

Glossario

Versione 1.0

Redazione Miotto Fabio

Sorgato Mattia

??

Verifica Jack??

Responsabile ??

Uso Esterno

Lista di distribuzione Prof Vardanega

Prof Cardin

CoffeeStrap

Descrizione

Questo documento contiene definizioni delle parole che possono portare ad ambiguita’, utilizzate all’interno dei documenti redatti dal gruppo Aperture Software.

Diario delle modifiche

|  |  |  |  |
| --- | --- | --- | --- |
| **Versione** | **Data** | **Autore** | **Descrizione modifica** |
| 1.2.0 | 2013-12-19 |  | Approvazione finale del documento |
| 1.1.1 | 2013-12-17 | secondo ver | Verifica del documento |
| 1.1.0 | 2013-12-16 | primo ver | Verifica del documento |
| 1.0.6 |  | mattia | Inserimento termini |
| 1.0.5 |  | mattia | Inserimento termini |
| 1.0.4 |  | mattia | Inserimento termini |
| 1.0.3 |  | miotto | Inserimento termini |
| 1.0.2 |  | miotto | Inserimento termini |
| 1.0.1 |  | miotto | Inserimento termini |
| 1.0.0 | da guardare su gantt | miotto | Creazione struttura iniziale documento |

Indice

1. Introduzione

## 1.1 Scopo del documento

Questo documento raccoglie tutti i termini che sono sconosciuti al lettore esterno o che possono portare ad ambiguità. Per ciascun termine viene riportata a fianco una breve definizione che ne chiarisce il significato.

1.2 Riferimenti

1.2.1 Normativi

Norme di progetto: Norme di progetto v1.2.0

1.2.2 Informativi

Wikipedia: <http://www.wikipedia.org/>

OkPedia: <http://www.okpedia.it/>

W3C: [http://www.w3.org/](http://www.w3.org/Glossary)

Techterms: <http://www.techterms.com/>

ISO: <http://www.iso.org/iso/home/about.htm>

Francesco Ranzato, Appunti di programmazione oggetti, Edizione II.

Elenco definizioni

**Stub:**

**Driver:**

**Ambiente:** consiste sia del sistema hardware che di quello software sui quali è

stato pianiﬁcato l’utilizzo del prodotto software sviluppato;

**Framework:** struttura di supporto su cui un software puo’ essere progettato e realizzato. Alla base di un framework sono sempre presenti delle **librerie** **di codice** utilizzabili con uno o piu’ **linguaggi di programmazione**; esse sono spesso corredate da una serie di strumenti di supporto allo sviluppo **software**, come ad esempio un **IDE** o un **debugger**, o altri strumenti ideati per aumentare la velocita’ di sviluppo del prodotto finito.

Lo scopo di un framework e’ quello di far risparmiare allo **sviluppatore** la riscrittura di **codice** gia’ scritto precedentemente per fini simili. La necessita’ di questo strumento si e’ venuta a creare quando le **interfacce** utente sono diventate sempre piu’ complesse ed e’ cominciata ad aumentare la quantita’ **software** con funzionalita’ secondarie simili.

**File di descrizione:** file scritto con linguaggio DSL dallo sviluppatore, e serve per generare pagine di tipo collection index e document show

**File di configurazione:**

**File system:** meccanismo con il quale i **file** sono posizionati e organizzati su un dispositivo di archiviazione.

**Upload:** caricamento, ovvero il processo di trasmissione di un file da un **client** ad un **server**.

**Form:** termine usato per indicare l’interfaccia di una applicazione che consente ad un utente di inserire e inviare ad un server dei dati liberamente digitati dallo stesso utente.

**Filtro:** componente che ha il compito di selezionare una fonte in ingresso secondo dei criteri, in modo da avere un risultato finale utile all’utente.

**Chiave:**

**Document:** istanza, o record, di un database in MongoDB. esso e’ costituito da un insieme di chiavi con il rispettivo valore

**template:** documento nel quale e’ rappresentata una struttura generica o standard dove ci sono spazi temporaneamente bianchi da riempire in seguito. In italiano viene indicato come scheletro o modello di base.

