei mesi), i punti mensili e il saldo attuale, il numero di componenti il nucleo familiare e le relative fasce d'età (perché influenzano i prodotti da consigliare). Si memorizza inoltre l'elenco di chi può utilizzare i punti per quel nucleo (con i relativi dati anagrafici e di contatto), tipicamente sono autorizzati tutti i componenti il nucleo familiare sopra i 16 anni di età.

Il social market raccoglie prodotti e donazioni in denaro. Le donazioni in denaro vengono a loro volta utilizzate per acquistare prodotti o per coprire le spese di gestione. I prodotti vengono raccolti da supermercati e negozi (prodotti vicino alla scadenza donati direttamente dal supermercato, prodotti donati dai clienti nei carrelli posti dopo le casse, prodotti raccolti in occasione di apposite iniziative/giornate es. colletta alimentare) o anche da privati che li portano direttamente al market. Nel caso di supermercati e negozi i prodotti vengono ritirati presso la loro sede e trasportati al market a cura di volontari, utilizzando mezzi propri dei volontari.

Il social market tiene traccia di tutti i donatori, sia che siano privati che associazioni o esercizi commerciali con i relativi dati fiscali, recapiti e dati di contatto. Vengono inoltre registrate tutte le donazioni ricevute (data, importo se in denaro o ingresso merci se in prodotti).

Per ogni ingresso di prodotti nel market (donazione o acquisto diretto), viene registrata data e ora, chi riceve i prodotti, chi li consegna, e, nel caso di prodotti acquistati direttamente dal social market l'importo speso per l'acquisto (il market può acquistare prodotti perché ad esempio ha richiesta di shampoo e tra i prodotti donati non vi è lo shampoo).

In particolare, per i prodotti viene mantenuto l'inventario corrente, con, per ogni tipologia di prodotto, il numero di punti corrispondenti, la quantità disponibile e, nel caso di beni deperibili, le relative date di scadenza. Per alcune tipologie di prodotto si registra inoltre fino a quando, oltre la data di scadenza, il prodotto è ancora commestibile (es. olio di oliva: 1 anno aggiuntivo, pasta di grano duro 2 mesi, ecc). Le modifiche rispetto alla versione precedente sono indicate in rosso.

I clienti vengono ricevuti su appuntamento e per ogni appuntamento viene registrato il titolare dell'autorizzazione, data e ora dell'appuntamento, il componente del nucleo che verrà al market, il volontario che lo accoglierà e lo accompagnerà nella scelta dei prodotti. Il volontario registrerà la lista dei prodotti acquistati con relative quantità, in modo che sia aggiornato l'inventario prodotti. All'appuntamento saranno associati il saldo punti iniziale e finale.

Per motivi di privacy, viene accolto un solo cliente per volta, gli appuntamenti sono scaglionati di 5 minuti e hanno durata di 15 minuti.

Il social market si basa sul lavoro di volontari per accoglienza e assistenza negli acquisti, riordino dei prodotti, trasporto al market dei prodotti offerti dai supermercati, collaborazione a eventi/raccolte. Per ogni volontario si memorizzano le informazioni anagrafiche, i dati di contatto, eventuali associazioni a cui è collegato (es. scout Age-sci Genova 3), tipo di servizi per cui è disponibile, finestre temporali di disponibilità (es. giovedì pomeriggio dalle 3 alle 5), e, nel caso di servizi di trasporto, tipo di veicolo (auto, furgone, ecc).

Mensilmente il market prepara i turni per i servizi di accoglienza/riordino del mese successivo e i trasporti. Nel caso dei trasporti, si registra data, ora, chi vi prende parte, sede del ritiro e il numero di cestelli/scatoloni da ritirare mentre l'inventario dei prodotti ritirati viene effettuato quando i prodotti arrivano nel market.

Su base giornaliera, settimanale o mensile (a seconda della tipologia di prodotto) viene inoltre effettuato lo "scarico" dall'inventario dei beni non più distribuibili perché troppo vicini alla reale scadenza. Tali prodotti, a seconda dello stato e a discrezione del volontario che effettua la procedura, vengono posti in un apposito cestello da cui gli acquirenti possono prenderli senza utilizzare punti o gettati come rifiuti. L'unica operazione registrata nella base di dati è lo "scarico", per mantenere la disponibilità dei prodotti consistente, mentre la gestione successiva non viene registrata.

