



On-By-One

Glossario

On-By-One:

Sara Ballardin	2101048
Shuhui Chen	2138001
Simone Giuliano	2145591
Alvise Greggio	2141004
Daniele Nuzzaci	2139039
Federico Temporin	2146764
Tommaso Zorzetto	2148629

Informazioni documento			
Versione	Data	Stato	Destinatari
1.0.1	2026/10/29	Approvato	On-By-One Prof. Tullio Vardanega Prof. Riccardo Cardin

Contatti: onbyone.swe@gmail.com

Registro delle modifiche

Versione	Data	Autore	Descrizione	Verificatore
1.0.1	2026/10/29	Sara Ballardin	Modifica path immagini e stile documento	Shuhui Chen
1.0.0	2026/10/29	Tommaso Zorzetto	Stesura glossario e inserimento dei primi termini tratti dal WoW	Federico Temporin

Indice

1	Introduzione	3
1.1	Obiettivo del documento	3
1.2	Struttura del documento	3
A		4
B		5
C		6
D		8
E		9
F		9
G		10
H		10
I		11
K		11
L		12
M		12
N		13
O		13
P		13
R		14
S		15
T		15
V		17
W		18

1 Introduzione

1.1 Obiettivo del documento

Questo documento è stato redatto con l'obiettivo di raccogliere, definire e spiegare in modo chiaro e univoco tutti i termini tecnici, gli acronimi, i ruoli e i concetti (inclusi quelli di base legati al controllo di versione e al flusso di lavoro) utilizzati all'interno della documentazione e del *repository* del gruppo **On-By-One**. Lo scopo principale è prevenire ambiguità interpretative e costruire una base di conoscenza condivisa e omogenea tra tutti i membri del *team* e gli *stakeholder*.

1.2 Struttura del documento

Il documento è organizzato in sezioni ordinate alfabeticamente. All'interno di ciascuna lettera sono elencati i termini specifici o potenzialmente ambigui incontrati nei documenti del progetto. In conformità con le convenzioni di formattazione adottate dal gruppo:

- Il *corsivo* viene impiegato alla prima occorrenza di termini stranieri poco comuni o specialistici e ad ogni occorrenza di espressioni italiane usate con un significato tecnico specifico all'interno del progetto.
- Il `codice inline` viene utilizzato per indicare nomi di file e *directory*, *branch*, prefissi, comandi, identificatori e brevi frammenti di codice.

A

Admin (prefisso)

Prefisso utilizzato nelle convenzioni di *branching* (`admin/`) e nei messaggi di *commit* (`admin:`) del *repository* per identificare attività di natura organizzativa e amministrativa. Rientrano in questa categoria le modifiche ai documenti relativi al *Way of Working*, la riorganizzazione delle *directory* e la configurazione o manutenzione dei *workflow* automatici.

Approvazione

Atto formale mediante il quale almeno un membro del *team* diverso dall'autore (il *revisore*) attesta, al termine della *code review*, che le modifiche proposte in una *Pull Request* soddisfano i requisiti dell'attività e rispettano le convenzioni del gruppo. L'approvazione è condizione necessaria per procedere al *merge* nel *branch main* e per aggiornare lo stato di un documento formale da *Bozza* ad *Approvato*.

Assegnatario

Membro del gruppo a cui viene affidata la responsabilità operativa di portare a termine una specifica attività (*task*) o di risolvere una *issue* entro una scadenza concordata.

Attività (Task)

Unità elementare di lavoro pianificata dal gruppo per raggiungere un determinato obiettivo progettuale, documentale o tecnico. Ogni attività che comporta modifiche al *repository* viene tracciata tramite una *issue*, sviluppata su un *branch dedicato* e valutata secondo i criteri della *Definition of Done*.

Autore

Membro del *team* che redige o modifica un documento, scrive codice o apre una *Pull Request*. L'autore è responsabile della qualità e della correttezza delle modifiche proposte, del rispetto della *Definition of Done*, della compilazione dei campi di propria competenza nel *Registro delle modifiche* e della risoluzione delle osservazioni ricevute durante la *code review*.

