



MaaP: MongoDB as an admin Platform

Glossario

Versione	1.2.0
Data creazione	2013-11-28
Data ultima modifica	2013-12-18
Stato del documento	Formale
Uso del documento	Esterno
Redazione	Alberto Garbui, Fabio Miotto, Mattia Sorgato, Michele Maso, Andrea Perin, Alessandro Benetti
Verifica	Giacomo Pinato
Approvazione	Alessandro Benetti
Distribuzione	Prof. Tullio Vardanega Prof. Riccardo Cardin CoffeeStrap

Sommario

Questo documento si prefigge di chiarire le possibili ambiguità tra i vari termini utilizzati all'interno dei documenti redatti dal gruppo Aperture Software

Diario delle modifiche

Versione	Data	Autore	Modifiche effettuate
1.2.0	2013-12-18	Alessandro Benetti	Approvazione finale del documento
1.1.0	2013-12-17	Giacomo Pinato	Verifica del documento
1.0.8	2013-12-16	Alessandro Benetti	Inserimento termini
1.0.7	2013-12-13	Fabio Miotto	Inserimento termini
1.0.6	2013-12-12	Michele Maso	Inserimento termini
1.0.5	2013-12-09	Mattia Sorgato	Inserimento termini
1.0.4	2013-12-06	Mattia Sorgato	Inserimento termini
1.0.3	2013-12-06	Andrea Perin	Inserimento termini
1.0.2	2013-12-02	Fabio Miotto	Inserimento termini
1.0.1	2013-11-28	Alberto Garbui	Inserimento termini
1.0.0	2013-11-28	Fabio Miotto	Creazione struttura iniziale del documento

Tabella 1: Registro delle modifiche

Indice

1	Introduzione	7
1.1	Scopo del documento	7
1.2	Riferimenti	7
1.2.1	Normativi	7
1.2.2	Informativi	7
2	Termini	8
A		9
	Account	9
	Agente di provider	9
	Alfanumerico	9
	Ambiente	9
	Ambiguo	9
	Amministratore	9
	Applicazioni client-server	9
	Applicazione web	9
	Array	9
	Attributo	10
B		11
	Base di dati	11
	Baseline	11
	Best practice	11
	Branch	11
	Browser	11
	Business	11
	Business logic	11
	Bytecode	11
C		12
	Chat	12
	Chiave	12
	Classe	12
	Client	12
	Cloud	12
	Codice	12
	Codice oggetto	12
	Codice sorgente	12
	Collection	12
	Collection-Index	12
	Compatibilità	13
	Compilazione	13
	Criptato	13

D	14
Dashboard	14
Database	14
Database administration	14
Debugger	14
Default	14
Deployment	14
Diagramma di Gantt	14
Distribuito	14
Distribuzione	14
Document	15
Document-Show	15
Driver	15
DSL	15
E	16
Editor	16
Email	16
Express	16
F	17
File di configurazione	17
File di descrizione	17
File system	17
Filtro	17
Firefox	17
Form	17
Framework	17
Funzione-software	17
G	19
Gantt	19
Git	19
GitHub	19
Gmail	19
Google	19
Google Chrome	19
Google Drive	19
Google Inc.	19
H	20
Hardware	20
Heroku	20
Host	20
Hosting	20

I	21
IDE	21
IEC	21
Infrastruttura	21
Input	21
Interfaccia	21
Internet	21
Internet Protocol	21
Interprete	21
IP	22
Ipertesto	22
ISO	22
Istanza	22
J	23
Java	23
JavaScript	23
JSON	23
JVM	23
L	24
LaTeX	24
Layout	24
Libreria di codice	24
Linguaggio di alto livello	24
Linguaggio di programmazione	24
Linguaggio di scripting	24
Linguaggio formale	24
Link	24
Linux	24
Linux Foundation	25
Linguaggio interpretato	25
Linguaggio macchina	25
Logger	25
M	26
MaaP	26
Macchina virtuale	26
MaaP's web	26
Metodo	26
Milestone	26
MongoDB	26
N	27
Node.js	27
NoSQL	27

O

Oggetto-software	28
Open source	28
Orientamento agli oggetti	28
Orientamento ai documenti	28
Orientamento funzionale	28
Output	28
Overflow	28

P

Package	29
Pagina web	29
Paradigma di programmazione	29
Password	29
Periferica	29
Permesso	29
Piattaforma Java	29
Plug in	29
Processore	30
Profilo	30
Programma	30
Programmatore	30
Programmazione	30
Protocollo di comunicazione	30
Provider	30

Q

Query	31
-----------------	----

R**32**

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 riportato sotto il nome una breve definizione che ne chiarisce il significato.

