

APP STORE

ELABORATO PER IL CORSO DI BASI DI DATI

Elia Urbini matr.0000874536

Michele Pio Prencipe matr.0000875129

STRUTTURA DELL'ELABORATO

La presente documentazione tratta nel dettaglio la progettazione e l'implementazione dell'elaborato "App Store" di Elia Urbini e Michele Pio Prencipe, ed è strutturata come segue:

1. Introduzione
2. Analisi dei requisiti
 - 2.1. Requisiti in linguaggio naturale
 - 2.2. Estrazione dei concetti fondamentali
3. Progetto dello schema concettuale

1. INTRODUZIONE

Il progetto consiste nella realizzazione di un sistema database che funga da supporto ad una piattaforma online che permetta agli utenti di scaricare e acquistare applicazioni sull'impronta dell'App Store per iOS e al Play Store di Android. Le applicazioni possono essere sia gratuite che a pagamento e possono essere scaricati da diversi dispositivi.

Coloro che possono interagire col software si distinguono in sviluppatori, ovvero coloro che hanno progettato e creato una o più applicazioni e utenti, i quali hanno la possibilità di scaricare gli applicativi piuttosto che effettuare pagamenti all'interni degli stessi oppure lasciare eventuali giudizi attraverso recensioni. Tuttavia essere uno sviluppatore non preclude ad una persona di usufruire dei servizi offerti per l'"utente standard".

Dal momento che le interrogazioni da poter fare al database possono essere suddivise a seconda della funzione del fruitore, abbiamo preferito distinguere due interfacce di accesso diverse.

Il Sistema informativo offrirà quindi due client distinti. In questa relazione verrà esposta un'interfaccia client d'amministrazione, che implementi sia le funzioni che verranno usate dagli utenti che le funzioni degli autori delle applicazioni.

2.ANALISI DEI REQUISITI

2.1-Requisiti in linguaggio naturale

La seguente descrizione riporta in linguaggio naturale i requisiti per il nostro sistema informativo:

Un'azienda che sta lanciando sul mercato un nuovo modello di smartphones richiede di poter tener traccia dei dati relativi al suo app store. Tale piattaforma è accessibile sia agli utenti che agli sviluppatori di applicazioni. Gli utenti così come gli sviluppatori possono usufruire dei servizi messi a disposizione solamente previa registrazione. In particolare, l'utente standard è colui che scarica le applicazioni le quali possono essere così distinte: applicazioni gratuite, il cui download non richiede nessuna transazione e applicazioni a pagamento. Inoltre ogni applicazione può prevedere pagamenti all'interno della stessa che consentono all'utente di accedere a contenuti e funzionalità esclusivi e possono o meno essere verificate, ovvero certificate dal gestore dello store. I pagamenti possono essere effettuati tramite diverse carte di credito piuttosto che attraverso giftcard, ossia carte acquistabili fisicamente che con l'inserimento di un codice forniscono denaro utilizzabile nello store stesso. L'azienda vuole che dell'utente vengano memorizzati oltre che ai suoi dati di registrazione anche i dispositivi, le carte di credito e le giftcard da egli utilizzate. Si ha inoltre la necessità di memorizzare le ricerche effettuate e le recensioni fornite dagli utenti, per cercare di proporre applicazioni affini ad ogni fruitore. Per quanto riguarda gli sviluppatori, si intende tenere in memoria i dati relativi agli acquisti associati alle proprie applicazioni e le informazioni del conto bancario sul quale verranno versati i guadagni. Infine, a proposito delle applicazioni, si intende modellare oltre che ai download e agli acquisti effettuati anche i requisiti di installazione, la categoria di appartenenza (ad esempio giochi, social networks ecc...), le versioni con relativa data di pubblicazione e le informazioni per rivolgersi al supporto. Per rendere l'esperienza degli utenti migliore, è possibile segnalare agli autori eventuali bug presenti in una determinata versione dell'applicativo.

