

BASI DI DATI – INGEGNERIA INFORMATICA
SPECIFICHE DI PROGETTO PER L'ANNO ACCADEMICO 2013 – 2014
Prof. Gigliola Vaglini, Ing. Francesco Pistolesi

1 Descrizione dei requisiti delle fasi di progettazione

Si desidera progettare un database relazionale per la gestione dei dati relativi a un social network avente le funzionalità descritte nel paragrafo 2. Il progetto è articolato nelle seguenti fasi:

1. Analisi delle specifiche fornite;
2. Progettazione concettuale e produzione del diagramma entità-relazione;
3. Ristrutturazione del diagramma entità-relazione;
4. Analisi delle dipendenze funzionali e normalizzazione dello schema;
5. Individuazione di interrogazioni interessanti sui dati;
6. Analisi delle prestazioni delle operazioni individuate;
7. Introduzione/eliminazione di eventuali ridondanze;
8. Progettazione logica e produzione del modello logico relazionale;
9. Scrittura di uno script MySQL che crei il database e lo popoli;
10. Progettazione fisica.

1.1 Analisi delle specifiche fornite

In questa fase preliminare devono essere lette e analizzate nel dettaglio le specifiche fornite nel paragrafo 2. Lo scopo dell'analisi delle specifiche è quello di capire le funzionalità per le quali si progetta il database, al fine di dotare ognuna di esse del supporto per i dati di cui necessita.

1.2 Progettazione concettuale e produzione del diagramma entità-relazione

Scegliere una strategia di progettazione concettuale fra quelle viste a lezione (top-down, bottom-up, inside-out...). Il diagramma entità-relazione deve contenere: nomi di entità e associazioni; attributi di entità e associazioni; identificatori primari delle entità; cardinalità delle associazioni; eventuali generalizzazioni. La notazione da utilizzare per il diagramma entità-relazione è quella standard vista a lezione. Non sono accettate notazioni alternative.

1.3 Ristrutturazione del diagramma entità-relazione

La ristrutturazione prevede che siano eliminate le generalizzazioni mediante opportune traduzioni alternative (accorpamenti, introduzione di associazioni...). Inoltre, gli eventuali attributi multivalore presenti devono essere opportunamente gestiti in modo tale da poter essere tradotti nel modello logico relazionale.

1.4 Analisi delle dipendenze funzionali e normalizzazione dello schema

Per ciascuna relazione (tabella) individuata, devono essere ricercate tutte le dipendenze funzionali. In particolare, è richiesto che il database progettato sia, a questo livello della progettazione, in forma normale di Boyce-Codd.

1.5 Individuazione di interrogazioni interessanti sui dati

Devono essere ideate 8 operazioni (query) interessanti da effettuare sui dati che devono essere scritte in SQL utilizzando la sintassi MySQL. Le operazioni individuate possono essere sia query di selezione, che query di inserimento/modifica/cancellazione.

1.6 Analisi delle prestazioni delle operazioni

Per analizzare le prestazioni delle operazioni individuate si devono dapprima compilare le tavole dei volumi con le quali stimare la mole di dati contenuta in ciascuna tabella del database. Ovviamente per effettuare un'analisi delle prestazioni che sia relativamente indicativa, le stime devono essere fatte con giudizio. Dopodiché, per ciascuna operazione si deve effettuare una stima sulla sua frequenza giornaliera di esecuzione, per esempio, 100 volte al giorno. Infine, per ciascuna operazione deve essere compilata la tavola degli accessi, ottenendo così una stima delle operazioni elementari (letture e scritture) necessarie all'esecuzione, sempre in funzione delle stime presenti nelle tavole dei volumi.

