



Università degli Studi di Padova

Laurea: Informatica

Corso: Ingegneria del Software

Anno Accademico: 2024/2025



Gruppo: SWEg Labs

Email: gruppo.sweg@gmail.com

Glossario

Versione 1.0.0

| | |
|---------------------|--|
| Stato | Approvato |
| Redazione | Federica Bolognini Michael Fantinato Giacomo Loat Filippo Righetto Riccardo Stefani Davide Verzotto |
| Verifica | Federica Bolognini Michael Fantinato Giacomo Loat Filippo Righetto Riccardo Stefani Davide Verzotto |
| Approvazione | ... |
| Proprietario | Gruppo <i>SWEg Labs</i> |
| Uso | Esterno |
| Destinatari | Prof. Tullio Vardanega Prof. Riccardo Cardin <i>AzzurroDigitale Srl</i> |

Registro delle modifiche

| Versione | Data | Descrizione | Autore | Verifica |
|----------|----------|---|--------------------|-------------------|
| 1.0.0 | ... | Approvazione del documento | ... | ... |
| ... | ... | Verifica del documento | ... | ... |
| ... | ... | ... | ... | ... |
| 0.2.0 | ... | Verifica del documento allo stato attuale | ... | ... |
| 0.1.4 | 11-11-24 | Aggiunti i termini <u>ISO/IEC 31000:2018</u> , | Federica Bolognini | Michael Fantinato |
| 0.1.3 | 10-11-24 | Aggiunti i termini <u>Baseline</u> , <u>Branch</u> , <u>Feature</u> , <u>Git</u> , <u>IDE</u> , <u>Product Baseline (PB)</u> , <u>Proponente</u> , <u>Proof of Concept (PoC)</u> , <u>Merge</u> , <u>Minimum Viable Product (MVP)</u> , <u>Snake Case</u> e <u>Visual Studio Code (VSC)</u> | Riccardo Stefani | Michael Fantinato |
| 0.1.2 | 09-11-24 | Aggiunti i termini <u>Ciclo di vita del software</u> , <u>Processo</u> , <u>Verifica</u> e <u>Way of Working</u> | Riccardo Stefani | Giacomo Loat |
| 0.1.1 | 08-11-24 | Inserimento dei primi termini, tratti dai primi due verbali interni della RTB | Riccardo Stefani | Filippo Righetto |
| 0.1.0 | 05-11-24 | Creazione del documento | Riccardo Stefani | Filippo Righetto |

Tabella 1: Registro delle modifiche

Indice

| | | |
|----------|-----------------------------------|-----------|
| 1 | Introduzione | 1 |
| 1.1 | Obiettivo del documento | 1 |
| 1.2 | Struttura del documento | 1 |
| A | | 2 |
| B | | 3 |
| C | | 4 |
| D | | 5 |
| E | | 6 |
| F | | 7 |
| G | | 8 |
| H | | 9 |
| I | | 10 |
| J | | 11 |
| K | | 12 |
| L | | 13 |
| M | | 14 |
| N | | 15 |
| O | | 16 |
| P | | 17 |
| Q | | 18 |
| R | | 19 |
| S | | 20 |
| T | | 21 |
| U | | 22 |
| V | | 23 |
| W | | 24 |
| X | | 25 |
| Y | | 26 |
| Z | | 27 |

1 Introduzione

1.1 Obiettivo del documento

Questo documento è stato creato con l'intento di raccogliere tutti i termini che possano risultare sconosciuti ad un lettore esterno o che possano creare ambiguità. Lo scopo è quello di creare una base omogenea di conoscenza specifica.

1.2 Struttura del documento

Il documento è suddiviso in sezioni rappresentanti le lettere dell'alfabeto; all'interno di queste si troveranno i termini ambigui o specifici del progetto che sono stati contrassegnati nel resto della documentazione con un pedice_G alla fine della parola

A

Analisi dei Requisiti

Processo fondamentale dello sviluppo di un prodotto software che si concentra sulla raccolta, analisi e definizione delle necessità e delle aspettative degli utenti finali, degli stakeholder e del sistema nel suo complesso. Questo processo mira a comprendere e documentare in modo chiaro e completo le esigenze, le funzionalità, le prestazioni e i vincoli che il sistema deve soddisfare. L'obiettivo principale dell'analisi dei requisiti è fornire una base solida per tutte le fasi successive dello sviluppo del software, assicurando che il prodotto finale soddisfi le esigenze degli utenti e raggiunga gli obiettivi del progetto.