B

Bozza

Valore del campo *Stato* nella tabella *Informazioni documento* che identifica un documento non ancora approvato o sottoposto a modifiche rispetto all'ultima versione approvata. Lo stato rimane impostato su **Bozza** per tutta la durata della stesura e della fase di revisione, fino all'approvazione finale da parte del *revisore*.

Branch

Letteralmente “ramo”, rappresenta una linea di sviluppo indipendente e parallela all'interno di un sistema di controllo di versione come *Git*. L'utilizzo dei *branch* permette ai membri del *team* di lavorare contemporaneamente a nuove funzionalità, documenti o correzioni in un ambiente isolato, senza alterare la stabilità del *branch principale* (**main**) fino al momento dell'integrazione finale (*merge*).

Branch dedicato (o di lavoro)

Branch temporaneo creato a partire dal *branch main* aggiornato per sviluppare una singola attività o risolvere una specifica *issue*. Secondo le convenzioni del gruppo, il suo nome deve essere scritto in minuscolo separando le parole con trattini (*Kebab Case*) secondo il formato `<tipo>/<numero-issue>-<descrizione>` (ad esempio `docs/24-aggiornamento-readme`), e viene eliminato una volta completato il *merge*.

Branch di destinazione

Il *branch* (nella maggior parte dei casi **main**) verso il quale viene aperta una *Pull Request* e all'interno del quale confluiranno le modifiche provenienti dal *branch di lavoro* al termine della revisione.

Branch principale (main)

Il ramo ufficiale, stabile e di riferimento del *repository*. Contiene esclusivamente materiale verificato e approvato; su di esso sono vietati i *push* diretti e ogni modifica deve essere integrata tramite una *Pull Request* che abbia superato la *code review* e i controlli automatici.

Branching (Strategia di)

L'insieme delle regole, delle convenzioni di denominazione e delle procedure operative adottate dal *team* per creare, gestire, aggiornare e unire i *branch* nel *repository*, con l'obiettivo di mantenere una cronologia *Git* ordinata e proteggere l'integrità del ramo principale.

C

Candidatura

Fase iniziale del progetto didattico durante la quale il gruppo definisce la propria struttura organizzativa e il proprio *Way of Working*, analizza i capitolati proposti e redige i documenti e i verbali necessari per candidarsi alla realizzazione di uno specifico capitolato.

Closes (Keyword)

Parola chiave (*keyword*) riconosciuta automaticamente da *GitHub* che, se inserita nella descrizione di una *Pull Request* o nel messaggio di un *commit* seguita dal riferimento numerico di una *issue* (ad esempio **Closes #46**), provoca la chiusura automatica dell'*issue* stessa non appena la *Pull Request* viene integrata nel *branch* principale.

Code review (Revisione del codice)

Processo di verifica sistematica applicato a ogni *Pull Request* prima della sua integrazione nel *branch main*. Viene svolto da almeno un membro del *team* diverso dall'autore (*revisore*) e mira ad accertare la correttezza, la completezza, la chiarezza espositiva, la coerenza terminologica e il rispetto delle convenzioni di formattazione e dei *template* di progetto.

Codice inline

Stile di formattazione tipografica (ottenuto racchiudendo il testo tra accenti gravi in *Markdown* o tramite carattere monospazio in *LaTeX*) utilizzato all'interno di un testo discorsivo per distinguere visivamente nomi di file e *directory*, nomi di *branch*, comandi, identificatori e brevi frammenti di codice.

Commento bloccante

Osservazione formulata dal *revisore* durante una *code review* per segnalare problemi critici, quali errori sostanziali nei contenuti o nel codice, requisiti dell'*issue* non soddisfatti o violazioni delle norme del *repository*. Tutti i commenti bloccanti devono essere obbligatoriamente risolti dall'autore e verificati dal *revisore* prima del *merge*.