1.2 Riferimenti

1.2.1 Normativi

- Norme di Progetto: *Norme_di_Progetto_v1.2.0.pdf* (allegato alla presente documentazione)

1.2.2 Informativi

- Wikipedia:
<http://www.wikipedia.org/>
- OkPedia:
<http://www.okpedia.it/>
- W3C:
<http://www.w3.org/>
- Techterms:
<http://www.techterms.com/>
- ISO:
<http://www.iso.org/iso/home/about.htm>
- Francesco Ranzato, Appunti di programmazione oggetti, Edizione II.

2 Termini

Di seguito saranno elencati e definiti tutti i vari termini in ordine alfabetico.

A**Account**

L'insieme di funzionalità, 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.

Agente di provider

Un agente che ha la capacità e l'autorizzazione di eseguire azioni associate con un *servizio web_G* dedicato al suo possessore, ovvero l'entità di *provider_G*.

Alfanumerico

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

Ambiente

Consiste sia del sistema *hardware_G* che di quello *software_G* sui quali è stato pianificato l'utilizzo del prodotto software sviluppato.

Ambiguo

Termine usato per indicare una cosa che non è chiara, ovvero presenta un doppio senso o un fraintendimento.

Amministratore

Classe di utenza privilegiata rispetto ad *utente_G* normale, ovvero ha funzionalità in più che gestiscono un sistema.

Applicazioni client-server

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

Applicazione web

Applicazione fruibile via *web_G*, solitamente con architettura *client-server_G*.

Array

Detto anche vettore, è una struttura dati complessa, statica e omogenea, usata in molti *linguaggi di programmazione_G*. 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_G* tradizionale che rappresenta un elemento dell'array; tutte le celle sono variabili di uno stesso tipo preesistente, detto *tipo base_G* dell'array.

Attributo

Descrizione di un campo dati di una *classe_G*.

B**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_G* attraverso le cosiddette *query_G* (di ricerca o interrogazione, inserimento, cancellazione, aggiornamento ecc.) grazie a particolari applicazioni *software_G* dedicate, basate su un'architettura di tipo *client-server_G*.

Baseline

Durante la pianificazione rappresenta la suddivisione iniziale delle attività nel tempo a cui si fa riferimento per l'avanzamento del processo.

Best practice

Tradotto in buona prassi, in genere identifica le esperienze più significative, o comunque quelle che hanno permesso di ottenere migliori risultati, relativamente a svariati contesti.

Branch

Ramo di sviluppo parallelo del filone principale.

Browser

Programma_G che consente di usufruire dei *servizi_G* di connettività in *Internet_G*, o di una *rete_G* di computer, e di navigare sul *Web_G*.

Business

Letteralmente tradotto in affari; identifica un'attività economica.

Business logic

Si riferisce a tutta quella logica applicativa che rende operativa un'applicazione. È un termine largamente utilizzato nella progettazione del *software_G* per individuare un componente software di una architettura software. Nelle *applicazioni web_G* viene eseguita da un *server_G* su richiesta di un *client_G* attraverso il *browser_G* e interfacciandosi con la parte dati che può essere un *database_G*.

Bytecode

Codice_G di *programmazione_G* che, una volta effettuata la *compilazione_G*, è eseguibile attraverso una *macchina virtuale_G* invece che dal *processore_G* di un computer.

C**Chat**

Sistema di comunicazione in tempo reale che permette a più *utenti_G* di scambiarsi brevi messaggi scritti, emulando una conversazione o chiacchierata.

Chiave

Identificatore di un elemento del *database_G* di tipo *NoSQL_G*.

