PROGETTAZIONE DI BASI DI DATI CON MODELLI DI NUOVA GENERAZIONE

A.A. 2014/2015

URL Shortener

Prof. Fabio Fumarola

**Gruppo 1:**

**Fabiola Bubici; mat. 601565;** [**fabby\_b93@hotmail.it**](mailto:fabby_b93@hotmail.it)

**Caterina Scavo ; mat.606290; katia.sc@hotmail.it**

**Vincenzo Molfetta; mat.608163;** [vincyfetta91@libero.it](mailto:vincyfetta91@libero.it)

**Raffaele D’ Alessandro; mat. 611054; dalessandroraffaele1993@gmail.com**

**INDICE**

1. Storia e motivazione del contesto 3

2. Analisi delle funzionalità 4

3. Scenari Procedurali 5

4. Analisi dei costi delle operazioni 8

5. Scelta dei Data Base NOSql 9

6. Modello dei dati 11

7. Statistiche 12

8. Grafico 13

* 1. **Storia e motivazione del contesto**

L’URL Shortener è un tecnica utilizzata nell’ambito del web che si occupa dell’abbreviazione degli URL lunghi in URL brevi, questi ultimi nel momento in cui vengono utilizzati rimandano alla pagina relativa al long URL. Ci sono diversi servizi web che offrono questa opportunità ed il primo a nascere e a portare al successo lo shortener URL fu **TinyURL nato nel 2002, che venne poi soppiantato nel 2008 da Bit.ly.**

L’algoritmo utilizzato per la generazione non è fisso, ogni servizio ne può implementare uno proprio, ma solitamente viene utilizzato quello presente sul web.

Venne realizzata questa tecnica per poter ottimizzare gli spazi, infatti inizialmente venne realizzata per poter gestire al meglio URL molto lunghi, ma questa necessità di utilizzare URL molto più brevi aumentò ancor di più a causa dell’avvento nel 2006 del social network Twitter ed altri microblog che permettono di scambiare messaggi con uno spazio limitato(140 caratteri). Inoltre questa tecnica è utili per monitorare le statistiche relative ai click effettuati e alla geolocalizzazione dagli utenti.

Viene utilizzato un metodo che si occupa di rilevare la presenza di termini inappropriati nella generazione di uno short URL personalizzato. Ciò viene fatto effettuando un confronto dello short URL inserito con l’elenco dei termini sospetti inseriti in un apposito file. Se il dominio inserito appartiene ad un dominio sospetto non viene generato lo short URL e viene visualizzato un messaggio di errore. Nel nostro programma il file al cui interno sono presenti tutti i termini sospetti prende il nome di “UndesiderableWords.txt”.

* 1. **Analisi delle funzionalità**

Utente

L’utente che si trova ad interagire con il nostro sistema è un utente qualsiasi, che ha l’esigenza di creare uno short URL. I casi d’uso che questo può effettuare sono:

1. creazione short URL;
2. creazione short URL personalizzato;
3. visualizzare le statistiche;
4. aprire la pagina web dello short URL;
5. visualizzare il grafico delle statistiche.
   1. **Scenari procedurali**

**Creazione short URL**

|  |  |
| --- | --- |
| **Breve descrizione** | L’utente vuole creare uno short URL inserendo nell’apposita riga il long URL. |
| **Postcondizioni per successo** | Viene generato lo short URL e viene salvato nel database associando al long URL relativo. |
| **Postcondizioni per fallimento** | Viene generato un messaggio di errore che indica che non è stato generato lo short URL, in quanto il long URL inserito non è valido. |
| **Evento innescante** | Necessità di generare uno short URL. |
| **Attore primario** | Utente |
| **Scenario di base** | 1. L’utente inserisce il long URL nell’apposita sezione. 2. Preme il tasto “Shorten URL”. 3. Il sistema dopo aver effettuato i dovuti controlli, genera lo short URL e lo visualizza sullo schermo. Viene salvato nel database associandolo al long URL. |
| **Scenario alternativo: url lungo inserito errato** | 3.Il sistema effettua i dovuti controlli, si accorge che il long URL inserito non è valido e notifica che non può essere generato lo short URL in quanto l’URL non è valido |

