I.I.S. “Levi-Ponti”

Esami di stato 2015

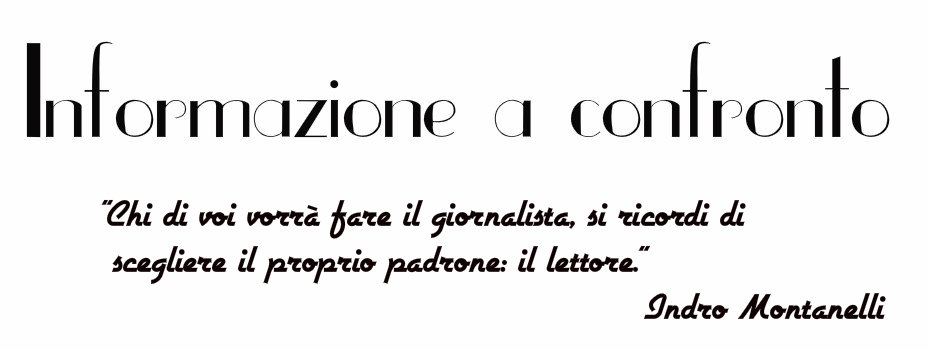
Alunno: **Taschin Federico**

Classe V, sez. A INF

Anno scolastico 2014-2015

|  |
| --- |
|  |

**Progetto di ambito tecnico-scientifico**

****

**Indice dei contenuti:**

**1. Informazione**

**2. Informazione a confronto**

* 1. **Obiettivi**

**2.2 Criticità:**

* **Come recuperare le news**
* NewsFeed (RSS protocol)
* Metalinguaggio
* Parsing dei siti
* **Come organizzare le news**
* Sistema di punteggi
* Appoggio ad appositi servizi esterni

**2.3 WBS**

**2.4 Struttura:**

* **Front-end HTML e PHP**
* **Database**
* **Recupero delle news**
  1. **Sviluppo:**
* **Recupero delle news**
* **Divisione in moduli**
* **NewsFeed**
* Recupero news attraverso il protocollo RSS
* **Metalinguaggio**
* Parser (automa a stati finiti)
* Definizione della grammatica
* **News parsing**
* Parsing delle pagine
* Procedura di estrapolazione dei dati
* Inserimento nel database
* **Point updating**
* Update ciclico del punteggio di ogni articolo
* Appoggio ai servizi appositi di Facebook, Twitter e LinkedIn
* **Database**
* Diagramma ER

**2.6 Sviluppi futuri**

* Aggiunta commenti da parte degli utenti
* Indicizzazione delle ricerche effettuate dagli utenti per migliorare la classificazione degli articoli

1. **Informazione**

L’informazione è sempre stata, e continua ad essere, di importanza fondamentale nella società. Dal latino “informare” significa dare forma, perché è proprio questo il ruolo che l’informazione ha sempre svolto: dare forma alla mente ed alla coscienza. L’informazione non è un fatto volontario ma un processo inevitabile e continuo che coinvolge ogni individuo attraverso gli stimoli ed i dati che il mondo esterno e la società gli trasmette. Non esiste quindi il “se”, ma solo il “come”.

Vi sono due tipi di processi che possono portare un individuo a questo “dare forma” alla propria coscienza rispetto ad un argomento: l’informazione consapevole e quella passiva. L’informazione consapevole è l’informazione volontaria, critica ed obiettiva, cui scopo è non apprendere una verità precostituita e pronta, ma far propri i molteplici aspetti di un problema, di una situazione, analizzandoli ed assimilandoli, per giungere ad una conclusione propria e consapevole. L’informazione passiva invece è l’informazione recepita senza il filtro della propria coscienza e volontà di approfondire. E’ la “verità”, che verità non è, ma che viene recepita come valida, appresa da una fonte ma non confrontata con altre. La molteplicità delle fonti è un elemento chiave del processo che può portare ad un’informazione attiva e consapevole. Non a caso il primo passo di tutti i regimi autoritari è stato quello di eliminare il pluralismo ed assumere quindi il controllo dell’informazione, sfruttando a pieno il significato del termine e “dando forma” alle menti ed alle coscienze della popolazione, divenendo gli unici detentori della “verità”.

1. **Informazione a confronto**

Informazione a confronto è un progetto che nasce con l’intento di facilitare una visione globale del mondo dell’informazione on-line, che sempre più sta prendendo piede nella società moderna.