Classe

Nella *programmazione_G orientata agli oggetti_G* una classe è un costrutto di un *linguaggio di programmazione_G* usato come modello per creare *oggetti_G*. Il modello comprende *attributi_G* e *metodi_G* che saranno condivisi da tutti gli oggetti creati (*istanze_G*) a partire dalla classe. Un oggetto è, di fatto, l'istanza di una classe.

Client

Componente che accede ai *servizi_G* o alle risorse di un'altra componente detta *server_G*.

Cloud

Insieme di tecnologie che permettono di archiviare e/o elaborare dati mediante l'utilizzo di risorse *hardware_G* e *software_G distribuite_G*.

Codice

Vedi codice sorgente.

Codice oggetto

Traduzione del *codice sorgente_G* in *linguaggio macchina_G*, comprensibile dal *processore_G*.

Codice sorgente

Linee di codice che compongono un *programma_G* scritto in un *linguaggio di programmazione_G*. Abbreviato anche come sorgente o solamente codice.

Collection

Raccolta di *Document_G* in *database_G* come *MongoDB_G*.

Collection-Index

Pagina generata dal *framework_G MaaP_G* che mostra un elenco di *Document_G* con delle specifiche coppie chiavi-valori; inoltre è presente un menù dove è possibile spostarsi tra le varie pagine *Collection Index_G* o sfruttare altre funzionalità messe a disposizione dallo *sviluppatore_G*.

Compatibilità

In informatica indica la proprietà di due o più computer che, scambiandosi dei dati, accettano gli stessi *programmi_G* senza fare modifiche ad essi.

Compilazione

Processo di traduzione che porta alla creazione di un *codice oggetto_G* partendo da un *codice sorgente_G*.

Criptato

Questo termine viene utilizzato per riferirsi a quella metodologia che fa sì che l'invio dei dati sia reso non comprensibile ad eventuali intercettazioni.

D**Dashboard**

Pagina principale dove l'*utente_G* può aver accesso alle varie funzionalità in modo chiaro e diretto.

Database

Vedi *basi di dati_G*.

Database administration

Funzione di gestione e manutenzione dei sistemi di gestione dei *database_G*; molte aziende importanti necessitano continuamente di una gestione delle *basi di dati_G*.

Debugger

Software_G specificatamente progettato per l'analisi e l'eliminazione dei bug, ovvero errori di programmazione interni al *codice_G* di altri *programmi_G*.

Default

Valore o un'azione standard che caratterizza qualsiasi aspetto globale a meno di un cambiamento improvviso.

Deployment

Traducibile letteralmente in spiegamento. È l'insieme di attività necessarie a rendere un sistema *software_G* disponibile all'uso.

Diagramma di Gantt

Strumento di supporto alla gestione dei progetti. È costruito partendo da un asse orizzontale, a rappresentazione dell'arco temporale totale del progetto, suddiviso in fasi incrementali (ad esempio, giorni, settimane, mesi), e da un asse verticale, a rappresentazione delle mansioni o attività che costituiscono il progetto. Un diagramma di *Gantt_G* permette dunque la rappresentazione grafica di un calendario di attività, utile al fine di pianificare, coordinare e tracciare specifiche attività in un progetto dando una chiara illustrazione dello stato d'avanzamento del progetto rappresentato.

Distribuito

Insieme di entità autonome (componenti *software_G* e *hardware_G*) fisicamente separate che comunicano e coordinano tra loro le azioni attraverso scambio di messaggi.

Distribuzione

Collezione di *programmi_G* relativi ad uno o più campi di applicazione, selezionati e rilasciati come un unico pacchetto. Ad esempio il *sistema operativo_G Linux_G* offre più diverse distribuzioni, come ad esempio Ubuntu, Debian, Fedora, eccetera.

Document

Istanza_G, o *record_G*, di un *database_G* in *MongoDB_G*. Esso è costituito da un insieme di *chiavi_G* con il rispettivo valore.

Document-Show

Pagina che, in seguito ad una selezione di una *chiave_G* di un *Document_G* nella pagina *Collection-Index_G*, mostra il Document per intero con tutte le chiavi e i relativi valori.