Prodotto	Tmc	Tempi di consumo consigliati oltre il Tmc
Succhi di frutta	6-12 mesi	Max 6 mesi
Caffè macinato	12-24 mesi	Max 1 anno
Olio extravergine di oliva	12-18 mesi	Max 1 anno
Conserve sottaceto e salsa di pomodoro	2-3 anni	1-2 mesi
Tonno sottolio	3-4 anni	1-2 mesi
Verdure sottolio	18-24 mesi	Rispettare Tmc
Prodotti da forno secchi	6-8 mesi	1 mese
Panettone, pandoro, colomba	4-5 mesi	1-2 settimane
Pasta, riso	24-30 mesi	1-2 mesi
Pesce surgelato	12-28 mesi	1-2 mesi (non consumare crudo)

Attività da svolgere

Il progetto consiste nelle seguenti attività (per ognuna di esse il materiale da consegnare è dettagliato nella sezione seguente).

I. **Progettazione logica:** Lo scopo di questa attività è la definizione di uno schema logico relativi a una base di dati relazionale per il dominio sopra descritto, ipotizzando un carico di lavoro a partire dalle specifiche sopra fornite (es. inserimento turni, pianificazione appuntamenti, gestione inventario prodotti, ecc). Il progetto dovrà essere completo della specifica di tutti i vincoli di integrità. Si dovrà verificare la qualità dello schema relazionale progettato, applicando i principi della teoria della normalizzazione. **In particolare, si dovrà stabilire se lo schema è normalizzato rispetto alla forma normale di Boyce Codd o rispetto alla terza forma normale. Se non è normalizzato rispetto alla terza forma normale, proporre una decomposizione senza perdita e che preserva le dipendenze per almeno una relazione non normalizzata e indicare quale forma normale soddisfa.**

II. Realizzazione:

Lo scopo di questa attività è la definizione in PostgreSQL dello schema logico, il suo popolamento e la realizzazione di alcune operazioni di interrogazione sulla base di dati, nonché di alcune routine e trigger.

A. La definizione dello schema logico della base di dati progettato dovrà contenere la specifica di tutti i vincoli individuati in fase di progettazione. Per ogni vincolo andrà individuata e indicata l'opportuna modalità di implementazione (vincoli di tipo CHECK o trigger). È richiesta l'implementazione dei soli vincoli di tipo CHECK. Per quanto riguarda il popolamento, la base di dati creata dovrà essere popolata con dati sufficienti almeno a verificare che i vincoli di dominio espressi siano verificati e che in generale le operazioni di cui si richiede l'implementazione funzionino correttamente.

Su tale basi di dati si richiede di realizzare:

B. La definizione di una vista che fornisca alcune informazioni riassuntive per ogni nucleo familiare: il numero di punti mensili a disposizione, i punti residui per il mese corrente, il numero di persone autorizzate per l'accesso al market, **il numero di componenti totali e quelli appartenenti alla fascia d'età più bassa**, il numero di spese effettuate nell'ultimo anno, i punti eventualmente non utilizzati nell'ultimo anno, la percentuale di punti utilizzata per prodotti deperibili e non deperibili nell'ultimo anno;

C. Le seguenti interrogazioni

- determinare i nuclei familiari che, pur avendo punti assegnati, non hanno effettuato spese nell'ultimo mese;
- determinare le tipologie di prodotti acquistate nell'ultimo anno da tutte le famiglie (cioè ogni famiglia ha acquistato almeno un prodotto di tale tipologia nell'ultimo anno);
- determinare i prodotti che vengono scaricati (cioè non riescono ad essere distribuiti alle famiglie) in quantitativo maggiore rispetto al quantitativo medio scaricato per prodotti della loro tipologia **(es. di tipologia: pasta/riso, tonno sottolio, olio, caffè, ecc.).**

D. Le seguenti procedure/funzioni:

- funzione che realizza lo scarico dall'inventario dei prodotti scaduti;
- funzione che corrisponde alla seguente query parametrica: dato un volontario e due date, determinare i turni assegnati al volontario nel periodo compreso tra le due date.

E. I seguenti trigger:

- verifica del vincolo che nessun volontario possa essere assegnato a più attività contemporanee (suggerimento: utilizzare il predicato OVERLAPS);
- mantenimento della disponibilità corrente dei prodotti.

III: Progettazione fisica, elaborazione delle interrogazioni, transazioni, controllo dell'accesso