2.2-Estrazione dei concetti fondamentali

Individuiamo adesso le parole e le espressioni chiave che ci consentiranno di realizzare uno schema significativo del progetto e di raffinarlo successivamente per ottenere lo schema definitivo. I termini di rilievo appaiono nel testo con una sottolineatura:

Un'azienda che sta lanciando sul mercato un nuovo modello di smartphones richiede di poter tener traccia dei dati relativi al suo app store. Tale piattaforma è accessibile sia agli **utenti** che agli **sviluppatori** di applicazioni. Gli utenti così come gli sviluppatori possono usufruire dei servizi messi a disposizione solamente previa **registrazione**. In particolare, l'utente standard è colui che scarica le **applicazioni** le quali possono essere così distinte: **applicazioni gratuite**, il cui download non richiede nessuna transazione e **applicazioni a pagamento**. Inoltre ogni applicazione può prevedere **pagamenti** all'interno della stessa che consentono all'utente di accedere a contenuti e funzionalità esclusivi e possono o meno essere **verificate**, ovvero certificate dal gestore dello store. I pagamenti possono essere effettuati tramite diverse **carte di credito** piuttosto che attraverso **giftcard**, ossia carte acquistabili fisicamente che con l'inserimento di un codice forniscono denaro utilizzabile nello store stesso. L'azienda vuole che dell'utente vengano memorizzati oltre che ai suoi dati di registrazione anche i **dispositivi**, le carte di credito e le giftcard da egli utilizzate. Si ha inoltre la necessità di memorizzare le **ricerche** effettuate e le **recensioni** fornite dagli utenti, per cercare di proporre applicazioni affini ad ogni fruitore. Per quanto riguarda gli sviluppatori, si intende tenere in memoria i dati relativi agli **acquisti** associati alle proprie applicazioni e le informazioni del **conto bancario** sul quale verranno versati i guadagni. Infine, a proposito delle applicazioni, si intende modellare oltre che ai **download** e agli acquisti effettuati anche i **requisiti di installazione**, la **categoria** di appartenenza (ad esempio giochi, social networks ecc...), le **versioni**

con relativa data di pubblicazione e le informazioni per rivolgersi al **supporto**. Per rendere l'esperienza degli utenti migliore, è possibile segnalare agli autori eventuali **bug** presenti in una determinata versione dell'applicativo.

La descrizione è già abbastanza chiara e completa. Occorre comunque fare alcune precisazioni.

- “Sviluppatore” si intende l'entità che produce un'applicazione. Non necessariamente è identificata da una singola persona, ma si può trattare anche di software house o aziende. Per ovviare a questa ambiguità abbiamo scelto l'utilizzo del termine autore.
- “Registrazione” si intende modellare le credenziali di accesso sia degli utenti che degli autori delle applicazioni.

3. PROGETTO DELLO SCHEMA CONCETTUALE

Lo sviluppo dello schema Entity-Relationship procederà ora per fasi successive più o meno indipendenti da loro. Lo schema generale può essere suddiviso in più ambiti, in ciascuno dei quali si discuteranno soluzioni adottate, vincoli espressi e inespressi e altro. Gli schemi verranno poi raffinati e arricchiti.

- Applicazione
- Utente
- Autore

Lo schema concettuale nella sua versione finale si avvarrà delle seguenti entità e associazioni (per ciascuna è fornita una breve descrizione):

NOME	TIPO	DESCRIZIONE
AUTORE	E	Rappresenta l'autore di almeno un'applicazione
ACCOUNT BANCARIO	E	Account bancario di un autore, nel quale vengono depositati i guadagni
POSSESSO	R	Lega un autore al suo account bancario

INDIRIZZO	E	Rappresenta le informazioni di un indirizzo
ALLOCAZIONE	R	Lega un autore al proprio indirizzo
CREDENZIALI	E	Rappresenta le informazioni di accesso all'app store
ACCESSO	R	Lega un autore alle proprie credenziali di accesso
UTENTE	E	Rappresenta un utente della piattaforma
RESIDENZA	R	Lega un utente all'indirizzo dove risiede
GIFTCARD	E	Rappresenta una carta con codice univoco che permette di ricaricare il conto di un utente
RISCOSSIONE	E	Lega un utente alle giftcard rimosse
PORTAFOGLIO	E	Rappresenta il conto di un utente dentro all'app store con il quale può effettuare acquisti
GESTIONE	R	Lega un utente al suo portafoglio
CARTA DI CREDITO	E	Rappresenta le informazioni di una carta di credito
POSSIEDE	R	Lega un utente alle carte di credito in suo possesso
RICERCA	E	Rappresenta una ricerca effettuata da un utente
COMPIE	R	Lega un utente alle ricerche da lui effettuate