**Creazione short URL personalizzato**

|  |  |
| --- | --- |
| **Breve descrizione** | L’utente vuole creare uno short URL personalizzato inserendo nell’apposita riga il long URL. |
| **Postcondizioni per successo** | Viene generato lo short URL personalizzato e viene salvato nel database associando al long URL relativo. |
| **Postcondizioni per fallimento** | Viene generato un messaggio di errore che indica che non è stato generato lo short URL personalizzato, in quanto il long URL inserito non è valido. |
| **Evento innescante** | Necessità di generare uno short URL personalizzato. |
| **Attore primario** | Utente |
| **Scenario di base** | 1. L’utente inserisce il long URL nell’apposita sezione. 2. Inserisce la personalizzazione da aggiungere all’URL generato. 3. L’utente preme il tasto“Custom Shorten URL”. 4. Il sistema dopo aver effettuato i dovuti controlli, genera lo short URL e lo visualizza sullo schermo. Viene salvato nel database associandolo al long URL. 5. Il sistema notifica l’URL personalizzato. |
| **Scenario alternativo: url lungo inserito errato** | 4.Il sistema effettua i dovuti controlli, si accorge che il long URL inserito non è valido e notifica che non può essere generato lo short URL in quanto l’URL non è valido |

**Visualizza le statistiche**

|  |  |
| --- | --- |
| **Breve descrizione** | L’utente vuole vedere le statistiche relative allo short URL. |
| **Postcondizioni per successo** | Viene generato lo short URL e viene salvato nel database associando al long URL relativo. Vengono visualizzate le statistiche relative. |
| **Postcondizioni per fallimento** | Viene generato un messaggio di errore che indica che non è stato generato lo short URL, in quanto il long URL inserito non è valido. Quindi non possono essere visualizzate le statistiche. |
| **Evento innescante** | Necessità di visualizzare le statistiche relative allo short URL. |
| **Attore primario** | Utente |
| **Scenario di base** | 1. L’utente inserisce il long URL nell’apposita sezione. 2. Preme il tasto “Shorten URL”. 3. Il sistema dopo aver effettuato i dovuti controlli, genera lo short URL e lo visualizza sullo schermo. Viene salvato nel database associandolo al long URL. 4. L’utente inserisce lo short URL nell’apposita riga. 5. Preme il tasto “Open web page”. 6. Preme il tasto “Stats”. 7. Il sistema visualizza sullo schermo le statistiche relative allo short URL inserito. |
| **Scenario alternativo: url lungo inserito errato** | 3.Il sistema effettua i dovuti controlli, si accorge che il long URL inserito non è valido e notifica che non può essere generato lo short URL in quanto l’URL non è valido |

**Aprire la pagina web dello short URL**

|  |  |
| --- | --- |
| **Breve descrizione** | L’utente vuole aprire la pagina web corrispondente allo short URL. |
| **Postcondizioni per successo** | Viene generato lo short URL e viene salvato nel database associando al long URL relativo. L’utente apre la pagina web corrispondente allo short URL. |
| **Postcondizioni per fallimento** | Viene generato un messaggio di errore che indica che non è stato generato lo short URL, in quanto il long URL inserito non è valido. Quindi non può aprire la pagina web corrispondente allo short URL. |
| **Evento innescante** | Necessità di aprire la pagina corrispondete allo short URL generato. |
| **Attore primario** | Utente |
| **Scenario di base** | 1. L’utente inserisce il long URL nell’apposita sezione. 2. Preme il tasto “Shorten URL”. 3. Il sistema dopo aver effettuato i dovuti controlli, genera lo short URL e lo visualizza sullo schermo. Viene salvato nel database associandolo al long URL. 4. L’utente inserisce lo short URL nell’apposita riga. 5. Preme il tasto “Open web Page”. 6. Il sistema apre la pagina web relativa allo short URL generato. |
| **Scenario alternativo: url lungo inserito errato** | 3.Il sistema effettua i dovuti controlli, si accorge che il long URL inserito non è valido e notifica che non può essere generato lo short URL in quanto l’URL non è valido |

**Visualizzare il grafico delle statistiche**

|  |  |
| --- | --- |
| **Breve descrizione** | L’utente vuole vedere il grafico relativo alle statistiche. |
| **Post condizioni per successo** | Viene generato lo short URL e viene salvato nel database associando al long URL relativo. Vengono visualizzate le statistiche relative. |
| **Evento innescante** | Necessità di visualizzare il grafico delle statistiche relative a quello short URL. |
| **Attore primario** | Utente |
| **Scenario di base** | 1. L’utente inserisce il long URL nell’apposita sezione. 2. Preme il tasto “Shorten URL”. 3. Il sistema dopo aver effettuato i dovuti controlli, genera lo short URL e lo visualizza sullo schermo. Viene salvato nel database associandolo al long URL. 4. L’utente inserisce lo short URL nell’apposita riga. 5. Preme il tasto “Viewgraph”. 6. Il sistema visualizza su un’altra pagina le statistiche relative allo short URL inserito. |

* 1. **Analisi dei costi delle operazioni**

Le operazione da effettuare per ***generare un urlshort*** sono:

1. Inserire il Long Url nell’apposito campo: InputText;
2. Cliccare il pulsante “Shorten URL”;
3. Allo stesso tempo avviene il salvataggio dell’url nel database e comparirà affianco al pulsante “Shorten URL” appunto lo short URL creato.

