



Carbon12 – Predire in Grafana

Glossario Interno

Informazioni sul documento

Versione	1.0.0
Stato	Approvato
Data di creazione	2019/11/29
Data di approvazione	2020/01/13
Redazione	Giacomo Callegari Manuel De Franceschi Nicolò Fassina Andrea Longo Alessandro Lovo
Verifica	Francesco Gobbo Veronica Pederiva
Approvazione	Giacomo Callegari
Uso	Interno
Destinatari	Carbon12 Prof. Tullio Vardanega Prof. Riccardo Cardin

Scopo del documento

Definizione dei termini di glossario presenti nei documenti interni.

Indice

1. A 1

2. B..... 1

3. C 1

4. D 2

6. F..... 3

7. G 3

8. H..... 3

9. I..... 3

10. J 3

11. K 3

12. L..... 4

13. M..... 4

14. N 4

15. O 4

16. P..... 4

17. Q 5

18. R 5

19. S 5

20. T..... 6

21. U 6

22. V 6

23. W 6

24. X 6

25. Y 6

26. Z..... 6

1. A

AIRBNB JAVASCRIPT STYLE GUIDE

Insieme di standard che delineano come il codice dovrebbe essere scritto e organizzato. Lo scopo è quello di permettere ad altri sviluppatori di comprendere più facilmente e rapidamente il codice.

API RESTFUL

Application Programming Interface (API) basata sul modello architetturale REST (Representational State Transfer), che utilizza richieste HTTP per gestire dati remoti.

2. B

BEHAVIOR DRIVEN DEVELOPMENT (BDD)

Processo di sviluppo software Agile che combina tecniche e principi di test-driven development per fornire agli sviluppatori e ai Business analysts degli strumenti di collaborazione.

BEST PRACTICE

Way of working noto, fatto di esperienze, procedure e azioni, che abbia mostrato di garantire l'ottenimento dei migliori risultati in contesti noti e specifici relativamente agli obiettivi preposti.

BLOCKCHAIN

Registro digitale condiviso e immutabile, strutturato in blocchi concatenati in ordine cronologico, in cui vengono archiviate in modo sicuro transazioni utilizzando principi crittografici.

BUSINESS APPLICATION LANGUAGE (BAL)

Linguaggio di programmazione che ha caratteristiche di basso livello, ma rende disponibili comandi più evoluti che lo rendono più semplice da utilizzare rispetto a un linguaggio assembler.

BUSINESS DOMAIN LANGUAGE (BDL)

Linguaggio utilizzato in determinati tipi di business, per esempio in documenti di testo rilevanti per un'azienda.

3. C

CHANGELOG

Registro delle modifiche apportate ad un progetto.

CHANGE SIGNIFICANCE

Modello di versionamento che classifica le modifiche in base al grado di importanza. La prima cifra viene modificata per cambiamenti di maggiore rilevanza, mentre le modifiche alle cifre successive indicano cambiamenti di importanza via via decrescente.

CICLO DI DEMING (PDCA – Plan Do Check Act)

Metodo di gestione dei processi iterativo incentrato sul principio del miglioramento continuo e basato sulle quattro attività di Pianificazione delle aree di intervento per il miglioramento, Esecuzione delle azioni migliorative, Valutazione dell'esito delle azioni effettuate e Azione, ovvero standardizzazione del comportamento se il processo di miglioramento ha avuto successo oppure ripetere le azioni per ottenere un ulteriore miglioramento.

CICLO DI VITA

Stati che attraversa lo sviluppo di un prodotto dal concepimento fino al ritiro dal mercato.

4. D

DASHBOARD (in Grafana)

Singola finestra di Grafana che può contenere uno o più pannelli di plug-in.

DESIGN PATTERN

Modello riutilizzabile che descrive un problema ricorrente e propone una possibile soluzione in termini di organizzazione di classi/oggetti che si è rilevata efficace a risolvere il problema stesso.

DIAGRAMMA DI GANTT

Il diagramma di Gantt è uno strumento di supporto alla gestione dei progetti. Un diagramma di Gantt permette 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

DRIVER

Componente software che simula una funzionalità per testare le sue sotto-funzionalità.

5. E

EDGE DATA POINT

Dispositivi che effettuano le misurazioni di diversi parametri ed inviano i dati ad un database centralizzato. Questi possono essere sia dei dispositivi fisici che dei simulatori a livello software.

EFFICACE

Capacità di un'attività di produrre i risultati voluti e raggiungere gli obiettivi richiesti o desiderati.

EFFICIENTE

Capacità di un'attività di raggiungere gli obiettivi attesi, impiegando le risorse minime indispensabili.

EFFICIENZA

Misura dell'abilità di raggiungere l'obiettivo atteso, con il minor utilizzo di risorse.

ETHEREUM

Piattaforma decentralizzata per la creazione e pubblicazione peer-to-peer di smart contracts, creati attraverso un linguaggio di programmazione.

6. F

FRAMEWORK

Un'architettura logica che supporta il programmatore nelle fasi di progettazione e sviluppo di un software. Fornisce funzionalità aggiuntive come estensione del linguaggio e può includere librerie e software di supporto.

