



UNIVERSITÀ
POLITECNICA
DELLE MARCHE

Corso di «Project Management per l'ICT»

A.A. 2021/2022

Parte XXV: Qualità di un progetto

Prof. Domenico Ursino

d.ursino@univpm.it

- Ci sono **moltissimi modi per definire la qualità**.
- Per Scrum **la qualità si definisce come la capacità del prodotto o dei deliverable completati di soddisfare i Criteri di Accettazione e di realizzare il valore di business atteso dal cliente**.
- Per fare in modo che un progetto soddisfi i requisiti di qualità, **Scrum adotta un approccio di miglioramento continuo**, in virtù del quale il team apprende dall'esperienza e dal coinvolgimento degli stakeholder per **mantenere il Prioritized Product Backlog costantemente aggiornato** con gli eventuali cambiamenti dei requisiti.
- **Il Prioritized Product Backlog non è mai davvero completo** fino alla chiusura o alla interruzione del progetto.
- Gli **eventuali cambiamenti dei requisiti rispecchiano cambiamenti dell'ambiente di business interno o esterno** e consentono al team di lavorare e adattarsi continuamente per realizzare quei requisiti.
- Poiché per Scrum il **lavoro** deve essere **completato per incrementi durante gli Sprint**, ciò **significa** che **gli errori e i difetti vengono notati prima attraverso test di qualità ripetitivi**, invece che quando il prodotto finale o il servizio è prossimo al completamento.
- In aggiunta, **le principali attività collegate alla qualità** (ad esempio, sviluppo, test e documentazione) sono **completate all'interno del medesimo Sprint e dallo stesso team** – il che assicura che la qualità sia inerente a ciascun deliverable finito creato come parte di uno Sprint.

- Perciò, il miglioramento continuo con test ripetitivi ottimizza la probabilità di realizzare i livelli di qualità attesi in un progetto Scrum.
- Le costanti discussioni fra lo Scrum Core Team e gli stakeholder (compresi i clienti e gli utenti), unite alla consegna di effettivi incrementi di prodotto alla fine di ogni Sprint, fanno sì che il divario fra le aspettative del cliente nei confronti del progetto e i deliverable realmente prodotti si riduca costantemente.

- I requisiti di ambito e di qualità di un progetto sono stabiliti prendendo in considerazione vari fattori, quali:
 - l'esigenza di business che il progetto realizzerà;
 - la capacità e la volontà dell'organizzazione di soddisfare l'esigenza di business identificata;
 - i bisogni attuali e futuri dei destinatari.
- L'ambito di un progetto è la somma totale di tutti gli incrementi di prodotto e del lavoro necessario a sviluppare il prodotto finale.
- La qualità è la capacità dei deliverable di soddisfare i requisiti di qualità del prodotto e di appagare i bisogni del cliente.
- In Scrum, l'ambito e la qualità del progetto sono registrati nel Prioritized Product Backlog, mentre l'ambito di ciascuno Sprint è stabilito attraverso la rifinitura degli elementi estesi del Prioritized Product Backlog (PBI - Prioritized Product Backlog Items) in un insieme di piccole ma dettagliate User Story che possono essere pianificate, sviluppate e verificate all'interno di uno Sprint.
- Il Prioritized Product Backlog è continuamente messo a punto dal Product Owner.

- Il **Product Owner** fa in modo che **tutte le User Story che ci si aspetta siano eseguite dallo Scrum Team in un determinato Sprint siano rifinite prima dell'inizio di quello Sprint.**
- In generale, **i requisiti** che hanno maggior valore per la risoluzione dei problemi del cliente o per la soddisfazione dei loro bisogni sono classificati come **di alta priorità**, mentre agli altri viene assegnata una priorità più bassa.
- Le **User Story meno importanti** sono **sviluppate negli Sprint successivi** o possono **essere del tutto escluse** in base ai requisiti del cliente.
- **Durante l'esecuzione dello Sprint**, il Product Owner, il cliente e lo Scrum Team possono **discutere la lista delle funzionalità del prodotto** per rispondere ai mutevoli bisogni del cliente.

- Qualità e valore di business sono strettamente collegati fra loro.
- Pertanto è fondamentale avere chiari la qualità e l'ambito di un progetto per poter avere uno schema chiaro dei risultati e dei benefici che il progetto ed il suo prodotto devono realizzare per consegnare valore di business.
- Per stabilire il valore di business di un prodotto, è importante comprendere l'esigenza di business che è alla base dei requisiti del prodotto.
- Pertanto, l'esigenza di business stabilisce quale sia il prodotto richiesto, e, a sua volta, il prodotto fornisce il valore di business atteso.
- La qualità è una variabile complessa.
- Un'estensione dell'ambito senza un incremento dei tempi o delle risorse tende a ridurre la qualità.
- Analogamente, anche una riduzione dei tempi o delle risorse senza una restrizione dell'ambito si traduce generalmente in un calo della qualità.
- Scrum crede nella necessità di mantenere un «ritmo sostenibile» del lavoro, che aiuta con il tempo a migliorare la qualità.