Commento non bloccante

Osservazione formulata dal *revisore* durante una *code review* che propone miglioramenti facoltativi, ottimizzazioni o suggerimenti stilistici che non pregiudicano la correttezza del lavoro. L'autore e il *revisore* possono concordare se applicare subito la modifica nella *Pull Request* corrente oppure rinviarla a un'attività futura senza bloccare il *merge*.

Commit

Operazione fondamentale di *Git* che registra nella cronologia del *branch* corrente un'istantanea (*snapshot*) di un insieme logicamente coerente e circoscritto di modifiche apportate ai file del *repository*. Ogni *commit* è identificato da un codice univoco ed è accompagnato da un messaggio descrittivo strutturato secondo la convenzione `<tipo>: <descrizione>` (ad esempio `docs: aggiornate le convenzioni di formattazione (#24)`).

Compilazione

Processo automatico mediante il quale un programma trasforma file sorgente scritti in un linguaggio formale o di marcatura in un documento finale pronto per la distribuzione o l'esecuzione. Nel contesto del progetto, si riferisce in particolare alla generazione dei documenti PDF a partire dai file sorgente *LaTeX* (`.tex`).

Conflitto (Merge conflict)

Situazione che si verifica in *Git* quando si tenta di unire due *branch* che contengono modifiche divergenti e incompatibili sulle stesse righe di un medesimo file. Prima di poter completare il *merge* di una *Pull Request*, l'autore ha la responsabilità di aggiornare il proprio *branch* rispetto a *main* e risolvere manualmente ogni conflitto.

Controlli automatici (Verifiche automatiche)

Procedure di verifica eseguite in modo automatico tramite i *workflow* di *GitHub Actions* quando vengono proposte modifiche nel *repository*. Servono ad accertare che i documenti *LaTeX* compilino senza errori e che i file rispettino i requisiti tecnici stabiliti prima di consentire l'integrazione nel *branch main*.

Controllo di versione

Sistema che registra nel tempo tutte le modifiche apportate a un insieme di file, permettendo di consultare lo storico completo degli interventi, confrontare versioni differenti, lavorare in parallelo su più *branch* e ripristinare stati precedenti in caso di errore.

Credenziali

Informazioni riservate di autenticazione (quali password, chiavi di accesso o *token*) utilizzate per verificare l'identità di un utente o di un servizio. Durante la *code review* viene sempre verificata l'assenza di credenziali inserite accidentalmente nei file del *repository*.

CSS (Cascading Style Sheets)

Linguaggio di stile utilizzato insieme a *HTML* per definire l'aspetto grafico e l'impaginazione del *sito web* del gruppo (colori, caratteri tipografici, margini e disposizione degli elementi), garantendo uniformità visiva tra le diverse pagine attraverso file di stile condivisi.

D

Definition of Done (DoD)

Documento e insieme di criteri minimi di qualità, correttezza, formattazione, verifica tecnica e tracciabilità che qualsiasi attività deve soddisfare per poter essere considerata completata dal *team* e pronta per essere integrata nel *branch* principale del *repository*.

Destinatari

Campo della tabella *Informazioni documento* presente nel frontespizio dei documenti formali, compilato dall'autore per indicare i soggetti a cui il documento è destinato (il gruppo **On-By-One**, i docenti del corso prof. Tullio Vardanega e prof. Riccardo Cardin ed eventualmente l'azienda *proponente*).

Directory (Cartella)

Contenitore logico del *file system* utilizzato per organizzare in modo gerarchico e strutturato i file all'interno del *repository* (ad esempio separando i *template LaTeX*, la documentazione e i *workflow* di configurazione).

Discord

Piattaforma di comunicazione vocale (*VoIP*), video e testuale utilizzata dal gruppo **On-By-One** come uno dei canali principali per tenere le riunioni a distanza e coordinare il lavoro tra i membri.