Driver

Insieme di procedure che permette ad un *sistema operativo_G* di controllare un dispositivo *hardware_G* o *software_G*.

DSL

Acronimo di Domain Specific Language, linguaggio specifico di dominio. È un *linguaggio di programmazione_G* altamente contestualizzato, cioè associato ad un dominio specifico.

E**Editor**

Programma_G di composizione di testi, il suo scopo è facilitare la scrittura di un testo. È generalmente incluso in ogni *sistema operativo_G*.

Email

Servizio_G internet_G grazie al quale ogni *utente_G* abilitato può inviare e ricevere dei messaggi utilizzando un computer o qualsiasi altro dispositivo elettronico connesso a *Internet_G*, attraverso un proprio *account_G* di posta registrato presso un *provider_G* del servizio.

Express

Applicazione web_G scritta in *Node.js_G* minima e flessibile, che fornisce un robusto set di funzionalità per la costruzione di singole e multi-pagine applicazioni web ibride.

F**File di configurazione**

File che permette di impostare parametri necessari al funzionamento dell'applicazione nella loro versione predefinita; ad esempio *database_G* per sistema di *autenticazione_G*, *e-mail_G*, *server_G*, ecc.

File di descrizione

File scritto con *linguaggio_G DSL_G* dallo *sviluppatore_G*, e serve per generare pagine di tipo *Collection-Index_G* e *Document-Show_G*.

File system

Meccanismo con il quale i file sono posizionati e organizzati su un dispositivo di archiviazione.

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_G*.

Firefox

Browser_G open source_G multiplatforma e secondo alcune statistiche è il secondo browser più popolare.

Form

Termine usato per indicare l'*interfaccia_G* di una applicazione che consente ad un *utente_G* di inserire e inviare ad un *server_G* dei dati liberamente digitati dallo stesso utente.

Framework

Struttura di supporto su cui un *software_G* può essere progettato e realizzato. Alla base di un framework sono sempre presenti delle *librerie di codice_G* utilizzabili con uno o più *linguaggi di programmazione_G*; esse sono spesso corredate da una serie di strumenti di supporto allo sviluppo software, come ad esempio un *IDE_G* o un *debugger_G*, o altri strumenti ideati per aumentare la velocità di sviluppo del prodotto finito. Lo scopo di un framework è quello di far risparmiare allo *sviluppatore_G* la riscrittura di *codice_G* già scritto precedentemente per fini simili. La necessità di questo strumento si è venuta a creare quando le *interfacce_G* utente sono diventate sempre più complesse ed è cominciata ad aumentare la quantità software con funzionalità secondarie simili.

Funzione-software

Particolare costruito con *sintassi_G*, in qualche *linguaggio di programmazione_G*, che permette di raggruppare, all'interno di un *programma_G*, 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 $input_G$ restituisca determinati $output_G$.

G**Gantt**

L'inventore dei *diagrammi di Gantt_G*, che vengono usati nella gestione e nella pianificazione delle attività.

Git

Sistema di controllo di *versione_G* *distribuito_G* gratuito e *open source_G* designato alla gestione di progetti *software_G*

GitHub

Servizio_G di *web_G* *hosting_G* orientato allo sviluppo *software_G* e basato sul sistema di controllo di *versione_G* di *Git_G*. GitHub offre servizi di *repository_G* online sia gratuiti che a pagamento.

Gmail

Servizio_G gratuito di posta elettronica gestito e offerto da *Google Inc._G*.

Google

Motore di ricerca per *Internet_G* che oltre alla funzione di effettuare ricerche offre molti altri *servizi_G*, per esempio gestione *e-mail_G*, gestione calendari ecc..

Google Chrome

Browser_G gratuito creato da *Google Inc._G*.

Google Drive

Servizio web_G offerto e gestito da *Google Inc._G* che permette l'archiviazione, la sincronizzazione, la condivisione e la modifica collaborativa di documenti.

Google Inc.

Azienda statunitense che offre *servizi_G* online, principalmente nota per il motore di ricerca *Google_G* e per servizi come *Google Drive_G* ecc.