- Lo **Scrum Guidance Body** può definire requisiti minimi di qualità e standard obbligatori per tutti i progetti dell'organizzazione.
- Gli standard devono essere seguiti da tutti gli Scrum Team all'interno dell'azienda.

Criteri di accettazione e Prioritized Product Backlog

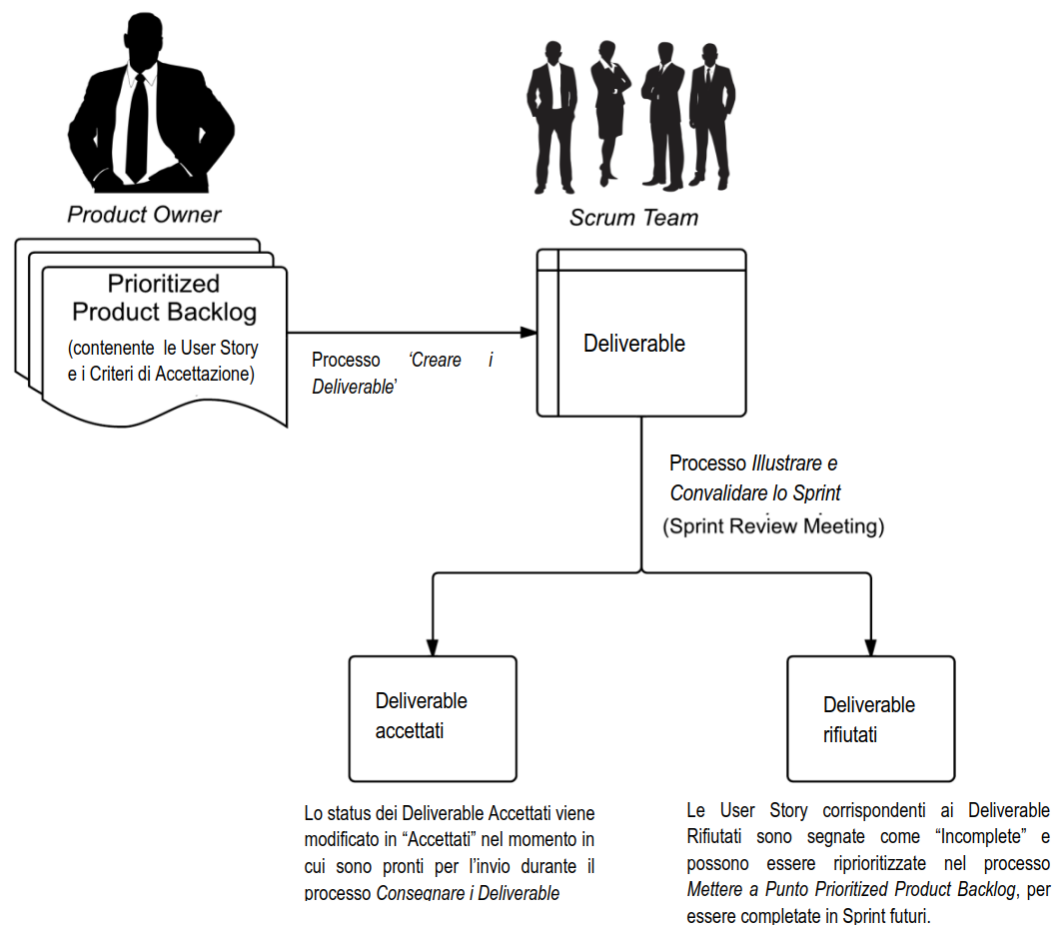
- Il **Prioritized Product Backlog** è un documento unico sui requisiti che definisce l'ambito del progetto attraverso una lista **prioritizzata delle caratteristiche del prodotto o servizio** che il progetto deve consegnare.
- Le **caratteristiche richieste** sono descritte nella forma di User Story.
- Le User Story sono **specifici requisiti delineati dai vari stakeholder** riguardo al prodotto o servizio proposto.
- **Ciascuna User Story avrà dei Criteri di Accettazione** della User Story ad essa associati (chiamati anche semplicemente «Criteri di Accettazione»), che rappresentano gli **elementi oggettivi attraverso i quali viene giudicata la funzionalità di una User Story**.
- I **Criteri di Accettazione** sono sviluppati dal Product Owner in base alla **propria comprensione esperta dei requisiti del cliente**.
- Il Product Owner comunica poi le User Story inserite nel Prioritized Product Backlog **ai membri dello Scrum Team e richiede la loro approvazione**.
- I **Criteri di Accettazione** devono **delineare in maniera esplicita le condizioni che le User Story hanno l'obbligo di soddisfare**.

