



IMMAGINAZIONE  
E LAVORO DAL 1978

Adecco  
Formazione

# Ingegneria del Software

Proprietà e Ciclo di Vita del Software

# Proprietà del Software 1/3

Le proprietà del software sono un insieme di caratteristiche che definiscono la qualità e la funzionalità del software stesso. Esistono diverse proprietà che possono essere considerate, tra cui:

**Affidabilità:** la capacità del software di funzionare in modo coerente e senza errori per un determinato periodo di tempo.

Esempio: [https://youtu.be/YOOBh8u2\\_LY](https://youtu.be/YOOBh8u2_LY)

**Usabilità:** la facilità con cui gli utenti possono utilizzare il software per raggiungere i loro obiettivi.

**Efficienza:** la capacità del software di utilizzare efficacemente le risorse del sistema, come la CPU, la memoria e il disco rigido.

# Proprietà del Software 2/3

**Manutenibilità:** la facilità con cui il software può essere modificato o riparato dopo la sua distribuzione.

- Scrivere codice leggibile e ben strutturato
- Utilizzare nomi di variabili e metodi descrittivi
- Scrivere commenti utili e significativi
- Utilizzare costanti invece di valori letterali
- Scrivere test unitari per il codice
- Gestire le eccezioni in modo appropriato
- Usa un sistema di controllo versione come Git

## Proprietà del Software 2/2

**Portabilità:** la capacità del software di funzionare su diversi sistemi operativi o ambienti senza la necessità di modifiche significative.

**Sicurezza:** la capacità del software di proteggere i dati dell'utente e il sistema da accessi non autorizzati o attacchi esterni.

I programmatori e gli sviluppatori di software devono tenere conto di queste proprietà durante il processo di sviluppo del software per garantire che il software sia di alta qualità e funzionale.

## Sicurezza: Esempio di funzione MD5

```
import java.security.*;
```

```
..
```

```
byte[] bytesOfMessage = yourString.getBytes("UTF-8");
```

```
MessageDigest md = MessageDigest.getInstance("MD5");
```

```
byte[] theMD5digest = md.digest(bytesOfMessage);
```

# Ciclo di Vita del Software 1/2

Il ciclo di vita del software è il processo attraverso il quale il software viene sviluppato, testato, distribuito e mantenuto. Esistono diversi modelli di ciclo di vita del software, ma in generale il ciclo di vita del software segue le seguenti fasi:

**Analisi dei requisiti:** in questa fase, gli analisti raccolgono le esigenze dell'utente e le specifiche del software. Questo aiuta a definire i requisiti funzionali e non funzionali del software.

**Progettazione:** in questa fase, gli architetti software progettano il software e definiscono la struttura del software. Questo include la definizione delle interfacce utente, la definizione delle funzioni del software e la definizione dell'architettura tecnica.

**Sviluppo:** in questa fase, i programmatori scrivono il codice del software in base alle specifiche della progettazione. Questa fase comprende anche la verifica del codice e la risoluzione degli errori.

# Ciclo di Vita del Software 2/2

**Test:** in questa fase, il software viene testato per verificare che funzioni correttamente e soddisfi i requisiti dell'utente. Ci sono diversi tipi di test, tra cui test funzionali, test di integrazione e test di accettazione.

Esempio di **JUnitTest**: <https://github.com/DiegoSgammeglia/CodingDS.git>

**Distribuzione:** in questa fase, il software viene distribuito all'utente finale o al cliente.

**Manutenzione:** in questa fase, il software viene monitorato e mantenuto per garantire che funzioni correttamente e soddisfi le esigenze dell'utente. Ci sono due tipi di manutenzione: la manutenzione correttiva, che risolve gli errori del software, e la manutenzione evolutiva, che migliora il software aggiungendo nuove funzionalità.