Docs (prefisso)

Prefisso utilizzato nelle convenzioni di *branching* (**docs/**) e nei messaggi di *commit* (**docs:**) per identificare tutte le attività che riguardano la stesura, la modifica, la correzione o la rimozione di documentazione.

E

Estensione (di un file)

Sequenza di caratteri posta al termine del nome di un file dopo un punto (ad esempio `.md`, `.tex`, `.pdf`, `.html`, `.css`), che identifica il formato del file stesso. Secondo le convenzioni di denominazione del gruppo, le estensioni devono essere sempre scritte in lettere minuscole.

F

File

Unità base di memorizzazione delle informazioni digitali all'interno del *repository*, contenente testo, documentazione, codice sorgente o impostazioni tecniche.

File di configurazione

File tecnico contenente parametri, direttive e impostazioni che regolano il funzionamento degli strumenti di sviluppo, dell'ambiente del *repository* o dei *workflow* automatici di *GitHub Actions*.

Fix

Termine inglese di uso comune in ambito informatico che indica la correzione di un errore, di un difetto (*bug*) o di un'imprecisione. Nelle convenzioni per i *commit* del gruppo, l'utilizzo di messaggi generici composti dalla sola parola `fix` è vietato, poiché ogni messaggio deve descrivere in modo specifico l'oggetto della correzione.

Formattazione

Insieme delle regole tipografiche e stilistiche adottate dal gruppo per strutturare e presentare il testo nei diversi formati (*Markdown*, *LaTeX*, *HTML/CSS*), regolando l'uso di grassetto, corsivo, *codice inline*, titoli e tabelle per garantire leggibilità e coerenza tra tutti gli elaborati.

G

Git

Software libero di controllo di versione distribuito (*Distributed Version Control System*) utilizzato dal gruppo per tracciare la storia delle modifiche ai file del progetto, gestire lo sviluppo parallelo tramite *branch* e coordinare l'integrazione del lavoro di tutti i membri.

GitHub

Piattaforma web di *hosting* per progetti gestiti con *Git*. Ospita il *repository* remoto del gruppo **On-By-One** e fornisce gli strumenti collaborativi impiegati nel flusso di lavoro quotidiano, tra cui la gestione delle *issue*, le *Pull Request*, le *code review* e i *workflow* di *GitHub Actions*.

GitHub Actions

Piattaforma di automazione integrata in *GitHub* che consente di eseguire *workflow* automatici al verificarsi di specifici eventi nel *repository* (come un *push* o l'apertura di una *Pull Request*), utilizzata dal gruppo per eseguire le verifiche tecniche obbligatorie, come il controllo di compilazione dei documenti *LaTeX*.

H

HTML (HyperText Markup Language)

Linguaggio di marcatura standard utilizzato per definire la struttura e i contenuti delle pagine che compongono il *sito web* del gruppo.

I

Identificatore

Codice o nome simbolico univoco utilizzato per fare riferimento in modo non ambiguo a un elemento del progetto, come il numero di una *issue* (es. **#24**), il codice di una decisione verbalizzata (es. VI-AAAA-MM-GG.1) o un elemento tecnico nel codice.

Informazioni documento

Tabella posizionata nel frontespizio dei documenti soggetti ad approvazione che riassume i metadati relativi all'ultima versione approvata o in lavorazione: *Versione*, *Data*, *Stato* (**Bozza** o **Approvato**) e *Destinatari*.

Issue

Elemento di tracciamento fornito da *GitHub* e identificato da un numero progressivo (es. **#12**), utilizzato dal gruppo per descrivere, pianificare e assegnare un'attività da svolgere, una modifica documentale o un problema da risolvere. Costituisce il punto di partenza del flusso di lavoro standard e viene referenziata nel nome del *branch*, nei messaggi di *commit* e nella *Pull Request*.