**utente sviluppatore:** persona che utilizza il framework MaaP.

**utente business:** persona alla quale interessano le pagine web create dal framework Maap; esso puo’ consultare o operare su queste pagine in base a degli specifici scopi o interessi.

**utente business autenticato: utente business** che ha effettuato con successo la procedura di login verso il sistema che contiene le pagine generate da MaaP

**Utente business autenticato amministratore: utente business autenticato** che ha funzionalita’ aggiuntive rispetto a degli utenti business autenticati.

**amministratore:** classe di utenza privilegiata rispetto ad **utente** normale, ovvero ha funzionalita’ in piu’ che gestiscono un sistema.

**default:** valore o un’azione standard che caratterizza qualsiasi aspetto globale a meno di un cambiamento improvviso.

**Logout:** procedura inversa dell’autenticazione, ovvero tramite questa azione si esce da un sistema e si abbandona la connessione stabilita con quest ultimo.

**permesso:** o autorizzazione, indica quali funzionalita’ possono competere a determinate cerchie di persone o utenti.

**Compatibilita’:** in informatica indica la proprieta’ di due o piu’ computer che, scambiandosi dei dati, accettano gli stessi programmi senza fare modifiche ad essi.

**ambiguo:** termine usato per indicare una cosa che non e’ chiara, ovvero presenta un doppio senso o un fraintendimento.

**Libreria di codice:** insieme di **funzioni** o **strutture dati** predisposte per essere collegate ad un **programma software** attraverso opportuno collegamento.

**Funzione (software):** particolare costrutto **sintattico**, in qualche **linguaggio di programmazione**, che permette di raggruppare, all'interno di un **programma**, una sequenza di istruzioni in un unico blocco di istruzioni espletando così una determinata e in generale più complessa operazione, azione o elaborazione sui dati del **programma** stesso in modo tale che a partire da determinati **input** restituisca determinati **output**.

**Input:** tradotto letteralmente in “immettere”. Sequenza di dati o informazioni, immessi per mezzo di una **periferica,** detta appunto di *input,* e successivamente elaborati. Il termine, approdato in Italia con la prima informatica degli anni sessanta indicava al contempo i *dati di entrata* e i *supporti* che li contenevano.

**Periferica:** una qualsiasi tipologia di dispositivo **hardware** del computer che si interfaccia in **input** e/o **output** con l'unità di elaborazione che sovrintende a tutte le funzioni del computer (**processore)**.

**Output:** tradotto letteralmente in “messo fuori”. Indica in senso stretto il risultato di una elaborazione ed in senso più ampio il *risultato* o l'insieme dei *risultati prodotti* a partire da un **input**.

**IDE:** acronimo di “Integrated Developement Enviroment”, tradotto in “ambiente di sviluppo integrato”. **Software** che, durante la **programmazione**, aiuta i programmatori nello sviluppo del **codice sorgente** di un **programma**.

**Debugger: software** specificatamente progettato per l'analisi e l'eliminazione dei bug, ovvero errori di programmazione interni al **codice** di altri **programmi**.

**Codice:** vedi **codice sorgente.**

**Pagina web:** il modo in cui vengono rese disponibili all’utente finale le informazioni reperibili su **internet**, tramite un **web browser**. Un insieme di pagine web tra di loro correlate formano un **sito web**. Una pagina web si puo’ suddividere in una parte relativa ai contenuti, una parte di **layout** e una parte dedicata al comportamento a seconda degli **input** dell’**utente.**

**Browser: programma** che consente di usufruire dei servizi di connettività in **Internet**, o di una **rete** di computer, e di navigare sul **Web**.

**Sito (web):** insieme di **pagine web** correlate, ovvero una struttura **ipertestuale** di documenti che risiede, tramite **hosting**, su un **web server** e accessibile all'**utente client** che ne fa richiesta tramite un **web browser** sul **World Wide Web** della rete **Internet**, digitando in esso il rispettivo **URL** o direttamente l'**indirizzo IP**.