- A. Lo scopo di questa attività è la progettazione fisica della base di dati, la sua creazione in PostgreSQL, un popolamento di dimensioni significative, la verifica dei piani di esecuzione scelti dal sistema per alcune interrogazioni prima e dopo la creazione dello schema fisico. In particolare, si richiede di
- individuare un carico di lavoro composto da tre interrogazioni a vostra scelta (almeno una interrogazione con selezione con condizione complessa e almeno una con join), specificandole in SQL e in linguaggio naturale;
 - progettare lo schema fisico a partire dal carico di lavoro individuato, motivando opportunamente, in modo sintetico, le scelte effettuate;
 - definire lo schema fisico progettato in PostgreSQL;
 - per le relazioni coinvolte nel carico di lavoro, inserire un numero di tuple significativo che portino ad occupare, per ciascuna relazione, qualche decina di pagine (in PostgreSQL ogni pagina, per default, ha dimensione 8 kb). Per far ciò, e solo per le relazioni coinvolte nel carico di lavoro, si suggerisce di utilizzare un generatore automatico di dati (ad esempio www.datanamic.com). Per raggiungere più velocemente l'occupazione di spazio disco indicata, si suggerisce di aggiungere solo a queste relazioni un attributo semanticamente inutile (dummy), definito con tipo char(n) o text. Questo permetterà di estendere la dimensione di ciascuna tupla e di arrivare all'occupazione di spazio disco indicata con un numero inferiore di tuple.
 - Individuare ed esaminare il piano di esecuzione scelto dal sistema e il tempo di esecuzione per le interrogazioni nel carico di lavoro **prima e dopo la creazione dello schema fisico**; confrontare i piani e i tempi ottenuti nei due casi, fornendo una giustificazione per i risultati ottenuti.
- B. Si richiede di specificare in linguaggio naturale e in PL/pgSQL una transazione con almeno una operazione di scrittura e almeno due operazioni di lettura, indicando quale livello di isolamento si ritiene opportuno.
- C. Si richiede di individuare per gli utenti **alice** (corrispondente a un gestore del market) e **roberto** (corrispondente a un volontario) i privilegi ritenuti ragionevoli nel dominio applicativo considerato, tenendo in considerazione quanto specificato nella descrizione del dominio, motivando le scelte effettuate, e specificare in SQL la politica di controllo dell'accesso individuata.

Deliverable (prodotti da consegnare)

Ogni attività al punto precedente dovrà produrre un deliverable (prodotto) di tipo software o di tipo report (documentazione) che illustri il processo di progettazione e le scelte effettuate. In particolare, i prodotti attesi per ogni attività sono i seguenti (S=codice SQL, D=documentazione in pdf):

PARTE I

1. [D] **Requisiti ristrutturati** in modo da eliminare ambiguità.
2. [D] **Progetto concettuale**, articolato in
 - (a) schema ER;
 - (b) documentazione relativa ai domini degli attributi (dizionario dati ed entità);
 - (c) vincoli non esprimibili nel diagramma;
 - (d) specifica dei tipi di gerarchie di generalizzazione.

Per i gruppi che hanno effettuato la peer review va effettuata la consegna completa, includendo anche i punti 1. e 2. finali che devono essere consistenti con i documenti di progettazione relativi ai passi successivi.

3. [D] **Progetto logico**, articolato in
 - (a) schema ER ristrutturato;
 - (b) eventuali modifiche dei domini degli attributi e informazioni sui domini di eventuali attributi introdotti;
 - (c) modifiche all'elenco di vincoli del modello concettuale (nuovi vincoli, eventuali vincoli eliminati o modificati);
 - (d) documentazione relativa alle scelte fatte per eliminare le gerarchie di generalizzazione;
 - (e) schema logico;
 - (f) verifica di qualità dello schema e eventuali ottimizzazioni applicate tenendo in considerazione il carico di lavoro

PARTE II

4. [S] Script SQL per la **creazione dello schema** logico della base di dati in accordo allo schema relazionale ottenuto alla fine della fase di progettazione logica, per la porzione necessaria per i punti successivi (cioè le tabelle coinvolte dalle interrogazioni nel carico di lavoro, nella definizione della vista, nelle interrogazioni, in funzioni, procedure e trigger). Lo schema dovrà essere comprensivo dei vincoli esprimibili con check, e per il **popolamento** di tale base di dati.
5. [D] diagramma che visualizza lo script SQL in forma grafica ottenuto con DataGrip (vedi [Aulabweb](#) per come crearla da 4.).
6. [S] Il codice SQL sviluppato per implementare le **interrogazioni** e la **vista** richieste, inserendo, come commento nello script, la corrispondente richiesta in linguaggio naturale.
7. [S] Il codice PL/pgSQL sviluppato per implementare le **funzioni e procedure** richieste, inserendo, come commento nello script, la specifica relativa a ciascuna funzione o procedura.
8. [S] Il codice PL/pgSQL sviluppato per implementare i **trigger** richiesti, inserendo, come commento nello script, la specifica relativa a ciascun trigger.