1.7 Introduzione/eliminazione di ridondanze

Dallo studio delle prestazioni di ciascuna operazione può emergere che alcune di esse potrebbero trarre beneficio dall'introduzione di ridondanze. Ogniquale volta si introduce una ridondanza per una operazione, per tale operazione deve essere compilata anche la tavola degli accessi che mostra il numero di operazioni elementari eseguite in presenza della ridondanza. Infine, deve essere presa una decisione sul mantenere o no la ridondanza introdotta, in base al risparmio di operazioni elementari che essa comporta. Si faccia attenzione che, scelta un'operazione, una ridondanza ne comporta un alleggerimento del carico (riduce le operazioni elementari), ma d'altra parte, inevitabilmente, essa deve essere mantenuta aggiornata per essere utilizzata. Al fine di decidere se mantenere o no una ridondanza, deve essere studiata anche l'operazione di aggiornamento della stessa, la sua frequenza giornaliera, e la sua complessità in termini di operazioni elementari. Ciò permette di calcolare un rapporto costo-beneficio, dove il beneficio è il risparmio di operazioni elementari che la ridondanza comporta per l'operazione della quale si è deciso di migliorare le prestazioni, mentre il costo (quindi lo svantaggio) è dovuto all'introduzione dell'operazione che mantiene aggiornata la ridondanza, in termini di operazioni elementari. Si noti che in assenza della ridondanza, l'operazione di aggiornamento della stessa non esiste, quindi tutto il carico computazionale che si introduce per mantenere aggiornata la ridondanza deve essere motivato dal beneficio che comporta la sua presenza. In buona sostanza, scelta un'operazione che si intende migliorare, si deve confrontare la tavola degli accessi dell'operazione stessa, così com'è, con la tavola degli accessi dell'operazione in presenza di ridondanza, e con la tavola degli accessi dell'operazione che mantiene aggiornata la ridondanza. La documentazione di progetto deve contenere questa analisi, e le motivazioni che spingono al mantenimento o no di una ridondanza introdotta.

Nella versione finale del database è richiesta la presenza di almeno due ridondanze e, per ciascuna ridondanza, deve essere fornita un'operazione di lettura e una di scrittura che impatti con essa.

1.8 Progettazione logica e produzione del modello logico relazionale

Il diagramma entità-relazione ristrutturato deve essere tradotto nel modello logico relazionale, producendo lo schema del database. In questa fase devono essere scelte le traduzioni più appropriate, qualora un costrutto concettuale sia traducibile in modi diversi. Le scelte effettuate devono essere opportunamente motivate. Inoltre, devono essere specificati chiaramente tutti i vincoli di integrità referenziale necessari al corretto funzionamento del database. Infine, eventuali vincoli di integrità generici devono essere espressi mediante trigger MySQL. Sono necessari almeno 2 vincoli di integrità generici. Un esempio di vincolo di integrità generico può consistere nel limitare il numero di post che un utente può pubblicare giornalmente sulla sua bacheca.

1.9 Scrittura di uno script MySQL che crei il database e lo popoli

Lo script MySQL deve creare il database e popolare ogni sua tabella con almeno 10 record. Lo script deve contenere anche tutti i vincoli di integrità referenziale, i trigger che gestiscono i vincoli

di integrità generici, e gli eventi e le stored procedure che realizzano la business intelligence (vedi paragrafo 2.5).

1.10 Progettazione fisica (opzionale)

La fase di progettazione fisica è mirata all'individuazione degli indici che possono essere necessari per rendere più efficienti le operazioni sul database. Deve essere ottimizzata almeno una delle operazioni individuate al punto 1.6. Gli indici introdotti devono essere motivati dai piani di esecuzione delle query prima e dopo l'introduzione degli indici. Deve essere mostrato che effettivamente il DBMS sfrutta gli indici introdotti e ne trae vantaggio.

2 Specifiche

Il database che si intende progettare deve poter supportare le funzionalità di un social network (si pensi, per esempio, a Facebook, Twitter, ecc). Si noti che l'oggetto del lavoro è la progettazione della organizzazione dei dati. Le funzionalità descritte nei seguenti paragrafi non devono essere realizzate dal punto di vista applicativo, ma deve essere offerta la possibilità a un'ipotetica applicazione web che implementa il social network, di memorizzare tutti i dati che occorrono, in modo strutturato.

2.1 Gestione dei profili

Un nuovo utente, una volta raggiunto il sito web che ospita il social network, ha la possibilità di creare un proprio profilo. Un profilo contiene le classiche informazioni sull'anagrafica, sugli interessi (musica, generi di film, libri...) e sulla vita in generale (istruzione, scuole frequentate, hobby, attuale impiego...). Un profilo può contenere messaggi, immagini (anche organizzate in gallerie), file audio, file video, documenti elettronici di vario genere (testo, fogli di calcolo...). Ovviamente un profilo sarà associato a un nome utente e a una password per effettuare l'accesso. Inizialmente un profilo non è collegato ad altri profili, ciò significa che non sono presenti legami pensabili come amici o follower.

2.1.1 Richieste di interesse

Man mano che l'utente utilizza il social network, può ricevere/inviare richieste di interesse da/a utenti che possiedono un profilo sul social network. Considerando due utenti u_1 e u_2 , si possono verificare i seguenti scenari:

- l'utente u_1 invia una richiesta all'utente u_2 e quest'ultimo la accetta
in questo caso u_1 comparirà fra i follower di u_2 , e u_1 avrà u_2 fra i suoi following;
- l'utente u_1 invia una richiesta all'utente u_2 e quest'ultimo non la accetta
nel profilo di u_1 e nel profilo di u_2 deve essere possibile visualizzare la richiesta pendente;
- l'utente u_1 invia una richiesta all'utente u_2 e quest'ultimo accetta e ricambia
in questo caso u_1 comparirà fra i follower di u_2 , e u_2 fra i follower di u_1 .