Analisi dei Rischi

L'analisi dei rischi è il processo di identificazione, valutazione e priorizzazione dei rischi in un progetto, sistema o attività, al fine di ridurre o gestire il loro impatto potenziale. Viene utilizzata per prevedere gli eventi negativi che potrebbero influenzare il successo di un progetto e per determinare le azioni preventive o correttive da intraprendere.

AzzurroDigitale

AzzurroDigitale è una società italiana con sede a Padova. Si occupa di digitalizzazione di processi sia con prodotti proprietari che di terze parti, e ha l'obiettivo di accompagnare le aziende manifatturiere nella transizione 5.0.

B

Backlog

Insieme di compiti/attività da completare per un certo obiettivo. All'interno del framework Scrum, ne esistono due tipi principali: il product backlog, che è la lista delle funzionalità da implementare, e lo sprint backlog, che contiene le attività da svolgere durante un particolare sprint. Un'attività interna al backlog porta valore ad un progetto perchè possiede:

- Stato, che segnala se l'attività è stata completata, in corso o non ancora iniziata.
- Priorità, che indica l'importanza dell'attività rispetto alle altre.
- Assegnatario, cioè una persona incaricata a svolgere l'attività. Questa assegnazione non è vincolante, infatti se un membro del team ha terminato la sua attività può prendersi a carico un'altra attività presente nel backlog anche se non era stata inizialmente assegnata a lui.
- Scadenza, cioè un termine temporale entro il quale l'attività deve essere svolta.

Baseline

Nel contesto dell'ingegneria del software, stato di avanzamento che rappresenta un insieme di punti di arrivo che ci si pone come obiettivo di raggiungere in una milestone, dimostrando che l'incremento delle modifiche condotte ha portato a un risultato.

Best Practices

Nello sviluppo software, metodologie che attraverso l'esperienza e la sperimentazione sono state identificate come modi efficaci e raccomandati di affrontare determinati problemi o compiti nel processo di sviluppo del software. Queste pratiche sono considerate migliori (best) perché hanno dimostrato di portare a risultati di alta qualità, facilitando la manutenzione del codice e promuovendo una migliore collaborazione nel team di sviluppo.

Branch

Letteralmente "ramo", indica un'entità che si sviluppa o si dirama da un punto principale. Nel contesto di un sistema di controllo delle versioni, un branch rappresenta una linea di sviluppo separata. Può essere utilizzato per sviluppare nuove funzionalità, risolvere bug o implementare modifiche senza influenzare direttamente il ramo principale del codice, noto come master o main.

C

Checklist

Lista dettagliata di elementi, attività o criteri specifici che devono essere controllati, esaminati o completati durante le diverse fasi del ciclo di vita del software. E' utilizzata come strumento di controllo e verifica.

Ciclo di vita del software

Serie di fasi attraverso le quali un software passa dal suo concepimento iniziale fino al suo ritiro o dismissione. È un concetto chiave nell'ingegneria del software e fornisce una struttura organizzativa per il processo di sviluppo del software.

D

Diagramma di Gantt

Strumento di visualizzazione temporale utilizzato nella gestione dei progetti per rappresentare le attività pianificate nel tempo. È composto da una barra orizzontale che rappresenta l'arco temporale totale del progetto e da barre orizzontali più piccole che rappresentano le singole attività del progetto. Ogni barra è posizionata lungo l'asse temporale in base alle date di inizio e fine previste per l'attività.

Diagramma UML

Acronimo di Unified Modeling Language, un diagramma UML diagramma utilizzato per modellare, descrivere e visualizzare sistemi software e processi di sviluppo software. È uno standard industriale nel campo dell'ingegneria del software e fornisce una serie di diagrammi, ognuno dei quali si concentra su un aspetto specifico del sistema o del processo. Sono diagrammi UML ad esempio i diagrammi dei casi d'uso, i diagrammi delle classi e i diagrammi delle funzionalità.

Diagramma UML dei casi d'uso

Diagramma UML che rappresenta le interazioni tra utenti (attori) e il sistema, descrivendo come gli utenti utilizzano il sistema per raggiungere obiettivi specifici. Ogni caso d'uso rappresenta una funzione o un'attività significativa, utile per descrivere i requisiti funzionali. È spesso il primo passo nella progettazione di un sistema software e aiuta a identificare le funzioni principali e il modo in cui il sistema interagisce con gli utenti.

Diagramma UML delle classi

Diagramma UML che descrive la struttura statica di un sistema, mostrando le classi, i loro attributi, i metodi e le relazioni tra di esse (come ereditarietà, associazioni e aggregazioni). Questo diagramma è fondamentale per la programmazione orientata agli oggetti poiché fornisce una rappresentazione visiva della struttura del codice, aiutando a comprendere le interconnessioni tra le varie entità e a definire i componenti principali.