**2 operazioni: 2 scritture**

Le operazioni da effettuare per ***generare un urlShort personalizzato*** sono:

1. Inserire un longurl nell’apposito spazio: Enter long url;
2. Inserire uno shortUrl personale nel secondo spazio: Enteryour short URL;
3. Cliccare il pulsante posto sotto: “Custom ShortenUrl”;
4. Dopo di che accanto al tasto “Custom ShortenUrl” comparirà lo stesso shortner personalizzato da noi immesso e verrà salvato automaticamente nel Database.

**2 operazioni: 2 scritture**

Le operazioni da effettuare per ***aprire la pagina web dello short URL*** sono:

1. Inserire lo short url precedentemente generato nell’ultimo spazio Enter short Url di URL Details;
2. Cliccare il pulsate “Open web page” per avviare la pagina web. La pagina riporterà al long url.

**2 operazioni: 1 scrittura, 1 lettura**

Le operazioni da effettuare per ***visualizzare le statistiche*** sono:

1. Inserire lo short url precedentemente generato nell’ultimo spazio Enter short Url di URL Details;
2. Cliccare il pulsate “Stats” per visualizzare le statistiche che compariranno sotto il pulsante “Stats”.

**1 operazione : 1 lettura**

Le operazioni da effettuare per ***visualizzare il grafico delle statistiche*** sono:

1. Inserire lo short url precedentemente generato nell’ultimo spazio Enter short Url di URL Details;
2. Cliccare il pulsate “Viewgraph” per visualizzare il grafico delle statistiche che apparirà in un'altra pagina.

**1 operazione : 1 lettura**

* 1. **Scelta del database NOSql**

Il modello di database NOSql scelto da noi è Redis.

Redis è un database non relazionale nato nel 2009 per mano di Salvatore Sanfilippo, ora è supportato da Pivotal.

È un NoSql di tipo Key-value.

**Gli svantaggi di un database key-value sono:**

* non hanno filtri di query complesse;
* tutti i join devo essere fati nel codice;
* non ci sono foreignkey, ovvero vincoli di integrità referenziale di chiavi esterne;
* non ci sono trigger.

**I vantaggi di un database key-value sono:**

* esecuzione di query efficienti (molto prevedibili in termini di prestazioni);
* database facile da distribuire in un cluster;
* ricerca dei dati attraverso la chiave
* nessun oggetto-relazionale miss-match;
* non ha nessun tipo di oggetto relazionale.

**Perché abbiamo scelto Redis:**

Come soluzione in-memory, Redis è ottimo per memorizzare dati transitori. La lettura e la scrittura dei dati sono di breve durata, ma si verificano con alto volume e frequenza.

Redis offre anche meccanismi configurabili per persistenza. Tuttavia, l'aumento della persistenza tenderà ad aumentare e diminuire la latenza di throughput (il termine indica la banda effettiva misurata in un certo periodo, tenuto conto del flusso dei dati e del percorso di instradamento).

Redis supporta cinque diverse strutture dati permettendo così di gestire entità quali: insiemi ordinati e dei dati di serie temporali. Un ulteriore punto di forza è nella varietà di linguaggi di programmazione supportati. Inoltre lavora completamente in RAM.

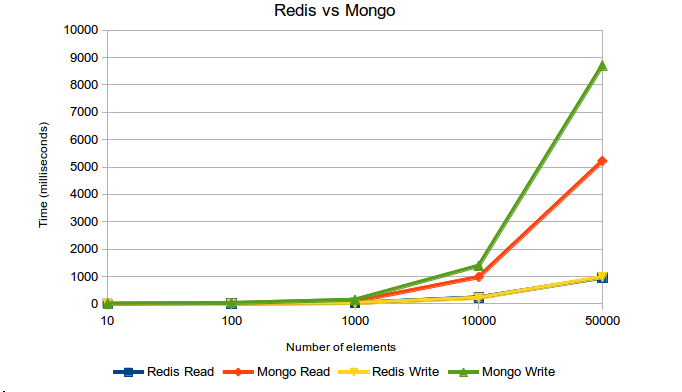
Redis non è altro che un server TCP che utilizza il modello client-server ed implementa quello che è chiamato un protocollo di tipo Request/Response o per semplicità telnet-like.