**Ipertesto:** insieme di documenti messi in relazione tra loro per mezzo di parole chiave.

**Hosting:** tradotto letteralmente in “ospitare”. **Servizio** di Rete che consiste nell'allocare su un **server web** le **pagine web** di un **sito web**, rendendolo così accessibile dalla rete **Internet** e ai suoi **utenti**.

**Internet:** (contrazione della locuzione inglese *interconnected networks*, ovvero "reti interconnesse") è una **rete** mondiale di **reti** di computer ad accesso pubblico, attualmente rappresentante il principale mezzo di comunicazione di massa, che offre all'utente una vasta serie di contenuti potenzialmente informativi e **servizi**.

**Rete:** serie di componenti, sistemi o entità interconnessi tra di loro.

**URL:** acronimo di “Uniform Resource Locator”. Sequenza di caratteri che identifica univocamente l'indirizzo di una risorsa in **Internet**, tipicamente presente su un **host server**, come ad esempio un documento, un'immagine, un video, rendendola accessibile ad un **client** che ne faccia richiesta attraverso l'utilizzo di un **web browser**.

**IP:** acronimo e contrazione di “Internet Protocol Address”. Etichetta numerica che identifica univocamente un dispositivo (**host**) collegato a una **rete** informatica che utilizza l**'Internet Protocol** come **protocollo di comunicazione**.

**Protocollo di comunicazione:** insieme di regole formalmente descritte, definite al fine di favorire la comunicazione tra una o più entità.

**Internet Protocol: protocollo di comunicazione** di **rete** appartenente alla suite di protocolli **Internet** TCP/IP su cui è basato il funzionamento della **rete** **Internet**.

**Layout:** identifica l’impaginazione e la struttura grafica di una pagina web o di un documento.

**MongoDB:** sistema gestionale di **basi di dati**, non **relazionale**, **orientato ai documenti** e di tipo **NoSQL**. Il **linguaggio di programmazione** utilizzato per la gestione dei dati e’ **JavaScript**, in particolare la sua notazione **JSON**.

**Orientamento ai documenti:** le **basi di dati** orientate ai documenti non memorizzano i dati in tabelle con campi uniformi per ogni **record** come invece succedeva nei **database** **relazionali**, ma ogni **record** e’ memorizzato come un documento che possiede determinate caratteristiche. Al documento puo’ essere aggiunto un numero qualsiasi di campi ed essi posso anche contenere pezzi multipli di dati.

**Orientamento agli oggetti: paradigma di programmazione** che permette di definire **oggetti software** in grado di interagire gli uni con gli altri attraverso lo scambio di messaggi.

**Orientamento funzionale: paradigma di programmazione** in cui il flusso di esecuzione del **programma** assume la forma di una serie di valutazioni di funzioni matematiche.

**Paradigma di programmazione:** stile fondamentale di programmazione, ovvero un insieme di strumenti concettuali forniti da un **linguaggio di programmazione** per la stesura del **codice sorgente** di un **programma**, definendo dunque il modo in cui il programmatore concepisce e percepisce il **programma** stesso.

**Oggetto software:** definito come un tipo di dato astratto, un oggetto e’ un insieme di valori ed operazioni che permettono di manipolare tali valori, dette operazioni proprie del tipo o metodi del tipo.

**Record:** un oggetto di una **base di dati** strutturata in dati compositi, che contengono un insieme di campi o elementi, ciascuno dei quali possiede nome e tipo di dato propri.

**Base di dati:** detta anche *banca dati,* indica un archivio di dati, o un insieme di archivi, in cui le informazioni in esso contenute sono strutturate e collegate tra loro secondo un particolare modello logico e in modo tale da consentire la gestione/organizzazione efficiente dei dati stessi e l'interfacciamento con le richieste dell'**utente** attraverso le cosiddette **query** (di ricerca o interrogazione, inserimento, cancellazione, aggiornamento ecc.) grazie a particolari applicazioni **software** dedicate, basate su un'architettura di tipo **client-server**.