DISPOSITIVO	E	Rappresenta un dispositivo in grado di scaricare applicazioni
DISPONE	R	Lega un utente ai dispositivi in suo possesso
DOWNLOAD	E	Rappresenta il download di una versione di un'app da parte di un utente mediante un dispositivo
SCARICA	R	Lega utente ad un download da lui effettuato
INSTALLAZIONE	R	Lega un dispositivo ad un download
RECENSIONE	E	Rappresenta una recensione di un utente relativa ad un'app sotto forma di voto e commento
PUBBLICAZIONE	R	Lega un utente ad una sua recensione
ACQUISTO	E	Rappresenta le informazioni di un acquisto effettuato nella piattaforma
EFFETTUA	R	Lega un utente ad un suo acquisto
APPLICAZIONE	E	Rappresenta un'applicazione disponibile nello store. Generalizzazione di app verificata, app a pagamento e app con pagamenti interni
SCARICAMENTO	R	Lega un'app ad un suo download
CREAZIONE	R	Lega un autore ad un'app da lui creata
VALUTAZIONE	R	Lega un'app ad una recensione effettuata su di essa

REQUISITO	E	Rappresenta i requisiti hardware e software necessari per il download di un'app
NECESSITA	R	Lega un'app ai suoi requisiti
CATEGORIA APPLICAZIONE	E	Rappresenta una categoria per catalogare le applicazioni
APPARTENENZA	R	Lega un'app alla categoria alla quale appartiene
VERSIONE	E	Rappresenta una versione di un'applicazione
RILASCIO	R	Lega un'app ad una sua versione
SCARICATA	R	Lega un download ad una versione di un applicativo
RIGUARDA	R	Lega una recensione ad una versione di un'app
SUPPORTO	E	Rappresenta le informazioni per contattare il supporto di un'app
INTEGRA	R	Lega un'app al suo supporto
APP VERIFICATA	E	Rappresenta un tipo di app che è stata verificata
APP CON PAGAMENTI INTERNI	E	Rappresenta un tipo di app che contiene pagamenti al suo interno
APP A PAGAMENTO	E	Rappresenta un tipo di app il cui download è a pagamento
OBBLIGO	R	Lega un'app a pagamento con acquisto

PREVEDE	R	Lega un'app con pagamenti interni con acquisto
SEGNALAZIONE BUG	E	Rappresenta le informazioni relative ad una segnalazione di un bug
RICEVE	R	Lega un'app ad una segnalazione di bug
RIPORTA	R	Lega una versione di un'app ad una segnalazione di bug
INDIVIDUA	R	Lega un utente ad una sua segnalazione di un bug

3.1 Sviluppo degli autori

Per modellare questo aspetto del database sarà semplicemente necessario rappresentare l'entità autore e account bancario messe in relazione dall'entità possesso. Siccome nel nostro modello ogni autore ha un unico account bancario e ogni account bancario è gestito da un

solo autore allora la relazione sarà di tipo one to one. Inoltre per gestire l'accesso degli autori nello store ci avvaliamo di un'entità credenziali la quale ovviamente sarà legata univocamente ad ogni autore. Infine è necessario mettere in relazione un autore con l'indirizzo a lui associato. In quanto ogni autore può non avere indirizzi come averne più di uno, queste due entità saranno legate mediante una relazione many to many.

3.2 Sviluppo degli utenti

In questo ambito si intende modellare le informazioni relative agli utenti registrati nell'app store. Per far ciò sarà necessaria sicuramente un'entità utente contenente i dati generali di ogni utente, la quale verrà associata alle altre entità del suo dominio relativo agli utenti. In particolare, si vuole modellare:

- Login: ogni utente, come precedentemente osservato per gli autori, deve disporre di credenziali univoche (username e password) per accedere allo store. Analogamente agli autori basterà un'associazione binaria che mette in relazione l'utente alle proprie credenziale.
- Indirizzo: sempre parallelamente agli autori, anche per gli utenti si vuole tener traccia dell'indirizzo di residenza.
- Dispositivo: un utente non è vincolato a scaricare applicazioni da un unico dispositivo ma può accedere da più dispositivi, ad esempio dal suo smartphone e da un laptop.
- Ricerca: per modellare le ricerche effettuate basterà realizzare un'entità con le informazioni su una ricerca, ovvero la data ed il contenuto cercato.
- Portafoglio: ogni utente è dotato di un portafoglio virtuale all'interno dello store attraverso il quale avrà la possibilità di effettuare acquisti. Per ovvi motivi il portafoglio sarà univoco per ogni utente.