Ciò significa che normalmente una richiesta viene eseguita mediante la seguente procedura:

1. Il client invia una query al server (comando) ed attende sul socket una risposta.

2. Il server elabora il comando e invia la risposta al client.

**Mongo VS Redis:** entrambi sono open source, Mongo è un database NoSQL document oriented. In entrambi si possono effettuare operazioni di lettura e scrittura. Sono state messe a confronto le due operazioni in entrambi i database, mettendo a disposizione lo stesso numero di dati e macchine con le stesse caratteristiche. Possiamo notare che per un numero piccolo di dati il tempo di esecuzione è quasi uguale, ma nel momento in cui vengono elaborati un gran numero di dati Redis ha una notevole superiorità per entrambe le operazioni. Prendendo in considerazione il CAP Theorem possiamo dire che sia Redis che MongoDB sostengono la consistenza sui dati.



**Cassandra VS Redis:** entrambi sono open source. Cassandra è un database column-family store. In entrambi si possono effettuare operazioni di lettura e scrittura, ma confrontando il throughput e avendo a disposizione lo stesso numero di dati e macchine con le stesse caratteristiche, per un numero piccolo di dati il throughput è quasi uguale, ma nel momento in cui abbiamo un gran numero di dati in scrittura e lettura Cassandra ha un throughput maggiore rispetto a Redis. Sulla base del CAP Theorem possiamo affermare che Redis garantisce consistenza dei dati, mentre Cassandra ne aumenta la disponibilità.

**HBase VS Redis:** entrambi sono open source. HBase è un altro database column-oriented. Volendo confrontare anche qui il throughput, possiamo dire che per un numero piccolo di dati il throughput è quasi uguale, ma nel momento in cui vengono elaborati un gran numero di dati Redis ha un throughput maggiore per entrambe le operazioni di scrittura e lettura. HBase a differenza di Cassandra garantisce la consistenza dei dati.

Nella nostra scelta si è privilegiato un database che garantisca la consistenza dei dati e alte performance in fase di scrittura e lettura.

* 1. **Modello dei dati**

La struttura dati del nostro database è composta in questo modo:

* shortUrl che è la nostra chiave formata da [www.sht.com/SHORT\_GENERATO](http://www.sht.com/SHORT_GENERATO) oppure dallo short URL personalizzato definito dall'utente;
* clicks, numero di click con la relativa data, utilizzato per le statistiche;
* longUrl , ovvero url di partenza a cui viene applicato l’algoritmo per la generazione dello shorturl.

**Esempi della struttura del nostro DB :**

Key : www.sht.com/MUdKC

Value:

{"shortUrl":"www.sht.com/MUdKC","clicks":[],"longUrl":"http://www.aaaaaaaaaaaaaaaaaaaaaaaaabbbbbbbbbbbbbbbbbbbbbb.com"}

Esempio 2 del nostro DB:

Key : www.sht.com/EE4SK

Value:

{"shortUrl":"www.sht.com/MUdKC","clicks":[],"longUrl":"http://www.aaaaaaaaaaaaaaaaaaaaaaaaabbbbbbbbbbbbbbbbbbbbbb.com"}

Key :www.sht.com/QmsaY

Value con i click:

{"shortUrl":"www.sht.com/QmsaY","clicks":[{"Date":"05 9 2015"},{"Date":"05 9 2015"}],"longUrl":"http://www.amazon.com"}

* 1. **Statistiche**

Abbiamo ritenuto opportuno implementare e rilevare le seguenti statistiche per ciascuno short URL specificato:

- il numero totale dei click effettuati su quello short URL;

- il numero totale dei click effettuati in data odierna;

- il numero totale dei click effettuati durante il mese corrente;

- il numero totale dei click effettuati durante l'anno corrente;

- il long URL più cliccato dagli utenti;

- il numero totale di click effettuati sul long URL maggiormente cliccato dagli utenti;

- geolocalizzazione: il paese e la città di provenienza della richiesta;

* 1. **Grafico**

Sotto viene riportato un esempio di grafico generato dall'inserimento , che rappresenta il numero di click nel mese corrente di un determinato short URL. Abbiamo sull'asse delle ascisse la data temporale espressa in giorni, mentre sulle ordinate il numero di click contabilizzati.

Key :www.sht.com/QmsaY

View Graph :

