



## Università degli Studi di Padova

Laurea: Informatica

Corso: Ingegneria del Software

Anno Accademico: 2025/26



## Gruppo 17

Nome: BitByBit

Email: swe.bitbybit@gmail.com

## Glossario

# Indice

<b>1</b>	<b>Glossario</b>	<b>3</b>
1.1	Acronimi . . . . .	3
<b>2</b>	<b>A</b>	<b>3</b>
<b>3</b>	<b>B</b>	<b>3</b>
<b>4</b>	<b>C</b>	<b>4</b>
<b>5</b>	<b>D</b>	<b>4</b>
<b>6</b>	<b>E</b>	<b>5</b>
<b>7</b>	<b>G</b>	<b>5</b>
<b>8</b>	<b>J</b>	<b>5</b>
<b>9</b>	<b>L</b>	<b>6</b>
<b>10</b>	<b>M</b>	<b>6</b>
<b>11</b>	<b>P</b>	<b>7</b>
<b>12</b>	<b>R</b>	<b>8</b>
<b>13</b>	<b>S</b>	<b>9</b>
<b>14</b>	<b>T</b>	<b>9</b>
<b>15</b>	<b>U</b>	<b>10</b>
<b>16</b>	<b>V</b>	<b>10</b>
<b>17</b>	<b>W</b>	<b>11</b>
<b>18</b>	<b>Redazione e revisioni del documento</b>	<b>12</b>

# 1. Glossario

## 1.1. Acronimi

<b>API</b>	Application Programming Interface.
<b>PoC</b>	Proof of Concept
<b>MVP</b>	Minimum Viable Product.
<b>RTB</b>	Requirements and Technology Baseline
<b>PB</b>	Product Baseline
<b>DB</b>	Database.
<b>JSON</b>	JavaScript Object Notation.

## Termini principali

## 2. A

### Agile development

Metodo per lo sviluppo sw ideato per produrre sw utile nel minor tempo possibile. Il sistema è sviluppato attraverso una serie di incrementi a cui partecipano anche gli stakeholders e gli utenti finali per una maggiore comunicazione in fase di sviluppo. I risultati di ogni incremento vengono utilizzati come base per la pianificazione del prossimo. Attività centrali sono: il **design** e l'**implementazione**. La stesura di un documento dei requisiti è superflua in quanto i requisiti cambiano così velocemente che il documento è obsoleto non appena viene scritto.

### API

Interfaccia che consente la comunicazione tra componenti o servizi.

### Architettura

Organizzazione generale del sistema e dei suoi componenti.

## 3. B

### Backlog

Elenco prioritario delle funzionalità, bug e attività da svolgere.

## **Baseline**

Versione approvata di un prodotto di lavoro (di progetto) che può essere modificato solo attraverso procedure formali di controllo delle modifiche.

## **Bug**

Difetto da controllare.

## **4. C**

### **Capitolato**

Documento che contiene le specifiche di un progetto software. Viene redatto dal proponente e presentato ai fornitori interessati a partecipare all'appalto. Descrive i requisiti lato bisogno (requisiti utente).

### **Committente**

Colui che paga il prodotto. Di solito (ma non necessariamente) è colui che decide i requisiti.

### **Componente**

Unità software con responsabilità ben definite e interfacce pubbliche.

### **Contratto**

Documento (legale) stilato tra committente e fornitore.

### **Click**

L'azione di premere e rilasciare rapidamente un pulsante del mouse (o un'area equivalente su un touchpad) per attivare un comando, selezionare un oggetto o seguire un collegamento ipertestuale su un'interfaccia grafica.

## **5. D**

### **Database**

Sistema di memorizzazione dei dati e la sua struttura definita.

## Deployment

Processo di pubblicazione del software in un ambiente esecutivo.

