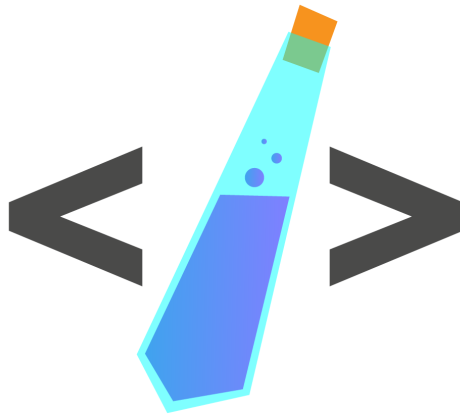




CODE ALCHEMISTS
TURNING IDEAS INTO CODE, LIKE MAGIC

Glossario

Stato

In redazione

Versione

0.5.0

Distribuzione

Code Alchemists
Prof. Tullio Vardanega
Prof. Riccardo Cardin



Registro delle Modifiche

Vers.	Data	Autore	Verificatore	Descrizione
0.5.0	16/07/2025	M. Pretto	N. Bolzon	Modifica di alcune definizioni del glossario, rimozione di alcune che risultavano ambigue, riordino dei vocaboli
0.4.0	22/05/2025	N. Moretto	S. Marana	Aggiunte definizioni al glossario e riordino dei vocaboli
0.3.0	28/04/2025	A. Shu	S. Marana	Aggiunte definizioni al glossario
0.2.1	15/04/2025	N. Bolzon	A. Shu	Riordino dei vocaboli in ordine alfabetico
0.2.0	08/04/2025	N. Bolzon	S. Marana	Aggiunta definizioni al glossario
0.1.0	28/03/2025	N. Bolzon	S. Marana	Aggiunta definizioni al glossario
0.0.1	26/03/2025	N. Moretto	N. Bolzon	Creazione template e struttura del documento

Indice

Introduzione	4
A	5
B	6
C	7
D	9
F	10
G	11
H	12
I	13
L	14
M	15
N	16
O	17
P	18
R	19
S	21
T	22
V	23
W	24

Introduzione

Il presente glossario raccoglie e definisce i termini ambigui o specialistici pertinenti al progetto, con l'obiettivo di garantire una comprensione univoca e condivisa tra tutte le parti coinvolte. Attraverso definizioni chiare e precise, esso favorisce una comunicazione efficace all'interno del team e con il committente, assicurando coerenza terminologica nella documentazione.

La nomenclatura adottata per indicare che un termine è presente nel glossario è la seguente:

parola^G

A

- **AGILE:**

Modello iterativo/incrementale che si basa sulla comunicazione/collaborazione. Lo scopo di questo modello è di suddividere il lavoro a piccoli incrementi a valore aggiunto, realizzati in sequenza continua dall'analisi all'integrazione, dimostrando in modo trasparente al cliente quanto fatto e tracciando l'avanzamento secondo un progresso reale e concreto.

- **Alert:**

Segnalazioni automatiche generate dal sistema in risposta a eventi rilevanti, come esaurimento scorte, errori di trasferimento, o necessità di approvazione.

- **Amministratore:**

Ricopre un ruolo di alto livello e dedica il minor numero di ore, in quanto non è costantemente coinvolto nello sviluppo. È responsabile di garantire l'efficacia delle tecnologie e degli strumenti utilizzati, supportando la collaborazione del team con attività come test automatici e risoluzione di malfunzionamenti. Assicura l'efficienza di procedure, strumenti e tecnologie a supporto del Way of Working, contribuendo alla stabilità del progetto e riducendo i tempi di inattività, migliorando così l'efficienza complessiva.

- **Analisi dei requisiti:**

Processo di esame e interpretazione del capitolato per individuare i bisogni del committente, formalizzare i requisiti funzionali e non funzionali del sistema, e proporre soluzioni che li soddisfino. Il risultato è un documento strutturato che guida lo sviluppo del software, assicurando coerenza tra le aspettative del cliente e il prodotto finale.

- **Analista:**

Svolge un ruolo chiave nelle fasi iniziali, fornendo basi solide a progettisti e programmatori. Il suo impegno orario, inferiore alla media degli altri ruoli, si concentra sull'analisi e la definizione dei requisiti, garantendo chiarezza e completezza delle specifiche progettuali.

- **API:**

Sta per «Application Programming Interface» e significa «interfaccia di programmazione delle applicazioni». È un insieme di regole e protocolli che consentono a diverse applicazioni software di interagire e scambiarsi dati e funzionalità.

- **Approvvigionamento:**

Attività di acquisizione di prodotti o materiali necessari per garantire il funzionamento del magazzino. Include ordini da fornitori esterni o trasferimenti da altri magazzini.