H**Hardware**

Componenti fisiche che compongono un computer.

Heroku

Piattaforma *cloud_G* per per fornire *servizi_G* tramite il *deployment_G* di applicazioni scritte in *linguaggi_G* quali *Node.js_G*, *Ruby_G*, *Scala_G*, e molti altri.

Host

Ogni terminale collegato ad una *rete_G* o più in particolare ad *internet_G*.

Hosting

Tradotto letteralmente in ospitare. *Servizio_G* di *rete_G* che consiste nell'allocare su un *server web_G* le *pagine web_G* di un *sito web_G*, rendendolo così accessibile dalla rete *Internet_G* e ai suoi *utenti_G*.

I**IDE**

Acronimo di Integrated Developement Enviroment, tradotto in ambiente di sviluppo integrato. *Software_G* che, durante la *programmazione_G*, aiuta i *programmatori_G* nello sviluppo del *codice sorgente_G* di un *programma_G*.

IEC

Acronimo di International Electrotechnical Commission, tradotto in Commissione Elettrotecnica Internazionale. È un'organizzazione internazionale per la definizione di standard in materia di elettricità, elettronica e tecnologie correlate. Molti dei suoi standard sono definiti in collaborazione con l'*ISO_G*.

Infrastruttura

Insieme di risorse *hardware_G* e risorse *software_G*.

Input

Tradotto letteralmente in immettere. Sequenza di dati o informazioni, immessi per mezzo di una *periferica_G*, detta appunto di *input_G*, e successivamente elaborati.

Interfaccia

Dispositivo fisico o virtuale che permette la comunicazione e l'interazione tra due entità. Ad esempio in un'*interfaccia_G* grafica di un *programma_G* posso inserire dei valori e visualizzare un eventuale risultato o una risposta da parte del programma stesso.

Internet

Contrazione della locuzione inglese interconnected networks, ovvero reti interconnesse. È una *rete_G* mondiale di reti di computer ad accesso pubblico, attualmente rappresentante il principale mezzo di comunicazione di massa, che offre all'*utente_G* una vasta serie di contenuti potenzialmente informativi e *servizi_G*.

Internet Protocol

Protocollo di comunicazione_G di *rete_G* appartenente all'insieme di protocolli *Internet_G* TCP/IP su cui è basato il funzionamento della rete Internet.

Interprete

Programma_G in grado di eseguire altri programmi a partire direttamente dal relativo *codice sorgente_G*. Un interprete ha lo scopo di eseguire un programma in un *linguaggio di alto livello_G*, senza la previa *compilazione_G* dello stesso (*codice oggetto_G*) cioè di eseguire le istruzioni nel *linguaggio_G* usato, traducendole di volta in volta in istruzioni in *linguaggio macchina_G*.

IP

Acronimo di Internet Protocol Address. L'IP è un'etichetta numerica che identifica univocamente un dispositivo ($host_G$) collegato a una $rete_G$ informatica che utilizza l'Internet Protocol come *protocollo di comunicazione_G*.

Ipertesto

Insieme di documenti messi in relazione tra loro per mezzo di parole chiave.

ISO

Acronimo di International Organization for Standardization, tradotto in Organizzazione Internazionale per la Normazione. Come dice il nome, è un'organizzazione internazionale atta alla realizzazione di specifiche standard per la realizzazione di prodotti, $servizi_G$ e pratiche corrette per aiutare ogni impresa a lavorare in modo efficace ed efficiente.

Istanza

Un particolare $oggetto_G$ di una particolare $classe_G$.

J**Java**

Linguaggio di programmazione_G orientato agli oggetti_G, creato dalla Sun Microsystems_G.

JavaScript

Linguaggio di scripting_G orientato agli oggetti_G comunemente usato nella creazione di siti web_G. In applicazioni di tipo client-server_G se il codice JavaScript è sul lato client_G allora viene eseguito sul client e non sul server_G, così da non sovraccaricare la parte server.

JSON

Acronimo di *JavaScript_G Object Notation*, tradotto in notazione oggetto JavaScript. È un formato di memorizzazione di dati, di piccolo peso e intuitivo.