## Documento analisi dei requisiti

Stabilisce esattamente cosa deve essere implementato nel sistema, cioè elenca i requisiti lato soluzione (requisiti software). Può essere parte del contatto con l'acquirente. Può essere soggetto a modifiche durante lo sviluppo del prodotto sw.

## 6. E

### Extreme programming (XP)

Tecnica di sviluppo software. I requisiti non sono documenti formali:

- vengono espressi come scenari d'uso (chiamati **user stories**), che sono implementati come una serie di task;
- vengono scritti insieme al cliente su story card;
- vengono scomposti in compiti tecnici (task);
- per ogni task vengono sviluppati test automatici.

Affinché il nuovo codice possa essere rilasciato tutti i test devono essere eseguiti correttamente.

## 7. G

### Glossario

Elenco termini utilizzati dal gruppo.

## 8. J

### JSON

Formato per lo scambio di dati tra client e server.

## 9. L

### **Life cycle**

Gli stati che il prodotto software assume da concezione, all'uso fino al ritiro. Si riferisce alla completa durata dello sviluppo del prodotto, dal concepimento al ritiro (=fine), che deve essere garantita. È composto da stati in cui si ha una certa sequenza di passaggi da seguire. La transizione da uno stato all'altro avviene come conseguenza dell'esecuzione di un'attività. In seguito al ritiro il prodotto può essere rimesso in vita, per questo viene indicato con ciclo.

### **Licenza**

Condizioni legali che regolano l'uso e la distribuzione del software.

## 10. M

### **Milestone**

Strumento utilizzato nella gestione di progetto per fissare una data di calendario che descrive degli obiettivi che portano un avanzamento nello sviluppo di progetto. Ogni milestone deve essere:

- specifica
- raggiungibile
- misurabile
- traducibile in compiti assegnabili
- dimostrabile dagli stakeholders

### **Model**

Insieme di specifiche che descrivono un fenomeno d'interesse.

### **MVP**

Versione minima del prodotto che permette di validare l'ipotesi principale.

## 11. P

### PB

Versione approvata del prodotto pronta al rilascio o alla validazione.

### Piano di progetto

Documento ufficiale di tipo esterno, soggetto ad approvazioni, con il quale si descrivono gli obiettivi di progetto e gli elementi necessari al loro raggiungimento. Si vedono le risorse disponibili e le loro assegnazioni alle attività con scansione nel tempo. Principalmente è scomposto in:

- Pianificazione (preventiva): Team, Analisi dei rischi
- Consuntivazione: valutazione retrospettiva delle attività svolte

Struttura tipica del PdP:

- Introduzione (scopo e struttura)
- Organizzazione del progetto
- Analisi dei rischi
- Risorse disponibili
- Suddivisione del lavoro (work breakdown)
- Calendario delle attività
- Meccanismo di controllo e rendicontazione

### Piano di qualifica

Documento incrementale di tipo esterno 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 (verifica), delle metriche di valutazione e dei criteri di accettazione del prodotto finale. L'obiettivo principale è garantire che il prodotto soddisfi gli standard di qualità prefissati e che il processo di sviluppo segua procedure coerenti ed efficaci. Per ogni passo fatto in analisi o altro è necessario preparare test e tracciamenti per non lasciare indietro nulla.

### PoC

Prototipo o esperimento usato per dimostrare la fattibilità di una soluzione.

## **Process**

Un insieme strutturato di attività, metodi, pratiche e trasformazioni utilizzate per sviluppare, mantenere e gestire un prodotto software o un sistema. Definisce cosa deve essere fatto, chi lo fa, quando e come, per raggiungere un obiettivo specifico.

## **Project**

È un insieme ordinato di attività sviluppate sulla base dei processi di ciclo di vita che soddisfano gli obiettivi dati.

- Le attività sono suddivise in compiti assegnabili a singoli individui.
- Le attività sono pianificate prima di essere svolte.

## **Prototipo**

Versione preliminare per testare design o comportamenti prima dello sviluppo completo.