PARTE III

9. [D] **Progetto fisico e sua validazione**, articolato in
 - a) Descrizione in linguaggio naturale e il codice SQL sviluppato per implementare le **interrogazioni del carico di lavoro**.
 - b) **progetto fisico**, contenente l'elenco degli indici che si intendono creare per le interrogazioni contenute nel carico di lavoro (specificando relazione di riferimento e chiave di ricerca), il loro tipo (ordinato/hash, clusterizzato/non clusterizzato) e la motivazione che ha portato alla loro creazione.
 - c) Una tabella che, per ogni tabella coinvolta nelle operazioni del carico di lavoro, riporti il numero di **tuple inserite e la dimensione in blocchi** della tabella.
 - d) Descrizione dei **piani di esecuzione** scelti dal sistema (prima e dopo la creazione dello schema fisico) per le interrogazioni contenute nel carico di lavoro, riportando la **visualizzazione del piano di esecuzione** prodotta da PostgreSQL e i tempi di esecuzione (prima e dopo la creazione dello schema fisico); confrontare i piani e i tempi ottenuti nei due casi, fornendo una giustificazione per i risultati ottenuti.
10. [D] Descrizione in linguaggio naturale e codice PL/pgSQL della **transazione** considerata e giustificazione per il livello di isolamento prescelto.
11. [D] Descrizione della politica di **controllo dell'accesso** scelta, motivando le scelte effettuate. Si suggerisce di riassumere la politica di controllo dell'accesso con una tabella contenente una riga per ogni tabella, una colonna per ogni utente. Ogni cella (i,j) dovrà contenere i privilegi che si intendono assegnare all'utente j sulla tabella i.
12. [S] Script SQL per la **creazione dello schema fisico** della base di dati e per la specifica delle **interrogazioni** contenute nel **carico di lavoro**, inserendo, come commento nello script, la corrispondente richiesta in linguaggio naturale.
13. [S] Il codice PL/pgSQL sviluppato per implementare la **transazione** considerata.
14. [S] Script SQL per l'implementazione della politica di **controllo dell'accesso**.

Modalità di svolgimento e consegna

Svolgimento

Il progetto deve essere svolto **a gruppi**. Nel caso di studenti che hanno aderito al patto d'aula il gruppo è quello **assegnato, per gli altri** la composizione dei gruppi (tre persone) va dichiarata su AulaWeb (sezione Progetto) al momento dell'inizio del lavoro e **al più tardi una settimana prima dell'appello in cui si consegna il progetto**.

Scadenze per la consegna

Sono previste consegne **in corrispondenza di ogni appello di esame**, con consegna da effettuarsi **via AulaWeb entro le ore 9.00 della data dell'esame scritto**.

Modalità di consegna su AulaWeb

Si deve consegnare, utilizzando il compito Consegna progetto su AulaWeb, un singolo file **zip** con nome = GruppoXX con XX numero del gruppo (sarà impostata la consegna di gruppo) contenente la documentazione e il codice richiesti ai punti precedenti, opportunamente organizzata in file.

Il file zip dovrà contenere

- un unico documento pdf per la parte I, organizzato in sezioni, il titolo di ciascuna sezione dovrà corrispondere al deliverable corrispondente
- un documento pdf contenente il diagramma DataGrip dello schema logico (prodotto II.5)
- un file SQL per la parte II, organizzato in sezioni (indicate come commenti nel file), inserendo come commento la richiesta a cui si riferisce il codice SQL (e riportando per ogni interrogazione/vista la specifica come commento). I file SQL devono essere file SQL ... cioè formato solo testo e apribili ed eseguibili in pgAdmin;
- un unico documento pdf per la parte III, organizzato in sezioni, il titolo di ciascuna sezione dovrà corrispondere al deliverable corrispondente (si noti che questo documento, oltre a spiegazioni in linguaggio naturale dovrà anche riportare del codice SQL, come da descrizione deliverable PARTE III);
- un file SQL per la parte III, organizzato in sezioni (indicate come commenti nel file), inserendo come commento la richiesta a cui si riferisce il codice SQL (e riportando per ogni interrogazione/vista la specifica come commento). I file SQL devono essere file SQL ... cioè formato solo testo e apribili ed eseguibili in pgAdmin.

Il progetto consegnato deve essere completo (rispetto alle specifiche) e funzionante!

Attenzione: sono permesse consegne ripetute, ma farà fede solo l'ultima e, se sufficiente, verrà valutato un solo progetto (quindi, se avete ambizioni di voto, consegnate solo se siete convinti di aver svolto il progetto in maniera per voi soddisfacente, perché non potrete rifarlo per alzarvi il voto). In caso di progetto con valutazione insufficiente, verrà richiesto di consegnare un nuovo progetto (una sola volta per ogni anno accademico), che dovrà tenere conto dei commenti/correzioni ricevuti sul progetto insufficiente.

Criteri di valutazione

- Correttezza e completezza dei deliverable (per esempio, lo schema concettuale, lo schema ristrutturato e lo schema logico devono essere singolarmente corretti).
- Consistenza complessiva dei deliverable e corretta applicazione delle metodologie (per esempio, lo schema concettuale, lo schema ristrutturato e lo schema logico devono essere tra di loro coerenti).
- Qualità della documentazione in termini di chiarezza e coerenza con le richieste.