**Database:** vedi **basi di dati.**

**Relazionale:** modello logico di rappresentazione o strutturazione dei dati di un **database**. Si basa sulla teoria degli insiemi e sulla logica del primo ordine ed è strutturato intorno al concetto matematico di relazione.

**NoSQL:** acronimo di “Not Only SQL”, “non soltanto SQL”. E’ un particolare tipo di **database** che fornisce un sistema di immagazzinamento dei dati e di un loro successivo recupero che richiede l’utilizzo di modelli meno vincolati e restrittivi rispetto ai **database** di tipo **relazionale**. Questo approccio permette una maggiore semplicita’ del processo di modellazione dei dati, una migliore propensione allo **scaling orizzontale** del **database** e un controllo maggiore della disponibilita’ dei dati.

**Scaling orizzontale**: per scalabilita’ si intende la proprieta’ di un sistema di crescere o decrescere in base alle necessita’. Nel mondo dei **database** scalare orizzontalmente una **base di dati** significa che anche se vado ad aumentare le componenti del sistema che accedono al database, questo non provoca interferenze, ed aumentatando le componenti si parallelizza il carico di lavoro, diminuendo il peso del singolo.

**JavaScript: liguaggio di scripting** **orientato agli oggetti** comunente usato nella creazione di **siti web**. In **applicazioni di tipo client-server** se il codice JavaScript e’ sul lato **client** allora viene eseguito sul **client** e non sul **server**, cosi da non sovraccaricare la parte **server.**

**Linguaggio di scripting:** **linguaggio interpretato,** destinato in genere a compiti di automazione del **sistema operativo**, oppure viene usato all’interno delle **pagine web** per gestire il comportamento delle **pagine web** stesse in base all’**input** dell’**utente**.

**Linguaggio interpretato:** Un linguaggio informatico è per definizione diverso dal **linguaggio macchina**. Bisogna quindi tradurlo per renderlo leggibile dal punto di vista del **processore**. Un **programma** scritto in un linguaggio interpretato ha bisogno di un **programma** ausiliario (l'**interprete**) per tradurre man mano le istruzioni del **programma** in **linguaggio macchina.**

**Linguaggio macchina:** definito come il linguaggio utilizzato dal **processore,** tramite sequenze di 0 e 1. Ogni sequenza ordinata di 0 e 1, raggruppata in gruppi di una certa dimensione, identifica una precisa istruzione per il **processore**.

**Processore:** (o unità di elaborazione). Esso è un tipo di dispositivo **hardware** del computer che si contraddistingue per essere dedicato all'[esecuzione](http://it.wikipedia.org/wiki/Esecuzione_(informatica)) di istruzioni. In altri termini l'unità di elaborazione è il dispositivo che nel computer esegue materialmente l'elaborazione dati.

**Interprete:** **programma** in grado di eseguire altri **programmi** a partire direttamente dal relativo **codice sorgente**. Un interprete ha lo scopo di eseguire un **programma** in un **linguaggio di alto livello**, senza la previa **compilazione** dello stesso (**codice oggetto**) cioè di eseguire le [istruzioni](http://it.wikipedia.org/wiki/Istruzione_(informatica)) nel linguaggio usato, traducendole di volta in volta in istruzioni in **linguaggio macchina**.

**Codice sorgente:** linee di codice che compongono un **programma** scritto in un **linguaggio di programmazione**. Abbreviato anche come “sorgente” o solamente “codice”.

**Linguaggio di alto livello: linguaggio di programmazione** più astratto del **linguaggio macchina**, direttamente eseguibile da un computer, ma più vicino o familiare alla logica del nostro linguaggio naturale. I programmi ad alto livello possono essere ricondotti a **programmi** in **linguaggio macchina** in modo automatico, ovvero da un altro **programma**, detto **interprete**.