Criteri di accettazione e Prioritized Product Backlog

- **Criteri di Accettazione** definiti in modo **chiaro** sono fondamentali per la consegna tempestiva ed efficace della **funzionalità individuata nelle User Story** e quindi, in definitiva, per il successo dello stesso progetto.
- Il Product Owner **utilizza questi criteri alla fine di ogni Sprint per verificare i deliverable completati**; e può accettare o rigettare singoli deliverable e le User Story ad essi associate.
- Se i deliverable sono accettati dal Product Owner, allora **la User Story si considera «Done»**, cioè fatta.
- Una **definizione chiara di ciò che può considerarsi «Done»** è essenziale in quanto aiuta a chiarire i requisiti e permette al team di conformarsi alle norme di qualità.
- Essa, inoltre, **aiuta il team a pensare secondo il punto di vista dell'utente** quando lavora alle User Story.
- Le **User Story corrispondenti a deliverable rifiutati sono reinserite nel Prioritized Product Backlog Aggiornato** durante il processo *Mettere a Punto il Prioritized Product Backlog*, per essere poi completate in futuri Sprint.
- Il rifiuto di **pochi, singoli deliverable** e delle corrispondenti User Story **non comporta il rifiuto del prodotto finale o dell'incremento di prodotto**. Il prodotto o incremento di prodotto potrebbe essere potenzialmente disponibile per la consegna anche nel caso in cui sia rifiutata qualche User Story.

Criteri di accettazione e Prioritized Product Backlog

- La seguente figura illustra il concetto di Criteri di Accettazione coniugato al flusso dell'incremento di prodotto:



- I **Criteri di Accettazione** sono propri e **specifici di ciascuna User Story** e **non sono il sostituto di una lista dei requisiti**.
- **Esempio:**
 - **Persona:** Janine è una libera professionista di 36 anni, sposata con tre figli. È una donna impegnata e di successo, che concilia la sua vita professionale con quella personale. Ha dimestichezza con la tecnologia ed è costantemente fra i primi ad utilizzare servizi e prodotti innovativi. È sempre connessa ad internet con diversi dispositivi e fa regolarmente acquisti dai portali di e-commerce.
 - **User Story:** «Io Janine, nella mia qualità di acquirente di prodotti alimentari online, devo poter salvare e vedere la mia bozza di ordine da qualsiasi dispositivo, in modo da poter completare il processo di acquisto quando mi è più comodo.»
 - **Criteri di Accettazione:**
 - tutti gli ordini in corso devono essere **salvati come bozza di ordine ogni 5 secondi nell'account dell'utente connesso**;
 - **le nuove bozze di ordine devono arrivare come notifiche su qualunque dispositivo** al quale l'utente si connetta.
- Per un Product Owner è importante notare che **le User Story che soddisfano la maggior parte dei Criteri di Accettazione, ma comunque non tutti, non possono essere accettate come «Done»**.

- I progetti Scrum operano in Sprint di durata predeterminata (Time-boxed), con uno Sprint Backlog dedicato per ciascuno Sprint.
- Spesso, l'ultimo pezzetto di lavoro può essere la parte più complicata di una User Story e può richiedere più tempo di quanto ci si aspettasse.
- Se a User Story incomplete si riconoscesse la qualifica di parzialmente «Done» e quindi le si prosegue nello Sprint successivo, l'avanzamento dello Sprint a seguire si potrebbe interrompere.
- Per questo motivo, lo status di «Done» è o bianco o nero. Una User Story può essere solo «Done» o «not Done».