Quando l'utente u_2 accetta la richiesta di interesse, dà la possibilità a u_1 di visualizzare tutti i contenuti del suo profilo. Deve essere possibile limitare, qualora u_2 lo desideri, la visibilità di taluni contenuti all'utente u_1 : per esempio, u_2 potrebbe desiderare che u_1 non visualizzasse alcune gallerie fotografiche, oppure altre aree del suo profilo. Inoltre, dal momento dell'accettazione, u_1 può inviare messaggi pubblici a u_2 , che saranno pubblicati sulla bacheca di quest'ultimo e visibili a tutti i follower di u_2 . Se u_2 compie la sola accettazione, senza ricambiare, u_1 non potrà inviare messaggi privati a u_2 . Un messaggio privato è visibile solamente al destinatario dello stesso.

Se, invece, l'utente u2 accetta e ricambia allora u1 e u2 potranno scambiarsi anche messaggi privati. Rimane per u2 la possibilità di limitare a u1 la visibilità di alcune aree del suo profilo. D'altra parte, anche u1 può scegliere quali contenuti del suo profilo rendere visibili a u2. Inoltre, sempre e solo se u2 accetta e ricambia, u1 ha la possibilità di inviare messaggi a u2 impostando un livello di visibilità intermedio fra privato e pubblico. In particolare, u1 può decidere quali follower di u2 potranno visualizzare il messaggio. L'utente u2 non può modificare (allargare o restringere) la visibilità dei messaggi inviati da u1.

Qualora u2 non accetti la richiesta di interesse di u1, questa rimane comunque visibile sia a u2 che a u1 come richiesta pendente. Trascorsi 30 giorni dall'invio della richiesta, se u2 non accetta, questa viene automaticamente rimossa. L'utente u1 non potrà inviare richieste di interesse all'utente u2 per i successivi sei mesi, a partire dalla rimozione della richiesta dal sistema.

2.1.2 Gruppi

Un gruppo è un insieme utenti, rappresentati dai relativi profili sul social network. Un gruppo ha un tema che rappresenta l'obiettivo per il quale esso è stato creato. Ad esempio, un gruppo può riguardare un insieme di studenti che condividono lo stesso corso di studi, un insieme di persone a cui piace un determinato personaggio famoso e così via. Un gruppo può essere creato da un qualsiasi utente del social network. L'utente che crea il gruppo è automaticamente assegnatario del ruolo di amministratore del gruppo stesso. L'amministratore sceglie il livello di visibilità e le politiche di accesso ai contenuti del gruppo. La visibilità può essere pubblica oppure ristretta. Se è pubblica, chiunque può visualizzare i contenuti del gruppo; se è ristretta, solo gli utenti che ricevono l'approvazione dell'amministratore possono visualizzare i contenuti del gruppo. Chi riceve l'approvazione può visualizzare tutti i contenuti del gruppo e inviare messaggi al gruppo. In generale, un gruppo è un tipo particolare di profilo dove ogni membro può pubblicare messaggi testuali, immagini, video, ecc.

2.1.3. Gestione dei blocchi

Gli utenti del social network hanno la possibilità di bloccare uno o più follower. Questo può avvenire solamente dopo due segnalazioni. In sostanza, un utente può "avvertire" preventivamente un suo follower che quest'ultimo è a rischio di blocco, senza specificarne una ragione. Il follower dovrà comportarsi di conseguenza al fine di non essere successivamente bloccato. La segnalazione impedisce al follower di visualizzare il profilo dell'utente infastidito per un mese, dopodiché tale profilo torna raggiungibile al follower. Qualora l'utente, infastidito da ulteriori comportamenti del follower, decida di bloccarlo, ciò impedisce al follower di visualizzare il profilo per un tempo indeterminato, fino a un eventuale sblocco.

2.1.4 Riservatezza

Un utente del social network può impostare regole di visualizzazione dei suoi contenuti a più livelli di granularità. In particolare, può decidere che cosa è visibile pubblicamente, che cosa può essere visto e da quali utenti, e così via.