# Modelli di ciclo di vita

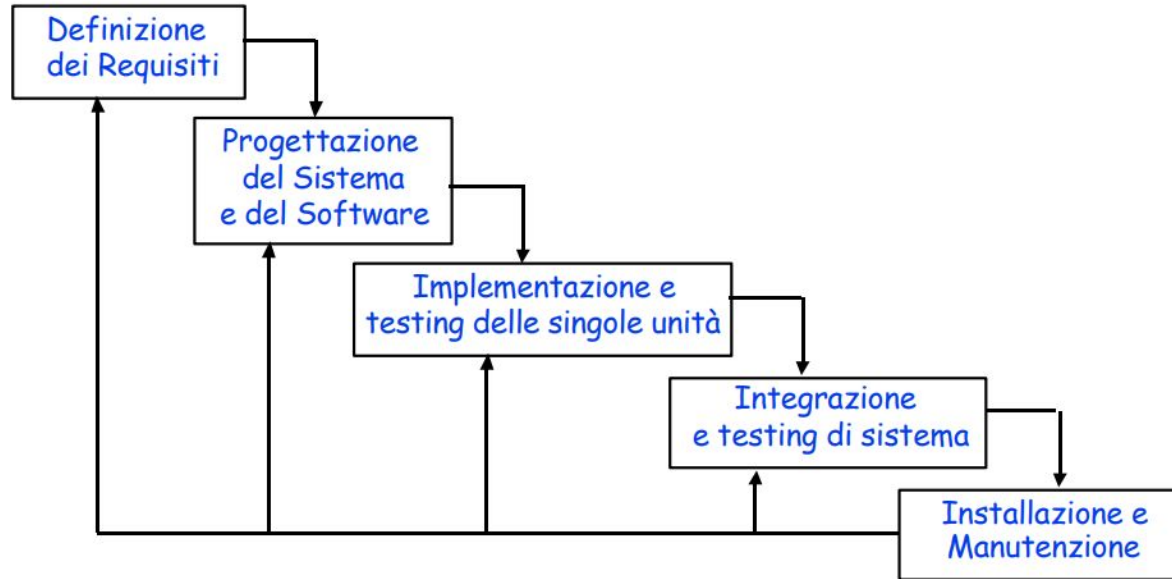
Esistono diversi modelli di ciclo di vita del software che possono essere utilizzati per lo sviluppo del software. Tra i modelli più comuni ci sono:

Modello a **cascata**: questo modello prevede che le fasi di sviluppo del software vengano eseguite in sequenza. In questo modello, ogni fase deve essere completata prima di passare alla successiva.



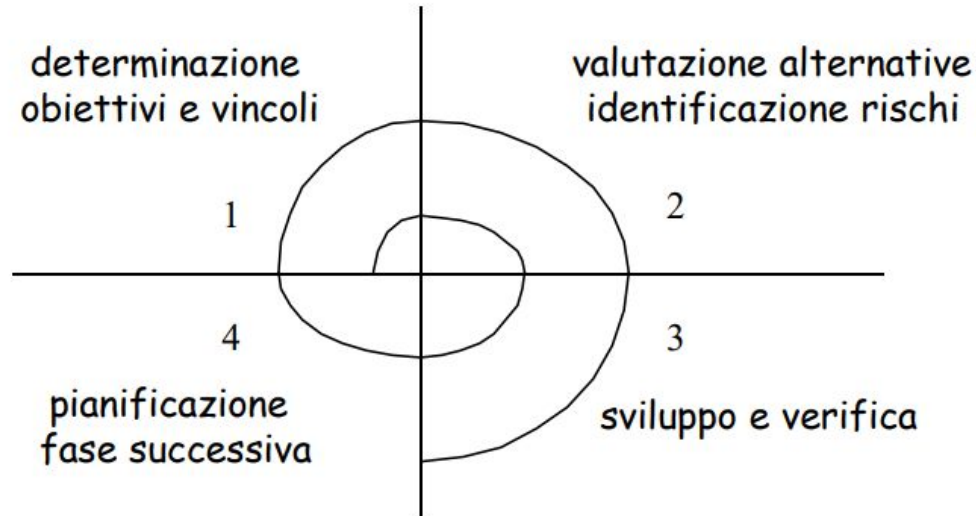
# Modello a cascata

Il ciclo di vita del software non è un processo lineare, ma piuttosto un processo iterativo in cui le fasi possono essere ripetute in base alle esigenze dell'utente.