7. G

GARANZIA DELLA QUALITÀ

Processo atto a garantire la qualità delle attività di sviluppo del software e del prodotto realizzato mediante la verifica di conformità rispetto agli standard o ai livelli di qualità prestabiliti. La garanzia della qualità comprende l'intero processo di sviluppo del software: analisi dei requisiti, progettazione, codifica, verifica e validazione, gestione della configurazione, testing, gestione del versionamento e dell'integrazione del prodotto. Il processo di garanzia della qualità prevede la definizione di obiettivi, capacità, attività, metriche e strumenti atti a misurare la qualità dello sviluppo del software.

GRAFANA

Software Open Source che consente di generare grafici e dashboard per il monitoraggio di ambienti e di sistemi.

8. H

9. I

ISSUE TRACKING SYSTEM

Software di tracciamento delle attività comprese nel processo di sviluppo di un software, che permette la loro organizzazione in modo efficiente. L'ITS permette categorizzare le attività, per esempio task, bug, test, change request, assegnare delle priorità e gestire la risoluzione delle attività in modo ordinato.

10. J

JSON

JSON (JavaScript Object Notation) è un semplice formato per lo scambio di dati indipendente dal linguaggio di programmazione usato.

11. K

12. L

13. M

MACHINE LEARNING

Branca dell'Intelligenza Artificiale che si basa sull'idea che i sistemi possono imparare dai dati e prendere decisioni autonomamente o con un intervento umano limitato.

METADATO

Informazione che descrive un insieme di dati.

MODELLO DI SVILUPPO

Principio teorico che indica il metodo da seguire nella progettazione software. È costituito da un insieme di processi che hanno lo scopo di aiutare gli sviluppatori nella realizzazione del prodotto.

14. N

15. O

16. P

PLUG-IN

Nell'ambito software, un plug-in è una componente che aggiunge specifiche funzionalità ad un programma già esistente.

PRODUCT BASELINE

Presenta la baseline architetturale del prodotto, coerentemente con quanto riportato nella Technology Baseline. Al suo interno contiene i diagrammi delle classi, delle attività e di sequenza e la contestualizzazione dei design pattern adottati nell'architettura del prodotto.

PROOF-OF-CONCEPT

Dimostratore eseguibile il cui scopo è quello di mostrare come la tecnologia individuata come soluzione per la realizzazione del prodotto sia utile allo sviluppo dello stesso e risponda alle aspettative.

17. Q

QUALITY ASSURANCE

È un processo di Supporto dello standard ISO/IEC 12207:1995 composto da attività volte a garantire il soddisfacimento degli obiettivi della qualità.

18. R

REGRESSIONE LINEARE (RL/LR – LINEAR REGRESSION)

Metodo di previsione statistica per stimare un valore numerico atteso condizionato dalla relazione esistente tra due o più fattori.

REPOSITORY

Ambiente di un sistema informativo, in cui vengono gestiti i metadati, attraverso tabelle relazionali.

RETE NEURALE (in IA)

Modello computazionale composto da numerose unità elaborative omogenee interconnesse che si ispirano al funzionamento delle reti neurali biologiche.

19. S

SERVERLESS

Termine che indica un network la cui gestione viene suddivisa fra i vari utenti che utilizzano il network stesso.

SMART CONTRACT

Applicativi che operano su tecnologie blockchain.

SNAKE CASE

Lo snake case è la pratica di scrivere gli identificatori separando le parole che li compongono tramite underscore, solitamente con le prime lettere delle singole parole in minuscolo, e la prima lettera dell'intero identificatore minuscola o maiuscola (es. "foo_bar" e "Hello_world").

STAKEHOLDER

Ciascuno dei soggetti direttamente o indirettamente coinvolti in un progetto o nell'attività di un'azienda.

STUB

Componente software che simula una sotto-funzionalità di una funzionalità, per permette di testare quest'ultima anche se la prima non è ancora stata realizzata.

SUPPORT VECTOR MACHINE (SVM)

Modelli di apprendimento supervisionato associati ad algoritmi di apprendimento per la regressione e la classificazione.

20. T

TECHNOLOGY BASELINE

Presenta la baseline tecnologica del prodotto, individua gli aspetti tecnici necessari ad incontrare la realizzazione delle funzionalità attese e definisce la suddivisione del sistema in componenti e la loro organizzazione.

TEST DI CARICO

Test di performance del prodotto effettuato aumentando costantemente il carico sul sistema tramite strumenti automatici. Ha lo scopo di individuare bug non emersi nelle attività di test precedenti ed assicurare che l'applicazione soddisfi le prestazioni attese.

21. U

UML

UML ("linguaggio di modellizzazione unificato") è un linguaggio di modellazione e di specifica basato sul paradigma orientato agli oggetti. Stabilisce un insieme di notazioni grafiche atte a supportare la descrizione e il progetto di sistemi software.

22. V

23. W

WAY OF WORKING

Il modo di rendere sistematiche, disciplinate e quantificabili le attività di progetto.

WIREFRAME

Indica la rappresentazione grafica dello scheletro di un progetto.

24. X

25. Y

26. Z