- **Architettura:**

Struttura organizzativa di un sistema, sia esso hardware, software o di rete, che definisce come i suoi componenti interagiscono e cooperano per svolgere le funzioni richieste. Si tratta di una progettazione che include dettagli come l'organizzazione fisica dei componenti, i protocolli di comunicazione e le interazioni tra i diversi elementi.

- **Autenticazione:**

Meccanismo per verificare l'identità di un utente prima di concedere accesso al sistema.

- **Autenticazione a due fattori (2FA):**

Sistema di autenticazione che richiede due forme di verifica (es. password + codice via SMS) per garantire una maggiore sicurezza.

B

- **Backlog:**
Elenco ordinato di attività, funzionalità o modifiche da implementare in un progetto.
- **Backup:**
Copia di sicurezza dei dati, utile per ripristinare le informazioni in caso di guasti o perdite.
- **Baseline:**
La baseline di progetto è un punto di riferimento fisso utilizzato per confrontare le prestazioni dei progetti nel tempo. Le baseline di progetto sono utilizzate dai project manager per verificare l'andamento dell'ambito, della programmazione e dei costi del progetto fino al suo completamento.
- **Branch:**
Versione separata di un progetto o codice, utilizzata per sviluppare nuove funzionalità senza modificare la versione principale.
- **Bugfix:**
Correzione di un bug, ovvero di un errore nel software o nel codice.

C

- **Capitolato:**
Documento formale che descrive in modo dettagliato le esigenze, i requisiti e i vincoli di un progetto software, forniti dal committente. Costituisce la base per l'analisi dei requisiti e guida lo sviluppo del sistema. Viene utilizzato anche come base per la negoziazione e per la gestione delle modifiche durante l'esecuzione del progetto.
- **Certificato Digitale:**
Documento elettronico che attesta l'identità di un'entità online, garantendo comunicazioni sicure.
- **Chief Technology Officer (CTO):**
Dirigente di primo livello responsabile delle scelte tecnologiche di un'azienda. Valuta, seleziona e propone le soluzioni tecniche più adatte per supportare lo sviluppo di prodotti o servizi, garantendo l'allineamento tra innovazione tecnologica e obiettivi strategici dell'organizzazione. Fa parte del consiglio direttivo e collabora con le altre funzioni aziendali per guidare l'evoluzione tecnologica.
- **CI/CD:**
Insieme di pratiche di integrazione continua (Continuous Integration) e distribuzione continua (Continuous Deployment) per automatizzare test e rilascio del software.
- **Cifratura:**
Processo di trasformazione dei dati in un formato illeggibile, al fine di proteggerli da accessi non autorizzati.
- **Client:**
È un componente software o hardware che accede ai servizi o risorse di un «server». In sostanza, il client è il programma o dispositivo che effettua una richiesta al server per ricevere dati, servizi o per eseguire un'azione.
- **Cliente:**
Persona, azienda o ente che commissiona, finanzia e utilizzerà direttamente o indirettamente il prodotto software. Il cliente può coincidere o meno con il committente, e ha interesse nel risultato finale del progetto. È coinvolto nei processi decisionali, nella validazione del prodotto e nella sua eventuale accettazione e manutenzione post-rilascio.
- **Commit:**
Operazione con cui si salvano modifiche al codice sorgente in un sistema di controllo versione.
- **Committente:**
Persona o organizzazione che incarica formalmente un team di sviluppo di realizzare un sistema software. È responsabile della definizione iniziale degli obiettivi, dei requisiti e dei vincoli del progetto, e può partecipare attivamente alle fasi di analisi, verifica e validazione. Il committente rappresenta il punto di riferimento principale per comprendere le esigenze da soddisfare.
- **Connettività:**
Capacità dei sistemi dei magazzini di restare collegati alla rete o al sistema centrale.
- **Container:**
Ambiente isolato e leggero che include tutto il necessario per eseguire un'applicazione (codice,

librerie, dipendenze), utilizzando il kernel del sistema operativo sottostante, garantendo portabilità e coerenza tra diversi sistemi.

D

- **Dashboard:**

Interfaccia utente centralizzata che visualizza in tempo reale informazioni sul sistema.

F

- **Firma Digitale:**

È un metodo sicuro e legale per firmare documenti digitali, garantendo l'autenticità, l'integrità e il non ripudio del firmatario.

G

- **Glossario:**

Documento contenente parole o frasi che non sono di uso comune, ma che sono state scelte per essere più specifiche e per evitare ambiguità.