JVM

Acronimo di *Java_G Virtual Machine*, tradotto in *macchina virtuale_G Java*. È il componente della *piattaforma Java_G* che esegue i *programmi_G* tradotti in *bytecode_G* dopo una prima *compilazione_G*.

L**LaTeX**

Linguaggio_G usato per la preparazione di testi basato sul *programma_G TeX_G*

Layout

Identifica l'impaginazione e la struttura grafica di una *pagina web_G* o di un documento.

Libreria di codice

Insieme di funzioni o strutture dati predisposte per essere collegate ad un *programma_G software_G* attraverso opportuno collegamento.

Linguaggio di alto livello

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

Linguaggio di programmazione

Linguaggio formale_G, ben definito e composto da una *sintassi_G* e una *semantica_G*.

Linguaggio di scripting

Linguaggio interpretato_G, destinato in genere a compiti di automazione del *sistema operativo_G*, oppure viene usato all'interno delle *pagine web_G* per gestire il comportamento delle pagine web stesse in base all' *input_G* dell'*utente_G*.

Linguaggio formale

Notazione o formalismo con *sintassi_G* e *semantica_G* definite in modo preciso (spesso matematico/formale) e, in molti casi, tali da consentire qualche forma di elaborazione automatica del *linguaggio_G* stesso.

Link

Abbreviazione di hyperlink. È un collegamento ipertestuale in grado di rinviare a un contenuto informativo presente in un dominio fisicamente o virtualmente separato. Il link è di solito associato ad una o più parole chiave, evidenziate visivamente da una diversa colorazione o sottolineatura.

Linux

Sistema operativo_G della *Linux Foundation_G*.

Linux Foundation

Associazione senza fini di lucro, specializzata nel campo dell'informatica *open source*_G.

Linguaggio interpretato

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

Linguaggio macchina

Definito come il *linguaggio*_G utilizzato dal *processore*_G, 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.

Logger

Componente non intrusivo di registrazione dei dati di esecuzione per analisi dei risultati.

M**MaaP**

Framework_G che genera *interfacce_G web_G* di amministrazione dei dati di *business_G* basati sulle tecnologie *Node.js_G* e *MongoDB_G*.

Macchina virtuale

Implementazione *software_G* di un *ambiente_G* di elaborazione in cui un *sistema operativo_G* o un *programma_G* possono essere installati ed eseguiti.

MaaP's web

Insieme delle *pagine web_G* prodotte dal *framework_G MaaP_G*.

Metodo

Detto anche funzione-software membro, è un termine che viene usato principalmente nel contesto della *programmazione_G orientata agli oggetti_G* per indicare un sottoprogramma associato in modo esclusivo ad una *classe_G* e che rappresenta (in genere) un'operazione eseguibile sugli oggetti-software e *istanze_G* di quella classe.

Milestone

Viene utilizzato nella pianificazione e gestione di progetti per indicare il raggiungimento di obiettivi definiti in fase di definizione del progetto stesso. Molto spesso le milestone sono rappresentate da eventi, come ad esempio scadenze per la consegna di documenti, e indicano importanti traguardi intermedi durante lo svolgimento del progetto.

MongoDB

Sistema gestionale di *basi di dati_G*, non *relazionale_G*, *orientato ai documenti_G* e di tipo *NoSQL_G*. Il *linguaggio di programmazione_G* utilizzato per la gestione dei dati è *JavaScript_G*, in particolare la sua notazione *JSON_G*.

N**Node.js**

Framework_G basato sul *linguaggio_G JavaScript_G*. Pone la sua attenzione alla manipolazione di grosse quantità di dati, quali, per esempio, la consultazione di *database_G*.

NoSQL

Acronimo di Not Only SQL, non soltanto SQL. È un particolare tipo di *database_G* 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_G*. Questo approccio permette una maggiore semplicità del processo di modellazione dei dati, una migliore propensione allo *scaling orizzontale_G* del database e un controllo maggiore della disponibilità dei dati.

O**Oggetto-software**

Definito come un *tipo-informatica_G* di dato astratto, un oggetto è un insieme di valori ed operazioni che permettono di manipolare tali valori, dette operazioni proprie del tipo o metodi del tipo.