K

Kebab Case

Convenzione di scrittura in cui i nomi sono composti esclusivamente da lettere minuscole e le singole parole sono separate dal carattere trattino - (ad esempio `code-review-policy.md` o `2026-10-28-vi`). È lo standard adottato dal gruppo per nominare file, verbali e *branch*.

Keyword

Parola chiave riservata a cui un sistema informatico attribuisce una funzione specifica, come la parola **Closes** riconosciuta da *GitHub* nelle descrizioni delle *Pull Request* per automatizzare la chiusura delle *issue* associate.

L

LaTeX

Sistema di composizione tipografica basato su linguaggio di marcatura (file con estensione `.tex`), utilizzato dal gruppo tramite appositi *template* condivisi per redigere tutta la documentazione ufficiale e i verbali destinati alla generazione in formato PDF.

Link (Collegamento ipertestuale)

Riferimento inserito all'interno di un documento (*Markdown*, *LaTeX* o *HTML*) che consente di navigare direttamente verso un altro documento del *repository*, una sezione interna o una risorsa esterna. La validità dei collegamenti è uno dei criteri verificati prima del completamento di un'attività.

M

Main

Nome del *branch principale* del *repository Git* del gruppo. Rappresenta la linea temporale ufficiale che conserva unicamente le versioni stabili, revisionate e approvate dei documenti e del codice.

Manutenibilità

Grado di facilità con cui un documento, un sito web o un file di configurazione può essere compreso, modificato, corretto o aggiornato nel tempo da qualsiasi membro del *team*, garantito dal rispetto delle convenzioni di *naming*, *formattazione* e *code review*.

Markdown

Linguaggio di marcatura leggero (file con estensione `.md`) progettato per essere facilmente leggibile sia in formato sorgente sia una volta convertito in testo formattato. È utilizzato dal gruppo per la documentazione interna del *repository*, per il file `README.md` e per le guide sul *Way of Working*.

Merge

Operazione di *Git* che conclude il ciclo di vita di una *Pull Request*, unendo le modifiche sviluppate su un *branch dedicato* all'interno del *branch di destinazione* (`main`), una volta ottenuta l'approvazione del *revisore* e superati i controlli automatici.

N

Naming (Convenzioni di denominazione)

Insieme delle regole definite dal gruppo per assegnare nomi coerenti, leggibili e standardizzati ai file, alle date (AAAA-MM-GG), ai verbali interni ed esterni e ai *branch* all'interno del *repository*.

O

On-By-One

Nome ufficiale del gruppo di lavoro impegnato nello svolgimento del progetto didattico per il corso di Ingegneria del Software presso l'Università degli Studi di Padova.

Ordine del giorno

Elenco dei temi e dei punti programmati per essere discussi durante una riunione interna o esterna del gruppo, riportato nell'apposita sezione iniziale di ciascun *verbale*.

P

PDF (Portable Document Format)

Formato di file standard utilizzato per distribuire in modo immutabile i documenti ufficiali e i verbali del gruppo, generato automaticamente tramite la compilazione dei file sorgente *LaTeX*.

Policy

Insieme di regole e direttive vincolanti adottate dal gruppo per disciplinare uno specifico processo di lavoro, come la *Policy di code review* che stabilisce i criteri di verifica e approvazione delle modifiche.

Proponente

Azienda o organizzazione esterna che presenta un capitolato d'appalto nell'ambito del corso di Ingegneria del Software e con cui il gruppo interagisce durante gli incontri documentati tramite *verbali esterni*.

Pull Request (PR)

Meccanismo offerto da *GitHub* tramite il quale l'autore richiede formalmente di integrare le modifiche effettuate su un *branch dedicato* all'interno del *branch main*. La *Pull Request* raccoglie la descrizione del lavoro svolto, il riferimento all'*issue* associata, l'esito dei controlli automatici e tutti i commenti della *code review* fino al *merge*.