H

- **Hashing:**

Processo che converte un input di dati di lunghezza variabile in un output di lunghezza fissa, chiamato «hash» o «digest». Questo output è una sequenza di caratteri che può essere utilizzata per rappresentare in modo compatto l'input originale, e che cambia in modo sensibile al cambiamento del contenuto dell'input. L'obiettivo principale dell'hashing è garantire l'integrità dei dati e convalidare i dati originali.

I

- **Incrementale:**

Qualcosa che viene sviluppato o costruito in modo graduale, per piccoli passi, aggiungendo funzionalità o miglioramenti a una versione precedente.

- **Integrità:**

Correttezza e coerenza dei dati nel sistema.

- **Intrusione:**

Accesso non autorizzato a un sistema informatico.

- **Inventario:**

Elenco dettagliato e aggiornato delle scorte disponibili in un magazzino.

- **Iterativo:**

Un processo iterativo è un ciclo di lavoro ripetuto in cui un team crea, prototipizza, testa e migliora un prodotto o un progetto fino a raggiungere il risultato desiderato. Questo approccio è molto comune nello sviluppo software, dove si suddivide il progetto in cicli o iterazioni più piccoli, permettendo di ricevere feedback e apportare modifiche in corso d'opera.

L

- **Log:**
Registro cronologico di eventi e operazioni eseguite su un sistema.

M

- **Mentore:**

Figura esperta che supporta uno sviluppatore o un team junior nell'ambito dell'ingegneria del software, offrendo guida tecnica, consigli pratici e orientamento nel processo di crescita professionale.

- **Merge:**

Operazione che unisce due rami di sviluppo in un unico ramo.

- **MFA (Autenticazione a più fattori):**

Meccanismo di autenticazione che richiede più elementi di verifica per confermare l'identità di un utente.

- **Microservizi:**

Architettura software composta da piccoli servizi indipendenti che comunicano tra loro.

- **Modulare:**

Caratteristica di un sistema progettato come insieme di componenti indipendenti e riutilizzabili.

- **Monitoraggio:**

Processo di osservazione, controllo e valutazione delle prestazioni di un sistema, di un componente o di un servizio informatico. Questo processo permette di monitorare lo stato attuale e le prestazioni di un sistema per individuare eventuali problemi o anomalie.

N

- **Nodo:**

Unità del sistema distribuito che può operare in modo autonomo e sincronizzarsi con il sistema centrale.

O

- **Orchestrazione:**

Coordinamento automatico di più servizi o componenti di un sistema complesso.

- **Ordine:**

Richieste formali per trasferimenti o rifornimenti di prodotti.

P

- **Parola:**

Termine utilizzato anche per riferirsi a una password o a un identificatore.

- **Piano di Qualifica:**

Documento che ha lo scopo di definire le modalità di qualifica e verifica del prodotto software. Esso rappresenta un indice di valutazione del prodotto software, delle sue componenti e degli strumenti utilizzati, al fine di garantire la qualità del prodotto finale e un continuo miglioramento del processo di sviluppo.

- **Prodotto:**

Oggetto fisico stoccato in un magazzino, trasferibile, vendibile o utilizzabile.

- **Progettista:**

Con un impegno orario medio-alto, garantisce una solida base al progetto. Trasforma i requisiti in un design architetturale strutturato, definendo l'organizzazione del sistema. Un progetto mal progettato può compromettere lo sviluppo, rendendo cruciale il suo ruolo. Assicura soluzioni affidabili, efficienti, sicure e conformi ai requisiti degli utenti finali.

- **Programmatore:**

Figura centrale nello sviluppo, dedica il maggior numero di ore al progetto. È responsabile dell'implementazione del software, rispettando specifiche e requisiti, in collaborazione con il Progettista. Cura l'ottimizzazione del codice per migliorarne le prestazioni e sviluppa funzionalità avanzate, richiedendo competenze nei linguaggi di programmazione, nelle tecnologie adottate e nella risoluzione di problemi tecnici.

- **Proof of Concept (PoC):**

Dimostrazione pratica, limitata e sperimentale, utilizzata per verificare la fattibilità tecnica o funzionale di un'idea, tecnologia o soluzione software. In ingegneria del software, una PoC serve a validare concetti chiave prima di investire risorse nello sviluppo completo, mostrando che un certo approccio può funzionare in un contesto reale. Non è un prodotto finito, ma uno strumento per ridurre il rischio progettuale e supportare decisioni tecniche o strategiche.

- **Protocollo:**

Insieme di regole che regolano la comunicazione tra dispositivi di rete o componenti software.

R

- **Rebase:**

Soprattutto nel contesto di Git, «rebase» significa ricalcolare o riapplicare i commit di un branch (una linea di sviluppo) su un altro branch, creando una cronologia lineare e pulita. In pratica, si muovono i commit di un branch su un altro, come se fossero stati fatti in sequenza invece di essere stati originati da un punto di divergenza.