Discord

Piattaforma VoIP (Voice over IP: tecnologia che rende possibile effettuare una conversazione sfruttando una connessione internet), messaggistica istantanea e distribuzione digitale progettata per la comunicazione.

Draw.io

Draw.io (ora chiamato diagrams.net) è uno strumento gratuito per la creazione di diagrammi, disponibile sia come applicazione web che come app desktop per vari sistemi operativi. Viene utilizzato ampiamente per progettare e documentare diagrammi di flusso, architetture software, diagrammi UML, organigrammi, mappe mentali, wireframe, e altri tipi di rappresentazioni visive.

E

...

F

Feature

Una feature è una specifica funzionalità o capacità di un sistema software che ne arricchisce il comportamento e le possibilità d'uso. Le feature sono aggiunte o miglioramenti che rispondono a esigenze degli utenti, migliorano l'esperienza d'uso, o risolvono specifici problemi. L'introduzione o il miglioramento di una feature segue un ciclo di sviluppo completo, dalla raccolta dei requisiti fino alla verifica e al testing.

Fogli Google

Fogli Google è un'applicazione web di Google, parte della suite di produttività Google Workspace, che consente di creare, modificare e condividere fogli di calcolo online. È uno strumento particolarmente apprezzato per il suo accesso immediato da browser, le funzionalità collaborative in tempo reale, e l'integrazione con altri servizi Google. Così come ogni foglio di calcolo, include la possibilità di creare grafici e diagrammi basati su dati tabellari.

G

Gantt, diagramma di

Vedi *Diagramma di Gantt* [G](#).

Git

Software per il controllo di versione distribuito utilizzabile tramite interfaccia a riga di comando.

GitHub

Servizio di hosting per progetti software. Il sito è principalmente utilizzato da sviluppatori che caricano il codice sorgente di programmi in dei repository e lo rendono scaricabile e migliorabile da altri sviluppatori. Questi ultimi possono interagire con i proprietari dei repository tramite un sistema per inviare segnalazioni di bug o richieste di funzionalità (issue tracker), un sistema per copiare il software in una versione modificabile (fork), un sistema per proporre modifiche agli sviluppatori originali (pull request) e un sistema di discussione legato al codice del repository (commenti).

GitHub Pages

GitHub Pages è un servizio di hosting gratuito offerto da GitHub che permette agli utenti di creare e pubblicare facilmente siti web statici direttamente dai loro repository GitHub. È utilizzato comunemente per creare siti di documentazione, pagine personali o di progetto, blog e portali di portfolio. GitHub Pages è particolarmente apprezzato perché permette di ospitare un sito senza costi e con aggiornamenti automatici ogni volta che il repository viene modificato.

GitHub Projects

GitHub Projects è uno strumento di gestione dei progetti integrato in GitHub, ideato per aiutare sviluppatori e team a organizzare, pianificare e tracciare il lavoro sui progetti direttamente all'interno dell'ambiente GitHub. Si basa su un sistema flessibile di "project board" simile a Kanban, che offre un modo visuale per coordinare i task e monitorare l'avanzamento del lavoro. GitHub Projects è uno strumento ideale per team che lavorano su progetti di sviluppo software, in quanto permette di gestire l'intero processo di sviluppo all'interno di GitHub stesso. Grazie alla sua integrazione nativa con il codice, le issues e le pull requests, aiuta a mantenere sincronizzati i task e a ridurre il contesto di cambiamento per gli sviluppatori, migliorando il flusso di lavoro e facilitando la collaborazione su GitHub.

Glossario

Elenco organizzato di termini tecnici, acronimi e definizioni utilizzati nel contesto del progetto. Questo documento fornisce una chiara comprensione dei concetti e dei linguaggi specifici impiegati nel progetto, aiutando a ridurre ambiguità e fraintendimenti tra i membri del team e gli stakeholder.

Gulpease, indice di

Indice di leggibilità di un testo specificamente in lingua italiana, che utilizza il numero delle parole, delle frasi e delle lettere per facilitare il calcolo automatico della leggibilità.

H

...

I

IDE

Acronimo di Integrated Development Environment, è un ambiente di sviluppo, ovvero un software che, in fase di programmazione, supporta i programmatori nello sviluppo e debugging del codice sorgente di un programma, segnalando errori di sintassi del codice direttamente in fase di scrittura, oltre a fornire una serie di strumenti e funzionalità di supporto alla fase stessa di sviluppo e debugging.