2.2 Gestione dei contenuti

I contenuti presenti sui profili di un utente (foto, messaggi in bacheca, video, documenti elettronici...), dipendentemente dalle regole di riservatezza impostate, possono costituire oggetto di interazione tra follower. In particolare, un follower può commentare foto, aggiungere semplicemente un "like", inserire un tag in un'immagine pubblicata da lui stesso o da un follower/following. Tutte le

informazioni relative alle interazioni precedenti devono essere mantenute nel database. L'insieme delle interazioni può essere esteso a piacere.

2.3 Gestione degli eventi

Il social network dà la possibilità ai suoi utenti di organizzare feste o eventi speciali. Ogni evento è caratterizzato da un luogo e una data nella quale si svolgerà, un insieme di utenti che (potenzialmente) parteciperanno, e un allestimento. Un allestimento è un insieme di oggetti e/o strutture che sono necessarie allo svolgimento dell'evento. Ad esempio, una festa di compleanno a sorpresa dovrà essere allestita considerando il cibo e le bevande, ma anche eventuale presenza di personale adibito all'intrattenimento, un programma di massima, e così via.

Un evento è creato da un utente del social network, il quale ne decide il luogo, la data, e spedisce inviti a chi desidera fra i suoi follower e following. Un invito è visibile, ovviamente, solo al diretto interessato che può accettarlo o rifiutarlo. Qualora lo accetti, può indicare una sua presenza sicura oppure potenziale all'evento. Gli invitati possono proporre delle varianti per l'allestimento. Tali varianti saranno gestite tramite sondaggio. Si veda a tale scopo il paragrafo successivo.

2.4 Gestione di sondaggi

Un sondaggio offre la possibilità a un insieme di utenti di esprimere la loro opinione circa un determinato argomento. Un sondaggio è creato da un utente del social network, il quale specifica l'argomento, le persone alle quali è rivolto, e le possibili risposte che possono essere fornite. Per esempio, un sondaggio di gradimento di una festa (ormai conclusa) può essere basato su un indicatore di gradimento da 1 a 10, oppure dalla scelta di una particolare risposta in un insieme di risposte impostate preventivamente dal creatore del sondaggio. Risposte testuali possono rappresentare un modo alternativo di esprimere un'opinione, anche se sono difficilmente processabili in maniera automatica.

2.5 Data mining e business intelligence

Il social network per il quale si intende progettare il database è un social network gratuito. L'utente che crea un profilo è informato preventivamente che sulle pagine del social network saranno presenti annunci pubblicitari. Il gestore del social network mette a disposizione nelle pagine spazi pubblicitari che possono essere acquistati da aziende produttrici di beni o fornitrici di servizi. Chiaramente, tali aziende investono tanto più capitale in pubblicità quanto più essa è in grado di provocare apprezzabili incrementi delle vendite. L'azienda che gestisce il social network (si pensi a Facebook) trae i propri profitti unicamente dalle somme di capitale che derivano da investimenti in pubblicità. Per fare in modo che sempre più sponsor desiderino inserire i propri banner pubblicitari sulle pagine del social network, c'è bisogno di "far fruttare" i banner visualizzati, mostrandoli a chi è veramente interessato ad essi, e che quindi possa rappresentare un potenziale cliente delle aziende sponsor. Per esempio, se sulla pagina dell'utente Tizio che è appassionato di buona cucina e di viaggi, sono visualizzati annunci pubblicitari di promozioni sull'acquisto di rocchetti di filo per cucire, è ovvio che la pubblicità non sarà remunerativa: Tizio non farà mai clic su tali banner, e l'azienda di rocchetti di filo per cucire non trarrà vantaggio dalla pubblicità.

La business intelligence e il data mining sono discipline che hanno l'obiettivo di estrapolare informazione nascosta nei dati per poterla utilizzare per qualche scopo, sia economico che di diversa utilità. Il problema della categorizzazione degli utenti di un social network al fine di proporre pubblicità mirata, quindi interessante per l'utente, è un problema di business intelligence. Si richiede che il database da progettare possieda un'area dedicata alla business intelligence, ovviamente basata anche su ridondanze, per la gestione intelligente degli annunci pubblicitari: devono essere quindi create le opportune tabelle, e deve essere scritto il codice degli eventi e delle stored

procedure MySQL che la implementano, tenendo conto che le informazioni di quest'area del database saranno aggiornate nei momenti di minore carico applicativo, ad esempio di notte.

(opzionale) Oltre alla categorizzazione degli utenti finalizzata alla pubblicità mirata, il database deve offrire almeno un'altra funzionalità (non banale) di business intelligence ed anche per tale funzionalità devono essere create le opportune tabelle e deve essere scritto il codice degli eventi e delle stored procedure MySQL che la implementano.