Open source

Software_G i cui autori ne permettono e favoriscono il libero studio e l'apporto di modifiche da parte di altri *programmatori_G* indipendenti.

Orientamento agli oggetti

Paradigma di programmazione_G che permette di definire *oggetti-software_G* in grado di interagire gli uni con gli altri attraverso lo scambio di messaggi.

Orientamento ai documenti

Le *basi di dati_G* orientate ai documenti non memorizzano i dati in tabelle con campi uniformi per ogni *record_G* come invece succedeva nei *database_G relazionali_G*, ma ogni record è memorizzato come un documento che possiede determinate caratteristiche. Al documento può essere aggiunto un numero qualsiasi di campi ed essi possono anche contenere pezzi multipli di dati.

Orientamento funzionale

Paradigma di programmazione_G in cui il flusso di esecuzione del *programma_G* assume la forma di una serie di valutazioni di funzioni matematiche.

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_G*.

Overflow

Errore che occorre quando si eccede la memoria allocata disponibile.

P**Package**

Un package, definito principalmente in ambiente *Java_G*, è un meccanismo per organizzare *classi_G* all'interno di sottogruppi ordinati. I *programmatori_G* spesso usano i package per riunire classi logicamente correlate o che forniscono servizi simili.

Pagina web

Il modo in cui vengono rese disponibili all'*utente_G* finale le informazioni reperibili su *Internet_G*, tramite un *web browser_G*. Un insieme di pagine web tra di loro correlate formano un *sito web_G*. Una pagina web si può suddividere in una parte relativa ai contenuti, una parte di *layout_G* e una parte dedicata al comportamento a seconda degli *input_G* dell'utente.

Paradigma di programmazione

Stile fondamentale di *programmazione_G*, ovvero un insieme di strumenti concettuali forniti da un *linguaggio di programmazione_G* per la stesura del *codice sorgente_G* di un *programma_G*, definendo dunque il modo in cui il *programmatore_G* concepisce e percepisce il programma stesso.

Password

Parola di riconoscimento impiegata a scopo di sicurezza per garantire che l'uso di una risorsa sia concesso solo agli *utenti_G* autorizzati; costituita da una sequenza ordinata di caratteri *alfanumerici_G* e/o speciali (quali, per esempio, @, %, \$).

Periferica

Una qualsiasi tipologia di dispositivo *hardware_G* del computer che si interfaccia in *input_G* e/o *output_G* con l'unità di elaborazione che sovrintende a tutte le funzioni del computer (*processore_G*).

Permesso

O autorizzazione, indica quali funzionalità possono competere.

Piattaforma Java

Piattaforma *software_G* sviluppata su specifiche e implementazioni da *Sun Microsystems_G* che è eseguibile su piattaforme *hardware_G* di diversa natura.

Plug in

Programma_G non autonomo che interagisce con un altro programma per ampliarne le funzionalità.

Processore

Detto anche unità di elaborazione. Esso è un tipo di dispositivo *hardware_G* del computer che si contraddistingue per essere dedicato all'esecuzione di istruzioni. In altri termini l'unità di elaborazione è il dispositivo che nel computer esegue materialmente l'elaborazione dati.

Profilo

Insieme di dati relativi ad un *utente_G* in un *sistema_G* informatico. Può contenere informazioni differenti, a seconda del contesto e delle necessità del sistema.

Programma

Insieme di istruzioni che, una volta eseguite su un computer, produce soluzioni per una data classe di problemi automatizzati.

Programmatore

Vedi *sviluppatore_G*.

Programmazione

Attività di sviluppo di *software_G*, consistente nella stesura di *codice sorgente_G*.

Protocollo di comunicazione

Insieme di regole formalmente descritte, definite al fine di favorire la comunicazione tra una o più entità.

Provider

La persona o l'organizzazione che fornisce un *servizio web_G*. Provider, letteralmente tradotto, significa fornitore.

Q

Query

L'interrogazione da parte di un $utente_G$ di un $database_G$ per compiere determinate operazioni sui dati.

R