- Giftcard: riallacciandoci al portafoglio appena trattato, l'utente può aumentare il proprio saldo attraverso le giftcard . Non c'è nessuna limitazione riguardo il numero di giftcard utilizzabili ma siccome ognuna può essere riscattata un'unica volta è necessario memorizzare la data dell'utilizzo.
- Carta di credito: in aggiunta al saldo virtuale appena menzionato gli utenti hanno la possibilità di fare acquisti anche mediante le carte di credito. In questo caso ogni utente può collegare più carte di credito le quali non sono a loro volta univocamente associate ad un utente.
- Acquisto: rimanendo nel dominio degli acquisti, ogni utente ha la possibilità di acquistare applicazioni a pagamento oppure di effettuare acquisti sempre all'interno di applicazioni. Per monitorare lo storico degli acquisti sarà necessario tracciarne la data e l'orario.
- Download: ovviamente un utente può scaricare applicativi all'interno dello store. Per memorizzare questi download abbiamo preferito modellare un'entità memorizzando anche in questo caso la data per avere uno storico.
- Recensione: ogni utente ha la possibilità di lasciare recensioni mediante una valutazione numerica ed un commento. In particolare abbiamo deciso che ogni utente può recensire una sola volta un'applicazione in una determinata versione, solo quando verrà rilasciata una nuova versione l'utente potrà rilasciare una nuova recensione.
- Segnalazione Bug: sempre in merito alle applicazioni, gli utenti hanno la possibilità di riportare bug agli autori per facilitare l'aggiornamento delle loro applicazioni. Similmente alle recensioni, una segnalazione bug sarà identificata dall'app in questione e dalla versione nella quale l'utente ha riscontrato il malfunzionamento.

In seguito alle considerazioni appena effettuate lo schema del dominio degli utenti è stato così realizzato:

3.3 Sviluppo delle applicazioni

Per quanto riguarda il dominio delle applicazioni, esse si possono distinguere a seconda di determinate caratteristiche, ovvero: la presenza di pagamenti all'interno delle stesse, il fatto che siano verificate o meno piuttosto che il loro download sia a pagamento. Nonostante questa differenziazione, nessuna caratteristica esclude le altre, perciò la generalizzazione dell'entità applicazione sarà sovrapposta. Inoltre, ci è risultato opportuno, per alleggerire lo schema, non introdurre sotto entità prive di relazioni e/o attributi, il che significa che la generalizzazione proposta avrà una copertura parziale. Una volta definite in modo chiaro le applicazioni passiamo alla definizione delle altre entità associate ad esse. Innanzitutto, ogni applicazione possiede dei requisiti hardware/software (es. specifico sistema operativo, piuttosto che un certo spazio in memoria) affinché possa essere installata su un dispositivo. Proseguendo, ogni applicazione può essere rilasciata in più versioni le quali possono essere identificate univocamente mediante il numero della versione e l'identificativo dell'app. Siccome le versioni possono presentare malfunzionamenti sarà necessario modellare un'entità supporto, univoca per ogni applicazione. Per quanto riguarda le segnalazioni dei bug, esse sono relative ad una specifica versione di un'applicazione. Sempre relativamente ad una versione, è possibile lasciare una singola recensione per utente quindi, anche in questo caso, sarà identificabile mediante versione ed applicazione in questione. Circa la suddivisione di app in categorie, un'applicazione può appartenere a più di una categoria (ad esempio può essere inquadrata sia come gioco sia come applicazione sportiva ecc...). Ovviamente, inoltre, ogni applicazione può essere scaricata, quindi anche in questo caso ricorriamo ad un'entità download che è definita, oltre che dall'applicazione scaricata, anche dalla versione in questione. Infine per gestire gli acquisti delle applicazioni che ne prevedono almeno uno,

ovvero quelle a pagamento e quelle con pagamenti interni modelliamo un'entità acquisto per gestirne uno storico.

Concludiamo riportando lo schema risultante dalle osservazione appena fatte in merito al dominio delle applicazioni.

3.4 Schema concettuale finale

Nella pagina successiva mostriamo come appare lo schema generale del database dopo aver collegato tra di loro i vari ambiti. Sono presenti anche alcune relazioni che fungono da “ponte” tra i vari ambiti che non erano state menzionate prima, nei singoli ambiti.