Push

Operazione di *Git* che invia e pubblica i *commit* registrati sul computer locale di un membro del gruppo verso il *repository* remoto ospitato su *GitHub*. Secondo la strategia di *branching* del gruppo, i *push* sono consentiti solo sui *branch di lavoro* e mai direttamente su *main*.

R

README (README.md)

File scritto in formato *Markdown* collocato nella radice del *repository* (o di una specifica *directory*) che fornisce una panoramica immediata dei contenuti e delle istruzioni principali. Costituisce un'eccezione esplicita alla regola generale che prevede l'uso di sole lettere minuscole nei nomi dei file.

Registro delle modifiche

Tabella presente nei documenti formali del gruppo che conserva la storia cronologica di tutte le modifiche apportate al documento stesso. Per ogni intervento registra cinque campi: *Versione*, *Data*, *Autore*, *Descrizione* e *Verificatore*.

Repository

Archivio digitale centralizzato, gestito tramite il sistema di controllo di versione *Git* e ospitato su *GitHub*, in cui sono conservati tutti i file del progetto (documenti *Markdown* e *LaTeX*, verbali, sito web, *template* e *workflow*) unitamente all'intera cronologia delle modifiche.

Requisito

Obiettivo, condizione o specifica che un documento, un artefatto tecnico o un'attività deve soddisfare per rispondere a quanto richiesto dall'*issue* associata e dalla *Definition of Done*.

Responsabile

Ruolo progettuale ricoperto a rotazione da un membro del gruppo, incaricato di coordinare il *team*, pianificare le attività, aprire le relative *issue* su *GitHub* e monitorare l'avanzamento complessivo del lavoro.

Revisore

Membro del *team*, obbligatoriamente diverso dall'autore, che prende in carico la *code review* di una *Pull Request*. Ha il compito di esaminare con attenzione i contenuti proposti, segnalare eventuali osservazioni bloccanti o non bloccanti, verificare il superamento dei controlli automatici e concedere l'approvazione finale.

S

Scadenza

Termine temporale entro il quale un'*attività* (*task*) assegnata a un membro del gruppo deve essere completata e sottoposta a revisione.

Sito web

Portale web del gruppo **On-By-One**, sviluppato in *HTML* e *CSS* all'interno del *repository*, destinato a raccogliere e rendere consultabili in modo ordinato i documenti e i verbali prodotti durante il progetto.

Standard merge

Strategia di integrazione adottata dal gruppo per unire le *Pull Request* nel *branch* **main**. Prevede che tutti i singoli *commit* creati sul *branch di lavoro* vengano mantenuti integralmente nella cronologia e che venga generato un ulteriore *commit* specifico di *merge*, preservando così la tracciabilità storica completa anche dopo la rimozione del *branch* dedicato.

Stato (di un documento)

Campo della tabella *Informazioni documento* che indica la fase corrente del ciclo di vita del documento: **Bozza** (impostato dall'autore durante la stesura e la revisione) oppure **Approvato** (impostato dal *revisore* al superamento della *code review*).

T

Task

Vedi *Attività* (*Task*).

Team

L'insieme dei componenti del gruppo **On-By-One** che collaborano allo sviluppo del progetto seguendo le convenzioni e i processi condivisi nel *Way of Working*.

Template

Modello strutturato predefinito (in particolare i file *LaTeX* presenti nella *directory docs/tex/templates/* del *repository*) che fornisce l'impaginazione, il frontespizio e gli stili standardizzati da utilizzare obbligatoriamente nella redazione dei documenti per garantire uniformità visiva e strutturale.

To-do

Tabella riassuntiva posta al termine dei verbali di riunione che elenca i compiti operativi emersi durante l'incontro, specificando per ciascuno l'identificativo dell'*issue* associata, la descrizione del *task*, l'*assegnatario* e la data di scadenza.

Token

Stringa segreta di caratteri utilizzata come credenziale digitale per autenticare utenti, applicazioni o *workflow* automatici verso servizi esterni come *GitHub*, senza dover utilizzare la password personale. Non deve mai essere incluso all'interno dei file salvati nel *repository*.