Issue

Una issue su GitHub (e altre piattaforme di gestione del codice e dei progetti) è un elemento utilizzato per tracciare problemi, richieste di funzionalità, idee o miglioramenti relativi a un progetto. È uno strumento fondamentale per organizzare il lavoro collaborativo e garantire che tutti i membri del team siano aggiornati sui task e le priorità. Ogni issue rappresenta un singolo elemento che richiede attenzione o azione, e fornisce un luogo centralizzato per discuterlo, seguirlo e risolverlo.

ISO/IEC 31000:2018

La norma ISO 31000 "Risk management - Principles and guidelines", in italiano UNI ISO 31000 Gestione del rischio - Principi e linee guida. È una guida che fornisce principi e linee guida generali per la gestione del rischio. Può essere utilizzata da qualsiasi organizzazione pubblica, privata o sociale, associazione, gruppo o individuo, e non è specifica per nessuna industria o settore. La ISO 31000 può essere applicata nel corso dell'intero ciclo di vita di un'organizzazione, ed essere adottata per molte attività come la definizione di strategie e decisioni, operazioni, processi, funzioni, progetti, prodotti, servizi e beni. Può inoltre essere applicata a qualsiasi tipo di rischio, sia per conseguenze di tipo positivo che negativo.

J

Jira

Jira è uno strumento di gestione dei progetti e di issue tracking sviluppato da Atlassian, ampiamente utilizzato per la pianificazione, il monitoraggio e il controllo di progetti, in particolare nell'ambito dello sviluppo software. Nato come strumento di gestione dei bug e dei problemi, Jira è diventato uno dei principali strumenti per il project management, soprattutto per le organizzazioni che adottano metodologie agili come Scrum e Kanban.

K

...

L

LaTeX

Linguaggio di marcatura per la preparazione di testi, basato sul programma di composizione tipografica TeX. LaTeX è ampiamente utilizzato per la creazione di documenti scientifici e tecnici grazie alla sua capacità di gestire formule matematiche complesse e alla sua alta qualità tipografica.

M

Merge

Operazione fondamentale nei sistemi di controllo delle versioni. Essa è utilizzata per combinare le modifiche apportate in due branch separati in un singolo branch.

Milestone

In ingegneria del software e nella gestione dei progetti, punto di riferimento o traguardo significativo che sancisce il termine di un periodo nel ciclo di vita di un progetto. Le milestone rappresentano generalmente eventi chiave, compimenti o obiettivi importanti che indicano il progresso del progetto. L'obiettivo che ci si pone durante una milestone è realizzare una baseline.

Minimum Viable Product (MVP)

Versione ridotta di un prodotto, che incorpora solo le funzioni essenziali per soddisfare le esigenze di base. Viene utilizzato per rilasciare un prodotto come test e ricevere feedback dall'utenza per migliorare poi il prodotto finito con tutte le funzionalità.

N

Norme di Progetto

Insieme di linee guida, procedure e regole stabilite per regolare e standardizzare l'approccio, il processo e l'output del lavoro all'interno del progetto. Queste norme possono riguardare diversi aspetti del progetto, come la gestione del codice, la documentazione, la comunicazione e la gestione dei rischi. L'obiettivo delle norme di progetto è promuovere la coerenza, la qualità e l'efficienza nel processo di sviluppo del software, consentendo al team di lavorare in modo più efficace e collaborativo.

O

...

P

Piano di Progetto

Documento formale che delinea in dettaglio la pianificazione, l'esecuzione, il monitoraggio e il controllo di tutte le attività coinvolte nella realizzazione di un progetto. Questo documento fornisce una roadmap chiara e organizzata, comprensiva di obiettivi, risorse, scadenze e strategie di gestione dei rischi. Essenziale per la gestione efficace di un progetto, il piano di progetto serve come guida per il team di lavoro e gli stakeholder, fornendo una struttura che facilita il coordinamento delle attività e l'assegnazione delle risorse.

Piano di Qualifica

Documento che stabilisce gli standard di qualità, i processi e le attività di testing che saranno implementati durante lo sviluppo di un progetto. Contiene una descrizione dettagliata delle strategie di testing, delle metriche di valutazione e dei criteri di accettazione del prodotto finale. L'obiettivo principale del Piano di Qualifica è garantire che il prodotto soddisfi gli standard di qualità prefissati e che il processo di sviluppo segua procedure coerenti ed efficaci.

Processo

Insieme strutturato di attività necessarie per lo sviluppo di un sistema software.

Product Baseline (PB)

È la seconda revisione di avanzamento del progetto didattico. Comprende un prodotto software con design definitivo, chiamato *Minimum Viable Product (MVP)_G*.