## **Pull Request / Merge Request**

Procedura per proporre e rivedere modifiche prima dell'integrazione.

## **12. R**

### **Repository / Repo**

Archivio centralizzato dove viene conservato e gestito il codice sorgente del progetto.

### **Repository / Branch**

Archivio del codice sorgente e linee di sviluppo parallele.

## **Requisito**

Condizione o capacità che il sistema deve soddisfare.

### **Requisito funzionale**

Funzionalità che il sistema deve offrire.

### **Requisito non funzionale**

Vincolo di qualità (prestazioni, sicurezza, usabilità, manutentibilità).



## **Requisito utente**

Requisiti astratti di alto livello. Dichiarazioni in linguaggio naturale (e anche diagrammi) dei servizi che il sistema dovrebbe fornire agli utenti e dei vincoli che deve rispettare. Possono variare da dichiarazioni generali a descrizioni dettagliate.

## **Requisito software**

Descrizioni dettagliate delle funzioni, servizi e vincoli di sistema, inseriti nel documento di analisi dei requisiti.

## **RTB**

Insieme dei requisiti approvati e della loro tracciabilità.

## **13. S**

### **Sprint**

Iterazione di lavoro con obiettivi definiti (2 settimane).

### **Stakeholder**

Tutti coloro che a vario titolo hanno influenza sul prodotto e sul progetto: utenti, committente, fornitore, regolatori.

## **14. T**

### **Task**

Attività da svolgere.

### **Technology baseline**

Documento di quello che si dovrebbe implementare.

### **Test unitario**

Verifica automatizzata di singole unità di codice.

### **Test di integrazione**

Verifica delle interazioni tra più componenti o servizi.

## Test-driven development

Vengono scritti prima i test e poi il codice. Scrivendo i test si definisce implicitamente l'interfaccia e le specifiche di comportamento di una funzionalità. I test devono essere automatici e verificare che il codice soddisfi le specifiche di output.

## To Do

Sezione di un verbale o di un documento in cui si elencano le azioni concrete che ogni membro del gruppo deve svolgere.

## 15. U

### Utente

La persona che usa e interagisce direttamente col sistema (spesso si identifica col committente).

### User story

Descrizione breve e informale di una funzionalità del sistema vista dal punto di vista dell'utente finale. Riassume un'esigenza che l'utente vuole ottenere e perché. L'utente lavora con il team per creare una **story card** che viene poi suddivisa in task.

## 16. V

### Validazione

Attività eseguita al culmine del progetto per accertare che il prodotto finale corrisponda alle attese.

### Verifica

Attività eseguita durante lo sviluppo per accertare che le attività non introducano errori.

### Verbale

Documento riassuntivo di una riunione. Deve specificare Ordine del Giorno, Discussioni, Decisioni, Issue e To Do. Ogni verbale deve essere verificato da un membro diverso da chi l'ha scritto.

## 17. W

### **Way of working (WoW)**

Comprende le metodologie di sviluppo (modello di sviluppo), la spartizione dei ruoli e delle attività, gli strumenti di lavoro (GitHub, GitLab, Trello, Jira, Slack, Discord, Obsidian, Google Docs), convenzioni standard e procedure condivise dai membri del team.

## 18. Redazione e revisioni del documento

Versione	Data	Autore	Descrizione	Verificatore
0.2.2	2025-11-05	Dennis Parolin	Aggiunti termini al documento e sistemato alcuni errori grammaticali	Gabriele Scaggianti
0.2.1	2025-11-06	Riccardo Manisi	Tolta la sezione delle convenzioni di scrittura, aggiunto indice e formattazione, inserito nuovi termini	Dennis Parolin
0.1.1	2025-11-05	Giovanni Visentin	Aggiunti termini al documento	Gabriele Scaggianti
0.1.0	2025-11-04	Giovanni Visentin	Stesura iniziale del glossario	Gabriele Scaggianti