Tracciabilità

Proprietà fondamentale del flusso di lavoro del gruppo che permette di risalire in modo chiaro e verificabile all'origine, all'autore, alla motivazione e alla revisione di ogni singola modifica, grazie al collegamento sistematico tra decisioni dei verbali, *issue*, *branch*, messaggi di *commit*, *Pull Request* e *Registro delle modifiche*.

V

Verbale

Documento ufficiale che riassume lo svolgimento e gli esiti di una riunione del gruppo, registrando informazioni generali (data, ora di inizio e fine, modalità, partecipanti e ruoli), *ordine del giorno*, sintesi delle discussioni, decisioni assunte con codice identificativo e tabella dei *to-do*.

Verbale esterno (ve)

Verbale redatto in seguito a una riunione svolta con la partecipazione di soggetti esterni al gruppo, come i referenti dell'azienda *proponente*. Secondo le convenzioni di denominazione, il file deve seguire il formato AAAA-MM-GG-ve (ad esempio 2026-10-28-ve.tex).

Verbale interno (vi)

Verbale redatto in seguito a una riunione che coinvolge esclusivamente i membri del gruppo **On-By-One**. Secondo le convenzioni di denominazione, il file deve seguire il formato AAAA-MM-GG-vi (ad esempio 2026-10-28-vi.tex) e le relative decisioni sono identificate dal codice [VI-AAAA-MM-GG.n].

Verifica

Insieme dei controlli manuali (*code review*) e automatici (*workflow*) eseguiti per accertare che un documento o una modifica tecnica rispetti i requisiti, non contenga errori di compilazione o di forma e soddisfi tutti i criteri della *Definition of Done*.

Verificatore

Ruolo riportato nella tabella *Registro delle modifiche* dei documenti formali e nella tabella *Informazioni documento*, ricoperto dal membro del gruppo (*revisore*) che ha esaminato e validato le modifiche introdotte dall'autore e che compila il numero di versione e lo stato di approvazione del documento.

Versionamento

Gestione strutturata dell'evoluzione nel tempo dei file e dei documenti del progetto, realizzata sia a livello tecnico tramite *Git*, sia a livello documentale tramite il sistema di numerazione delle versioni nel *Registro delle modifiche*.

Versione

Identificativo numerico a tre cifre (X.Y.Z) che contraddistingue ogni stato evolutivo di un documento soggetto ad approvazione:

- La prima modifica di un nuovo documento assume la versione iniziale 0.1.0.
- Ad ogni modifica successiva in stato di **Bozza** (o in caso di revisione non superata), viene incrementato di uno il contatore più a destra; se questo raggiunge 9, l'incremento successivo lo azzerà e incrementa di uno il contatore centrale.
- Al superamento della revisione con approvazione del documento, il *revisore* assegna la versione 1.0.0 (alla prima approvazione) oppure incrementa di uno il contatore più a sinistra azzerando gli altri due.

W

Way of Working (WoW)

L'insieme organico delle convenzioni, delle regole operative, dei ruoli e delle procedure condivise dal gruppo **On-By-One** per organizzare e svolgere il lavoro in modo coerente, disciplinato e collaborativo (comprendente convenzioni di *naming*, *formattazione*, *commit*, strategia di *branching*, *policy di code review* e *Definition of Done*).

Website (prefisso)

Prefisso utilizzato nelle convenzioni di *branching* (**website/**) e nei messaggi di *commit* (**website:**) per contrassegnare tutte le attività e le modifiche riguardanti i contenuti, la struttura *HTML* o i fogli di stile *CSS* del *sito web* del gruppo.

Workflow

Procedura automatizzata definita tramite file di configurazione nella *directory* `.github/workflows/` ed eseguita da *GitHub Actions* per compiere verifiche tecniche automatiche (come il controllo della corretta compilazione dei file *LaTeX*) ad ogni proposta di modifica nel *repository*.