# Modello a spirale

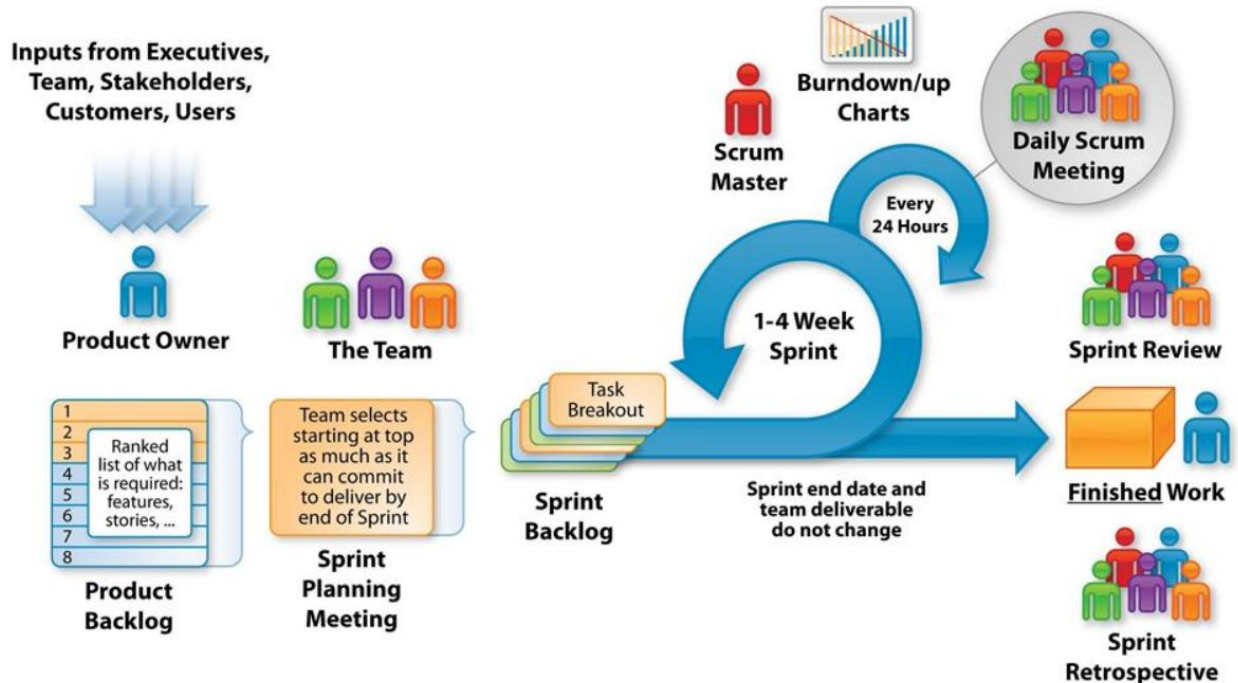
Modello a **spirale**: questo modello prevede che lo sviluppo del software avvenga in modo iterativo. In questo modello, le attività di sviluppo del software sono suddivise in cicli che si ripetono. Ogni ciclo aggiunge funzionalità al software.



# Modello Agile

Modello **agile**: questo modello prevede che lo sviluppo del software avvenga in modo collaborativo e iterativo. In questo modello, il software viene sviluppato in fasi brevi e si enfatizza la collaborazione tra gli sviluppatori e il cliente.

## The Agile - Scrum Framework



# Requisiti e Specifiche del Software

I requisiti del software sono le specifiche che definiscono cosa il software deve fare e come deve funzionare. Questi requisiti possono essere suddivisi in due categorie: requisiti **funzionali** e requisiti **non funzionali**.

I requisiti funzionali specificano le **funzionalità** che il software deve eseguire. Questi requisiti definiscono cosa il software deve fare in termini di **operazioni** specifiche, come la capacità di elaborare dati, eseguire calcoli, archiviare dati e interagire con l'utente. Questi requisiti descrivono ciò che il software deve fare e come deve funzionare.

# Requisiti non Funzionali