**Compilazione:** processo di traduzione che porta alla creazione di un **codice oggetto** partendo da un **codice sorgente**.

**Codice oggetto:** traduzione del **codice sorgente** in **linguaggio macchina**, comprensibile dal **processore**.

**Programma:** insieme di istruzioni che, una volta eseguite su un computer, produce (un applicativo volto a fornire un **servizio** o a risolvere un problema.) soluzioni per una data classe di problemi automatizzati.

**Linguaggio di programmazione:** e’ un **linguaggio formale**, ben definito e composto da una **sintassi** e una **semantica.**

**Linguaggio formale:** notazione o formalismo con **sintassi** e **semantica** definite in modo preciso (spesso matematico/formale) e, in molti casi, tali da consentire qualche forma di elaborazione automatica del linguaggio stesso.

**Sistema operativo:** insieme di componenti **software** che consente l’utilizzo di varie apparecchiature informatiche da parte di un **utente.**

**JSON:** acronimo di “JavaScript Object Notation”, tradotto in “notazione **oggetto** **JavaScript**”. E’ un formato di memorizzazione di dati, di piccolo peso e intuitivo.

**Applicazioni client-server:** applicazioni nelle quali esistono due componenti: un **client** che fa una richiesta al **server** e un **server** appunto che deve rispondere alla richiesta del **client**.

**User:** vedi **utente.**

**Utente:** coloro che possono usufruire di un **servizio** che gli viene messo a disposizione.

**Criptato:** questo termine viene utilizzato per riferirsi a quella metodologia che fa si che l’invio dei dati sia reso non comprensibile ad eventuali intercettazioni.

**Interfaccia:** dispositivo fisico o virtuale che permette la comunicazione e l’interazione tra due entita’. Ad esempio in un’interfaccia grafica di un **programma** posso inserire dei valori e visualizzare un eventuale risultato o una risposta da parte del **programma** stesso.

**Mongoose:** strumento di modellazione di **oggetti** in **MongoDB**, scritto in **JavaScript**.

**Infrastruttura:** insieme di risorse **hardware** e risorse **software.**

**Express: applicazione web** scritta in **node.js** minima e flessibile, che fornisce un robusto set di funzionalita’ per la costruzione di singole e multipagine e **applicazioni web** ibride.

**Server:** componente o sottosistema informatico di elaborazione che fornisce un qualunque tipo di **servizio** ad altre componenti (tipicamente chiamate **client**) che ne fanno una esplicita richiesta.

**Node.js:** e’ un **Framework** basato sul **linguaggio** **JavaScript**. Pone la sua attenzione alla manipolazione di grosse quantita’ di dati, quali, per esempio, la consultazione di **database**.

**/\* Linguaggio astratto: ?????? \*/**

**DSL:** acronimo di “Domain Specific Language”, “linguaggio specifico di dominio”. E’ un **linguaggio di programmazione** altamente contestualizzato, cioe’ assiociato ad un dominio specifico.

**Sintassi:** notazione semplice con vincoli.

**Semantica:** regole per la **sintassi**.

**Editor: programma** di composizione di testi, il suo scopo e’ facilitare la scrittura di un testo. E’ generalmente incluso in ogni **sistema operativo**.

**Deployment:** traducibile letteralmente in “spiegamento”. E’ l’insieme di attività necessarie a rendere un sistema **software** disponibile all’uso.

**Heroku:** piattaforma **cloud** per per fornire servizi tramite il **deployment** di applicazioni scritte in linguaggi quali **Node.js, Ruby, Scala**, e molti altri.

**Cloud:** insieme di tecnologie che permettono di archiviare e/o elaborare dati mediante l’utilizzo di risorse **hardware** e **software distribuite**.

**Ruby:** **linguaggio di scripting** completamente a **oggetti**.

**Scala: linguaggio di programmazione** che integra **linguaggi orientati agli oggetti** e **linguaggi funzionali**. Un **programma** Scala, una volta **compilato**, può essere eseguito su **JVM**.