* 1. **Obiettivi**

L’obiettivo di Informazione a confronto è realizzare un’interfaccia web che permetta agli utenti di rimanere aggiornati sulle informazioni riguardanti gli argomenti più importanti come politica, economia, cronaca, cultura, o di svago come sport e intrattenimento. Questo, offrendo un ampio panorama di informazione con articoli sempre nuovi dalle maggiori testate on line, classificate secondo la loro popolarità ed importanza. Essendo il lavoro di ricerca e classificazione degli articoli molto lungo e ampio, un sito che offrisse questo servizio necessiterebbe di una redazione composta da più persone. Per questo motivo ho deciso di rendere Informazione a confronto **automatizzato**: gli articoli verranno recuperati in automatico da un software da me creato senza l’intervento di un operatore, il quale dovrà soltanto svolgere un lavoro di controllo sul prodotto.

* 1. **Criticità**

Nel sviluppare il progetto ho dovuto risolvere diversi punti critici, riguardanti l’automazione del processo che gestisce il flusso in entrata di dati che è alla base del funzionamento del progetto

**Come recuperare le news:**

Il primo problema che emerge da un’analisi è sicuramente come recuperare gli articoli, che vengono pubblicati durante il giorno dai siti. Il protocollo **RSS** svolge proprio la funzione di standardizzare i flussi di informazioni e notizie attraverso il web, ed è il maggiormente utilizzato da quotidiani on-line, siti, blog etc. Basterà dunque sviluppare un modulo del software che si occupi di ricevere costantemente questi dati (le **NewsFeed**).

Le NewsFeed però hanno un limite: non contengono infatti l’intero articolo, ma soltanto una parte di esso, come titolo, autore, sommario ed il link all’articolo. Ricevere le NewsFeed non basterà quindi a raggiungere l’obiettivo, ma occorrerà recuperare l’intera pagina web a cui punta la NewsFeed e selezionare le informazioni necessarie.

Il codice html entro cui sono racchiuse le informazioni utili è però diverso sito per sito, e non è possibile quindi sviluppare un modello di automazione sufficiente a recuperarle. Ho deciso quindi di creare un **metalinguaggio** che funga da guida al software nella ricerca dell’informazione. Le espressioni scritte in questo metalinguaggio da chi amministra il sito indicheranno al software le operazioni di selezione e, in caso sia necessario, modifica sulla pagina web recuperata. Ho sviluppato quindi un modulo del software che si occupa di interpretare attraverso un **automa a stati finiti** le espressioni ed eseguirle.

Per il parsing della pagina html recuperata con il protocollo RSS necessito di utilizzare una libreria che costruisca il **DOM** (*Document Object Model*) di essa, in modo poi da poter eseguire le operazioni definite dal metalinguaggio.

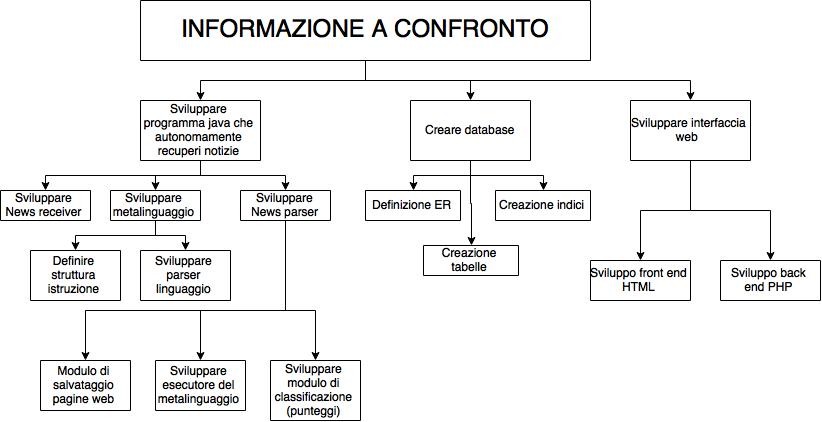
**Come organizzare le news**

Una volta risolto il problema del recupero, bisogna però affrontare quello dell’organizzazione degli articoli: la suddivisione in categorie e la classificazione.

La suddivisione delle categorie è svolta a monte in quanto i quotidiani on-line possiedono Feed suddivisi per categoria.

La classificazione degli articoli è invece più complessa: ho deciso di utilizzare un sistema di punteggi basati sulle condivisioni della notizia tra i maggiori social network quali Facebook, Twitter e LinkedIn. Per questo posso appoggiarmi ai servizi offerti dagli stessi social, che permettono di ricevere il numero di condivisioni di un determinato link. Una richiesta GET al servizio permetterà quindi di conoscere il numero di condivisioni.

**2.3 Work Brakedown Structure**



**2.4 Struttura**

**Front-end HTML**

L’interfaccia grafica del sito è sviluppata con l’utilizzo dei linguaggi HTML e JavaScript. Per rendere la grafica migliore ed ottimizzare la visualizzazione su dispositivi con schermi di dimensioni diverse ho utilizzato **Bootstrap 3**, un framework che offre diverse classi css e script per la realizzazione di pagine web dinamiche ed articolate.