- **Recovery:**

Indica il processo di ripristino di dati, applicazioni o sistemi informatici dopo un malfunzionamento, un errore o un disastro. Questo può includere il ripristino di dati da un backup, il recupero di file cancellati, o la riattivazione di un sistema interrotto.

- **Refactoring:**

È una tecnica di ingegneria del software che consiste nel ristrutturare il codice sorgente di un programma, senza alterare il suo comportamento esterno, per migliorarne la qualità, leggibilità e manutenibilità. L'obiettivo è rendere il codice più semplice, efficiente e più facile da comprendere per gli sviluppatori.

- **Repository:**

Archivio centralizzato e strutturato che consente la conservazione, la gestione e la condivisione di risorse digitali, quali codice sorgente, documentazione, librerie, configurazioni e dati. Garantisce la collaborazione tra più sviluppatori, l'integrità e la tracciabilità del progetto nel tempo.

- **Requirements and Technology Baseline (RTB):**

Nel contesto dello sviluppo software, la baseline di progetto è un riferimento stabile che definisce i requisiti del sistema e le tecnologie impiegate per la loro realizzazione. Comprende la specifica dei requisiti funzionali e non funzionali, l'elenco delle tecnologie, delle librerie e delle piattaforme adottate, nonché le strategie di sviluppo e testing. Questa baseline è essenziale per il controllo dell'avanzamento del progetto e per la verifica della conformità del prodotto rispetto ai requisiti concordati con il cliente.

- **Requisito:**

Specifica condizione o capacità che un sistema deve soddisfare per rispondere ai bisogni di business o utente.

- **Responsabile:**

Garantisce il completamento efficiente del progetto, rispettando obiettivi, tempistiche e aspettative del committente, coordinando ruoli e risorse. Data la rilevanza economica della posizione, il team ne ottimizza l'impiego per massimizzare l'efficienza e contenere i costi. Il ruolo richiede competenze nella gestione, risoluzione di criticità, coordinamento della comunicazione e pianificazione. Spetta inoltre al Responsabile approvare il rilascio di prodotti parziali o finali, software o documenti.

- **Riassortimento:**

Processo di rifornimento delle scorte per mantenere livelli adeguati di disponibilità. Può essere attivato manualmente o in modo automatico.

- **Ridondanza:**

Duplicazione di componenti critici per garantire la continuità di servizio anche in caso di guasti.

- **Ripristino:**

Procedura per riportare un sistema operativo o dati persi allo stato precedente al guasto

S

- **Scalabilità:**
Capacità del sistema di aumentare prestazioni e risorse in base alla domanda.
- **Scorte:**
Quantità di prodotti o materiali presenti in magazzino in un dato momento.
- **Scrum:**
È un framework agile che definisce un modo di lavorare per gestire progetti di sviluppo software in modo iterativo e incrementale. Si tratta di un approccio strutturato che aiuta i team a collaborare in modo efficace e a fornire risultati di alta qualità.
- **Sensibile:**
Dato che richiede protezione speciale contro accessi non autorizzati o divulgazioni.
- **Servizio:**
Componente funzionale del sistema, responsabile ciascuno di una specifica attività.
- **Sprint:**
Periodo di tempo specifico (timebox) in cui un team di sviluppo lavora in modo coordinato, seguendo una metodologia agile come Scrum, per completare una determinata quantità di lavoro.

T

- **Trasferimento:**

Operazione che comporta lo spostamento fisico di prodotti tra magazzini.

V

- **Validazione:**

È il processo di verificare che un software, un sistema o un processo informatico soddisfi determinati requisiti, standard o aspettative, e che sia idoneo all'uso previsto. Viene fatto una volta completato il progetto, confrontando il prodotto finale con le specifiche concordate.

- **Verifica:**

È un'attività continua. Accerta che l'esecuzione delle attività attuate nel periodo concordato non abbia introdotto errori. La verifica viene eseguita ad ogni avanzamento intermedio meritevole di attenzione.

- **Verificatore:**

Assicura l'affidabilità dei test e migliora la qualità del software e della documentazione, rispettando gli standard concordati. Conoscendo il metodo di lavoro, garantisce l'esecuzione precisa di ogni task secondo le best practice. Il suo ruolo è continuo, assicurando il rispetto delle aspettative del cliente e il successo del progetto.

- **Vulnerabilità:**

Debolezza o difetto che può essere sfruttato per compromettere la sicurezza o il funzionamento del sistema.

W

- **Way of Working:**

Approccio operativo e insieme di pratiche adottate da un team o organizzazione per gestire e completare un progetto.