**JVM:** acronimo di Java Virtual Machine, tradotto in “**macchina virtuale** Java”. E’ il componente della **piattaforma Java** che esegue i **programmi** tradotti in **bytecode** dopo una prima **compilazione**.

**Macchina virtuale:** implementazione **software** di un ambiente di elaborazione in cui un **sistema operativo** o un **programma** possono essere installati ed eseguiti.

**Sistema operativo:** insieme di componenti **software** che permettono l’utilizzo da parte di un **utente** di applicazioni installate su una data macchina.

**Bytecode:** codice di **programmazione** che, una volta **compilato**, è eseguibile attraverso una **macchina virtuale** invece che dal **processore** di un computer.

**Piattaforma Java**: è una piattaforma **software** sviluppata su specifiche e implementazioni da **Sun Microsystems** che e’ eseguibile su piattaforme **hardware** di diversa natura.

**Sun Microsystems:** azienda produttrice di **software**, nota per aver prodotto il **linguaggio** **di** **programmazione** **Java**.

**Git:** sistema di controllo di **versione** **distribuito** gratuito e **open source** designato alla gestione di progetti **software.**

**GitHub:** servizio di **web hosting** orientato allo sviluppo software e basato sul sistema di controllo di **versione** di **Git.** GitHub offre servizi di **repository** online sia gratuiti che a pagamento.

**Software**: l’informazione o le informazioni utilizzate da uno o piu’ sistemi informatici e memorizzare su uno o piu’ supporti informatici. Queste informazioni possono essere **programmi**, dati oppure una combinazione di tutte e due.

**Distribuzione:** collezione di **programmi** relativi ad uno o piu’ campi di applicazione, selezionati e rilasciati come un unico pacchetto. Ad esempio il **sistema operativo** **Linux** offre piu’ diverse distribuzioni, come ad esempio **Ubuntu**, **Debian**, **Fedora**, eccetera.

**Hardware:** componenti fisiche che compongono un computer.

**Linux**: sistema operativo della **Linux Foundation**.

**Linux Foundation:** associazione senza fini di lucro, specializzata nel campo dell’informatica **open source.**

**Open source: software** i cui autori ne permettono e favoriscono il libero studio e l'apporto di modifiche da parte di altri programmatori indipendenti.

**Ubuntu**: distribuzione del **sistema operativo Linux**.

**Distribuito (software):** insieme di entità autonome (componenti **software** e **hardware**) spazialmente separate che comunicano e coordinano tra loro le loro azioni attraverso scambio di messaggi.

**Sistema di autenticazione:** e’ un sistema che gestisce un processo tramite il quale un computer, un **software** o un **utente**, verifica la corretta, o presunta, identita’ di un altro computer, **utente**, che vuole comunicare attraverso una connessione.

**Username:** tradotto in “nome **utente**”. E’ il nome fornito da un **utente** di un **servizio** informatico (solitamente **web**) per identificarsi e accedere cosi’ al dato **servizio**.

**Password:** parola di riconoscimento impiegata a scopo di sicurezza per garantire che l’uso di una risorsa sia concesso solo agli **utenti** autorizzati; costituita da una sequenza ordinata di caratteri **alfanumerici** e/o speciali (quali, per esempio, @, $ e %).

**Alfanumerico:** carattere singolo composto solamente da lettere latine maiuscole o minuscole dalla a alla z e dai numeri dallo 0 al 9.

**Email: servizio** **internet** grazie al quale ogni **utente** abilitato puo’ inviare e ricevere dei messaggi utilizzando un computer o qualsiasi altro dispositivo elettronico connesso a **Internet**, attraverso un proprio **account** di posta registrato presso un **provider** del **servizio**.

**Account:** l’insieme di funzionalita’, strumenti e contenuti messi a disposizione da un sito web o da qualsiasi altro tipo di applicazione ad un utente, per usufruire di determinati servizi che il sistema offre.