Proof of Concept (PoC)

Versione preliminare di un'applicazione o di una soluzione software che viene sviluppata per dimostrare la fattibilità tecnica di un'idea o di un concetto. Viene utilizzata per testare rapidamente l'efficacia di un approccio, identificare eventuali limitazioni delle tecnologie scelte e valutare se l'idea può essere realizzata in modo pratico.

Proponente

Nel contesto dell'ingegneria del software, colui che presenta un'idea, un progetto o una proposta e ne sostiene la realizzazione. Il gruppo *SWEg Labs* ha come proponente l'azienda *AzzurroDigitale_G*.

Q

...

R

Repository

In termini informatici, un luogo o un archivio dove vengono conservati e gestiti dati, documenti o, nel contesto del software, il codice sorgente di un progetto. Nell'ambito dei sistemi di controllo delle versioni come Git, un repository è una struttura dati che archivia anche la cronologia completa delle modifiche apportate al codice sorgente di un progetto.

Requirement and Technology Retrospective (RTB)

La prima revisione di avanzamento del progetto didattico. Fissa i requisiti da soddisfare in accordo con il proponente, motiva le tecnologie, i framework e le librerie adottate dimostrandone adeguatezza e compatibilità tramite il Proof of Concept (PoC).

Retrospettiva

Vedi *Sprint Retrospective*_G.

S

Script

Uno script è un insieme di istruzioni scritte in un linguaggio di scripting (come Python, JavaScript, Bash, ecc.) progettato per automatizzare compiti specifici o per eseguire operazioni in modo diretto su un sistema, applicazione o ambiente web. A differenza di un programma complesso, uno script tende a essere più leggero e adatto per operazioni mirate e ripetibili, senza richiedere una fase di compilazione lunga o complessa.

Snake Case

Lo Snake Case è uno stile di scrittura di nomi in cui le parole sono separate da un trattino basso o underscore (-) e tutte le lettere sono generalmente minuscole (es.: nome_variabile_uno). Viene spesso contrapposto allo stile Camel Case, in cui invece le parole successive alla prima vengono concatenate con l'iniziale maiuscola (es.: nomeVariabileUno).

Sprint

Periodo di tempo prefissato entro il quale lavorare producendo dei risultati documentati. Gli sprint sono al centro delle metodologie Agile, atte a produrre risultati piccoli e in maniera costante.

Sprint Retrospective

Incontro finalizzato ad analizzare l'andamento dello sprint quando questo è terminato, per migliorare la performance futura del team di sviluppo. La riunione retrospettiva dello sprint è quindi propedeutica allo sprint successivo.

T

Tracciamento

Il tracciamento è il processo di monitoraggio e documentazione dell'evoluzione e delle modifiche di un sistema, dei suoi requisiti e dei suoi componenti, all'interno di un progetto. Questo processo è essenziale per mantenere una visione chiara dello stato attuale del sistema, delle sue modifiche, e delle relazioni tra diversi elementi (come requisiti, attività di design, e componenti di codice).

U

...

V

Verifica

Processo che include un insieme di attività volte a garantire che il lavoro svolto durante lo sviluppo del software rispetti gli standard, i requisiti e le aspettative stabilite. La verifica è essenziale per garantire che il software sia di alta qualità, risponda alle esigenze degli utenti e riduca il rischio di errori e difetti. Si svolge a periodi regolari durante il corso del progetto.

Versionamento

Processo che realizza il cosiddetto “controllo di versione”, stabilendo la storia cronologica delle azioni fatte per una certa attività, tracciando i cambiamenti occorsi e permettendo di tornare a uno stadio precedente qualora necessario.

Visual Studio Code (VSC)

IDE _G libero e gratuito sviluppato da Microsoft per Windows, Linux e macOS. Permette sia di scrivere codice sorgente per un prodotto software sia di scrivere testo di documentazione.

W

Way of working

Il termine “Way of working” (modo di lavorare) si riferisce al modo in cui un individuo, un team o un’organizzazione svolge le proprie attività lavorative. Questo concetto può includere processi, metodologie, abitudini, strumenti e culture aziendali che influenzano la gestione del lavoro e la collaborazione. Un modo di lavorare efficace ed efficiente può migliorare la produttività, la qualità del lavoro e la soddisfazione dei membri del team, promuovendo un ambiente di lavoro positivo e collaborativo.

Il way of working deve essere:

- *Sistematico*, cioè non deve dipendere dalle singole persone, bensì deve essere adattabile anche a futuri membri del team.
- *Disciplinato*, cioè deve far seguire norme di comportamento che blocchino l’istinto e favoriscano il ragionamento.
- *Quantificabile*, cioè si deve poter verificare concretamente la sua implementazione.

X

...

Y

...

Z

...