Il template di base di Informazione a Confronto è semplice e intuitivo: il logo del progetto, una barra di navigazione da cui si accede alle sezioni, il pannello centrale, dove vengono visualizzati gli articoli ed una sezione laterale che contiene le funzionalità quali ricerca di un articolo per argomenti, visualizzazione degli argomenti principali.

E’ possibile visualizzare gli articoli divisi per categoria, cliccando sul corrispettivo pulsante nella barra di menu, o ricercare uno o più argomenti particolari tramite la sezione ricerca argomenti. Nella homepage invece sono visualizzati gli articoli più rilevanti di tutte le categorie, per un massimo di 4 a categoria.

**PHP**

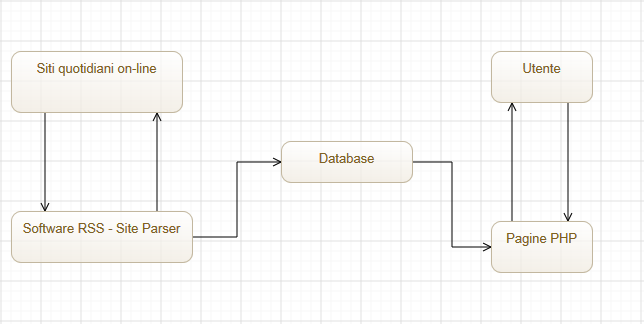
Le parti più significative della struttura web d Informazione a Confronto sono scritte in linguaggio PHP. Queste permettono all’utente di visualizzare gli articoli e i risultati delle sue ricerche sempre aggiornati, e compongono una struttura che lavora in maniera indipendente dai contenuti effettivi da visualizzare, funzionando quindi in maniera solida ed efficace. Comprendono la homepage, che seleziona dal database le notizie più rilevanti, le pagine di categoria, la sezione “tag più popolari”, le ricerche degli utenti.



*Homepage di Informazione a Confronto*

**Database**

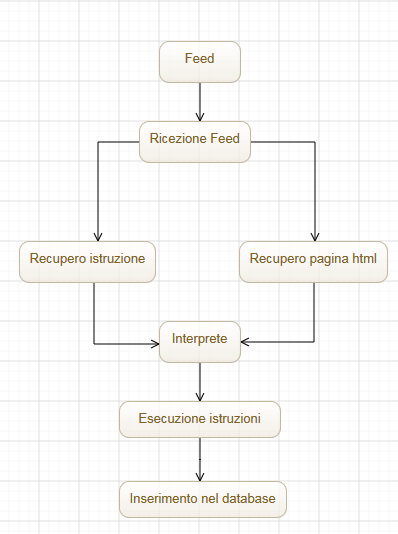
Tra il sito ed il software che si occupa di recuperare le notizie vi è un database. Esso contiene i dati utili al funzionamento del software come link ai feed, categorie ed istruzioni scritte in metalinguaggio, e gli articoli recuperati dal software. Interagiscono in due maniere differenti con esso il sito ed il software: il sito accede al database per recuperare gli articoli da visualizzare, il software invece vi accede per inserirli.



*Interazione software – database –sito*

**Recupero delle notizie**

Il recupero delle notizie è diviso in quattro fasi: la prima fase di ricezione delle news feed attraverso il protocollo RSS, la seconda di parsing della pagina html relativa all’articolo e dell’istruzione collegata, la terza di interpretazione dell’istruzione ed esecuzione sulla pagina html ricevuta, e la quarta di inserimento nel database.



*Fasi di recupero notizie*

**2.5 Sviluppo**

**Recupero delle news**

Il software che si occupa di recuperare le notizie ed inserirle nel database è sviluppato in java e rimane attivo continuamente nel server, eseguendo le procedure di ricezione, parsing e storage ogni ora.

**Divisione in moduli**

Per sviluppare un software consistente ed ordinato ho diviso il codice Java in moduli. Ogni modulo ha una funzione ben precisa, e, secondo la filosofia del “divide et impera”, si occupa di un problema preciso del vasto insieme di funzionalità che il software deve svolgere. Di seguito i moduli più significativi:

**News feed**

Il primo modulo nell’activity diagram ad entrare in funzione è quello adibito alla ricezione delle news feed. Ogni ora si avvia la procedura di lettura dei feed dal database, e per ogni feed vengono ricevuti i FeedMessages. Ogni feedMessage contiene i dati riassuntivi ed il link di un articolo. Questi sono scritti in un file xml che viene ricevuto e parsato. Ogni FeedMessage ricevuto viene passato al secondo modulo.

**Metalinguaggio**

Questo modulo si occupa di parsare l’istruzione relativa all’articolo ricevuto.