**Link:** abbreviazione di “hyperlink”. E’ un **collegamento ipertestuale** in grado di rinviare a un contenuto informativo presente in un dominio fisicamente o virtualmente separato. Il link e’ di solito associato ad una o piu’ parole chiave, evidenziate visivamente da una diversa colorazione o sottolineatura.

**Collegamento ipertestuale:** rinvio da un unita’ informativa ad un altra, ha lo scopo di condurre ad altre unita’ informative, ad esempio immagini, altre pagine, ecc.

**Google Chrome: browser** **web** gratuito creato da **Google.**

**Mozilla Firefox: browser web** gratuito creato dalla **Mozilla Foundation.**

**Plug in:** e’ un programma non autonomo che interagisce con un altro programma per ampliarne le funzioni.

**Collection-index:** pagina generata dal framework MaaP che mostra un elenco di Document con delle specifiche coppie chiavi-valori; inoltre e’ presente un menu’ dove e’ possibile spostarsi tra le varie pagine Collection Index o sfruttare altre funzionalita’ messe a disposizione dallo sviluppatore.

**Document-show:** pagina che, in seguito ad una selezione di una chiave di un Document nella pagina Collection index, mostra il Document per intero con tutte le chiavi e i relativi valori.

**Collection:** raccolta di Document in database come MongoDB

**/\*Entry: \*/**

**Sviluppatore (software):** e’ un programmatore che si prende cura di uno o piu’ aspetti del ciclo di vita del **software**. Questa figura può contribuire alla visione d'insieme del progetto ad un livello applicativo piuttosto che a livello di componenti o operazioni individuali di **programmazione** (la codifica).

**Database administration:** e’ la funzione di gestione e manutenzione dei sistemi di gestione dei database; molte aziende importanti necessitano continuamente di una gestione delle **basi di dati**.

**Servizio web:** risorsa astratta che rappresenta la capacita’ di effettuare compiti che compongono una funzionalita’ coerente dal punto di vista delle **entita’** **di** **provider** e dei richiedenti del servizio. Per essere utilizzato, un servizio deve essere realizzato da un **agente di provider** concreto.

**Provider:** la persona o l’organizzazione che fornisce un **servizio web.** “Provider”, letteralmente tradotto, significa “fornitore”.

**Agente di provider:** un **agente** che che ha la capacità e l’autorizzazione di eseguire azioni associate con un **servizio (web)** dedicato al suo possessore, ovvero l’**entità di provider.**

**Entità di provider:** vedi **provider.**

**Agente: programma** che agisce per conto di una persona o un’organzzazione.

**Web:** abbreviazione di “World Wide Web”. Nel linguaggio comune e’ associato ad “**Internet**”, ovvero e’ una **rete** di computer ad accesso pubblico, e’ il maggior sistema di comunicazione di massa che offre anche ulteriori servizi, come ad esempio condividere e cercare risorse.

**Query:** l’interrogazione da parte di un **utente** di un **database** per compiere determinate operazioni sui dati.

**Business logic:** si riferisce a tutta quella logica applicativa che rende operativa un’applicazione. E’ un termine largamente utilizzato nella progettazione del **software** per individuare un componente **software** di una architettura **software**. Nelle **applicazioni web** viene eseguita da un **server** su richiesta di un **client** attraverso il **browser web** e interfacciandosi con la parte dati che può essere un **database**.

**Google:** motore di ricerca per internet che oltre alla funzione di effettuare ricerche offre molti altri servizi, per esempio gestione mail, gestione calendari ecc.. (DA METTERE A POSTO)

**Mozilla Foundation:** organizzazione no profit fondata per supportare e fornire l'organizzazione e la direzione del progetto **open source Mozilla**. L’organizzazione coordina il rilascio delle versioni del **software Mozilla**, stabilisce le linee guida alla base del processo di sviluppo e mantiene l’infrastruttura base necessaria.

