**APPUNTI TSW**

Ipertesto= insieme di documenti collegati da monodirezionali (iperlink) e può essere visto come un grafo:

* Ogni documento costituisce un nodo del grafo
* Lo spostamento quindi da un nodo all’altro avviene questi ponti (archi) detti link ipertestuali

La potenzialità dell’ipertesto (e del WWW) è che può svilupparsi in modo non lineare.

Quando poi, come nel caso del WWW, oltre a testi si hanno collegamenti monodirezionali anche verso immagini o video si parla di ***ipermedia***

**Protocollo HTTP**

Si basa sulla comunicazione tra un server che memorizza le informazioni e le pagine da visualizzare, queste poi vengono richieste da un client. Quindi si basa su un sistema di request e response:

* Request:
  + http method
  + URL della pagina a cui si vuole accedere
  + Parametri formali (tipo argomenti del metodo chiamato)
* Response:
  + Un codice stato che indica la recezione della richiesta
  + Tipo di contenuto della risposta
  + Contenuto della risposta

Per velocizzare la comunicazione tra client e server si utilizza il pipelining, ovvero il client invia molteplici richieste al server senza aspettare la loro recezione, così da non creare un collo di bottiglia nell’aspettare la recezione di una singola richiesta (introdotto da http 2.0)

Metodi di http:

* GET
  + Metodo semplice che richiede un dato al server
* POST
  + Usato dall’utente per sottomettere dati sottoforma di form HTML ad un app server, e contemporaneamente richiede un dato al server

Parallelamente al sistema request/response, uno strumento dell’http è il sistema a cookie che aiutano a sorpassare il limite dell’html, ovvero quello di essere stateless. La funzione dei cookie è quindi quella di mantenere uno stato di un client nel server. Un cookie è composto da:

* Key = identificatore del cookie all’interno di un domain\_path
* Value = valore associato al cookie (stringa di max 255 caratteri)
* Path = posizione all’interno dell’albero di un sito
* Domain
* Max Age
* Version
* Secure

I cookie utilizzano due header:

* Set-Cookie: header della risposta, il client può memorizzarlo e rispedirlo alla prossima richiesta
* Cookie: header della richiesta, il client manda il cookie affidato nella risposta dal server, e il server analizza il cookie per

AJAX serve per poter effettuare chiamate asincrone, ovvero al client non è necessario refreshare la pagina per avere un aggiornamento

Siti statici e siti dinamici:

* Sito statico: server non interattivo
* Sito dinamico: basato su una programmazione dinamica con interazione e aggiornamenti attraverso servlet

Servlet: applicazione lato server

Deploy: aggiungere un progetto/risorsa ad un server

UnDeploy: rimuovere un progetto/risorsa da un server

**Web Dinamico e Servlet**

L’html utilizzato fin’ora è una rappresentazione di un sito web **statico** per via di un insieme di contenuti prefissati staticamente sul sito pur permettendo all’utente di navigare in modo dinamico (modulare) i contenuti attraverso i link ipertestuali forniti

**Persistenza e Storage**

La persistenza dei dati è strettamente legata ad una base di dati, nel caso di un sito web, è necessario mappare un oggetto manipolato dall’utente ad una base di dati relazionale.

L’accesso più comune ad una base di dati è quello ad accesso diretto, ovvero utilizzando JDBC (includendo query sql in java) attraverso la CLI (Call Level Interface), ovvero un interfaccia che si contrappone tra l’applicazione java e la base di dati