Per prima cosa recupera dal database l’istruzione relativa al sito da cui è stato ricevuto il FeedMessage. Successivamente si procede al parsing dell’istruzione attraverso un automa a stati finiti. Questo prevede una tabella di transizione che gestisce le procedure necessarie al parsing. Ho utilizzato un automa a stati finiti in quanto questa architettura permette un certo livello di aggiornamento nel caso si volesse in futuro modificarne la sintassi.

Una istruzione è così definita (dove il simbolo “:-” sta per “è composto/a” ed il simbolo “|” sta per “o”):

Istruzione :- Attributo=[Sottoistruzioni]

| Attributo=[Sottoistruzioni] Azioni

Sottoistruzioni :- Sottoistruzione

| Sottoistruzioni + sottoistruzione

Sottoistruzione :- (Verbo espressione)

| (Verbo espressioni)

Espressioni :- Espressione

|Espressioni IN espressione

Espressione :- TAG <nome\_tag> <intervallo>

| TAG <nome\_tag><intervallo> Condizioni

Condizioni :- Condizione

| Condizioni <operatore> Condizione

Condizione :- <attributo> <operatore> <valore>

Azioni :- Azione

| Azione Azioni

Azione :- ACTION <nome\_azione> Condizione ON <target>

**Esempio di metalinguaggio:**

BODY=[(TAKE TAG ‘span’ ALL WHERE class = ‘articleBody’ IN TAG ‘BODY’ FIRST)]

TITLE=[(TAKE TAG ‘h1’ FIRST WHERE class=’headline name’)]

ACTION DELETE\_TAGS ‘script’ ALL ON BODY;

**Parsing delle pagine**

Ricevuto il feed message e parsata l’istruzione si procede al parsing della pagina. Per fare ciò ho utilizzato una libreria, Jsoup, che si occupa di parsare il testo html trasformandolo nel modello ad oggetti. Su di questo sarà possibile effettuare le operazioni di selezione.

**Esecuzione del linguaggio**

L’interprete del metalinguaggio si occupa di eseguire le istruzioni relative all’articolo. Attraverso una serie di cicli esegue operazioni di selezione sul DOM della pagina HTML parsata. Eseguite queste esegue le azioni definite nell’istruzione.

**Inserimento nel database**

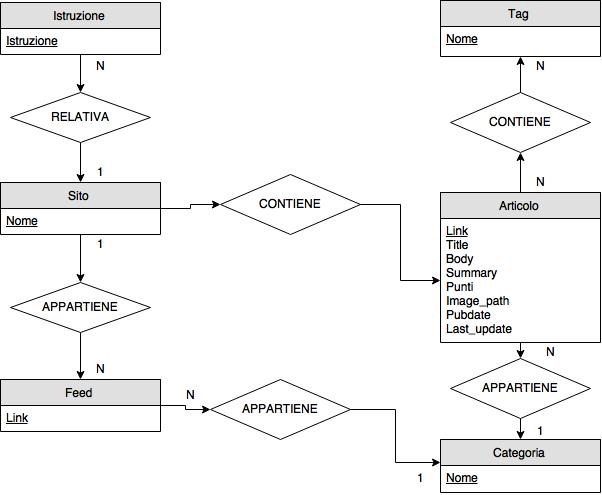
Una volta recuperate le informazioni necessarie i dati vengono inseriti nel database dal programma. Un secondo ciclo controlla che i dati inseriti siano coerenti.

**Point updating**

Ogni due ore viene eseguita poi l’operazione di update del punteggio. Questa consiste nel inviare una richiesta get ai servizi appositi di Facebook, Twitter e LinkedIn per ricevere i dati relativi alle condivisioni. Verrà poi fatto un update sul database SQL aggiornando il punteggio.

**Database**

Il database è così strutturato:



Sono presenti due indici di tipo FULLTEXT sui campi Title e Summary di ARTICOLO.

* 1. **Sviluppi futuri**

Informazione a Confronto è stato un progetto che mi ha coinvolto in modo particolare. Sia per il tema trattato, sia per gli ambiti tecnici che ricopre, vi ho lavorato con grande impegno e dedizione e credo sia qualcosa che possa essere portato avanti, forse non così come l’ho sviluppato, ma ritengo il modello una buona base per sviluppare, in un futuro non molto remoto, una piattaforma molto più articolata e potente, che possa essere di aiuto a chiunque voglia informarsi attivamente. Alcuni aspetti intendo migliorare:

In primo luogo la componente “social” del progetto, in modo da offrire oltre che un sito di informazione, una piattaforma di discussione ed approfondimento.

In secondo luogo sviluppare un sistema più potente per il confronto delle notizie, capace di collegare autonomamente notizie collegate tra loro e individuare notizie sullo stesso argomento in maniera più efficace e meno rudimentale.