**Gmail:** Servizio gratuito di posta elettronica gestito e offerto da **Google.**

**Google Drive: servizio web** offerto e gestito da **Google** che permette l’archiviazione, la sincronizzazione, la condivisione e la modifica collaborativa di documenti.

**TexMaker**: **editor** gratutito multipiattaforma per la scrittura di documenti in **Latex**.

**Repository**: ambiente che offre la possibilità di salvataggio di dati per la sicurezza di essi, inoltre è possibile godere di funzionalità di **versionamento** per tener traccia della storia dei dati salvati. Nel nostro caso l’ambiente utilizzato è **Git**.

**Requisito**: capacità che un sistema **software** deve soddisfare per rispettare un contratto.

**LaTeX**: linguaggio usato per la preparazione di testi basato sul **programma TeX.**

**TeX: programma** di tipografia digitale adatto alla stesura di testi matematici e scientifici.

**Gantt**: l’inventore dei **diagrammi di Gantt**, che vengono usati nella gestione e nella pianificazione delle attività.

**Client:** componente che accede ai **servizi** o alle risorse di un'altra componente detta **server**.

**Struttura dati:** entità usata per organizzare un insieme di dati all'interno della memoria del computer.

**Programmazione:** attività di sviluppo di **software**, consistente nella stesura di **codice sorgente**.

**Host:** ogni terminale collegato ad una **rete** o più in particolare ad **Internet**.

**Applicazione web:** applicazione fruibile via **web**, solitamente con architettura **client-server.**

**Java: linguaggio di programmazione orientato agli oggetti**, creato dalla **Sun Microsystems**.

**Versione:** identificativo univoco che rappresenta la variante di un documento o di un componente **software**.

**Versionamento**: gestione di **versioni** multiple di un insieme di informazioni.

**Diagramma di Gantt:** strumento di supporto alla gestione dei progetti.

**ISO:** acronimo di “International Organization for Standardization”, tradotto in “Organizzazione Internazionale per la Normazione”. Come dice il nome, e’ un’organizzazione internazionale atta alla realizzazione di specifiche standard per la realizzazione di prodotti, servizi e pratiche corrette per aiutare ogni impresa a lavorare in modo efficace ed efficiente.

**IEC:** acronimo di “International Electrotechnical Commission”, tradotto in “Commissione Elettrotecnica Internazionale”. E’ un’organizzazione internazionale per la definizione di standard in materia di elettricita’, elettronica e tecnologie correlate. Molti dei suoi standard sono definiti in collaborazione con l'**ISO** (Organizzazione Internazionale per la Normazione).

Le pubblicazioni dell’IEC servono come base per la normazione internazionale e come riferimento nella stesura di offerte e contratti internazionali.

**Best practice:** tradotto un “bouna prassi”,in genere identifica le esperienze più significative, o comunque quelle che hanno permesso di ottenere migliori risultati, relativamente a svariati contesti.

**Upper bound:** tradotto letteralmente con “limite superiore”. In informatica, si definisce upper bound il limite superiore di memoria occupata di una struttura dati in genere, di solito di un array.

**Array:** detto anche “vettore”, è una struttura dati complessa, statica e omogenea, usata in molti linguaggi di programmazione. Si può immaginare un array come una sorta di casellario, le cui caselle sono dette **celle** dell'array stesso. Ciascuna delle celle si comporta come una variabile tradizionale che rappresenta un *elemento* dell'array; tutte le celle sono variabili di uno stesso tipo preesistente, detto **tipo base** dell'array.

**Tipo (informatico):** nome che indica l'insieme di valori che una variabile, o il risultato di un'espressione, possono assumere e le operazioni che su tali valori si possono effettuare.

**Variabile (informatica):** insieme di dati modificabili situati in una porzione di memoria (una o più locazioni di memoria) destinata a contenere dei dati. Una variabile è caratterizzata da un nome (inteso solitamente come una sequenza di caratteri e cifre).

**Overflow:** eccessiva allocazione di memoria.

**Statement:** traducibile in “sequenza di istruzioni”.