- Un'unità di business di più alto livello può rendere pubblici dei Criteri di Accettazione Minimi obbligatori, che entreranno poi a far parte dei Criteri di Accettazione di tutte le User Story di quella unità di business.
- Qualsiasi funzionalità definita dall'unità di business deve soddisfare questi Criteri di Accettazione Minimi per essere accettata dal rispettivo Product Owner.
- L'introduzione di questi Criteri di Accettazione può portare ad una serie di Criteri di Accettazione a cascata per il portfolio, per il programma e per il progetto, come mostrato nella seguente figura:

Portfolio Product Owner	<ul style="list-style-type: none">• Fissa i Criteri di Accettazione minimi dell'intero portfolio• Rivede i deliverable del portfolio
Program Product Owner	<p>Fissa i Criteri di Accettazione minimi dell'intero programma, che includono i Criteri di Accettazione del portfolio</p> <ul style="list-style-type: none">• Rivede i deliverable del programma
Product Owner	<ul style="list-style-type: none">• Fissa i Criteri di Accettazione minimi del progetto, che includono i Criteri di Accettazione del programma• Rivede i deliverable del progetto

- Gli standard di qualità generali, le linee guida e i modelli di un intero portfolio sono fissati dal Portfolio Product Owner.
- I Criteri di Accettazione minimi per i programmi sono fissati dal Program Product Owner.

- Pertanto, i Criteri di Accettazione di una User Story che fa parte di un progetto includeranno implicitamente tutti i Criteri di Accettazione minimi dei livelli superiori, ove applicabili.
- Una volta definiti, tali Criteri di Accettazione minimi possono poi essere documentati nei documenti dello Scrum Guidance Body e utilizzati come riferimento dagli Scrum Team ove necessario.

- Esiste una differenza fondamentale fra i «Done Criteria» e i «Criteri di Accettazione».
- Mentre i Criteri di Accettazione sono propri e specifici di una singola User Story, i Done Criteria sono un complesso di regole applicabili a tutte le User Story di un dato Sprint.
- I Done Criteria generali possono includere una qualunque delle seguenti opzioni:
 - revisione da parte degli altri membri del team;
 - è stato completato il testing di unità della User Story;
 - completamento dei test di garanzia della qualità;
 - completamento di tutta la documentazione relativa alla User Story;
 - tutte le questioni sono state risolte;
 - dimostrazione con esito positivo agli stakeholder e/o ai rappresentanti del business.
- Come per i Criteri di Accettazione, perché la User Story possa essere considerata «Done», cioè fatta, devono essere soddisfatte tutte le condizioni dei Done Criteria.

- Lo Scrum Team dovrebbe **utilizzare una lista di controllo dei Done Criteria generali** per accertarsi con sicurezza che un'attività sia finita e che il risultato soddisfi la Definizione di «Done» (Definition of Done - DoD).
- Una **Definizione di «Done» chiara è di importanza fondamentale**, perché aiuta a rimuovere le ambiguità e permette al team di conformarsi alle norme di qualità prescritte.
- La **definizione di «Done» è solitamente stabilita e documentata dallo Scrum Guidance Body**.
- **Le registrazioni e i dati richiesti** per il rispetto dei requisiti di documentazione del progetto possono essere **generati man mano che il team va avanti con gli Sprint e i Rilasci**.
- L'inserimento di attività come la **conduzione di riunioni di revisione** e la **stesura di documenti di progettazione** può **contribuire ad assicurare il rispetto degli standard di qualità interni ed esterni**.
- **I principi di base di Scrum** - come le iterazioni brevi, la costruzione incrementale, il coinvolgimento del cliente, l'adattamento al cambiamento dei requisiti e, per finire, il costante aggiustamento dell'ambito, dei tempi e dei costi all'interno del progetto - **saranno applicati in ogni caso**.

Accettazione o rifiuto degli elementi del Prioritized Product Backlog

- Verso la fine di un'iterazione, le rispettive unità di business e gli stakeholder partecipano ad uno **Sprint Review Meeting** nel quale l'incremento di prodotto viene illustrato al Product Owner, allo sponsor, al cliente e agli utenti.
- Sebbene venga raccolto il feedback da tutti gli stakeholder, solo il **Product Owner** ha il potere di accettare o rifiutare una particolare User Story come «Done», in base ai Criteri di Accettazione concordati.
- Perciò **il ruolo dei Criteri di Accettazione nel mantenimento della qualità è fondamentale** e deve essere compreso chiaramente dal team.
- Spetta **allo Scrum Master** fare in modo che il Product Owner non modifichi i Criteri di Accettazione di una User Story nel mezzo di uno Sprint.
- Le **User Story parzialmente completate** sono **rifiutate come «not Done»** e reinserite nel Prioritized Product Backlog.

- Il cliente è lo stakeholder più importante di qualsiasi progetto. Per questo è importante capire i bisogni e di requisiti del cliente.
- La Voce del Cliente (Voice of the Customer - VOC) può essere definita come i requisiti impliciti ed espliciti del cliente, che devono essere compresi prima di progettare un prodotto o servizio.
- Generalmente, in un ambiente Scrum, l'attenzione del Product Owner è concentrata sui requisiti di business e sugli obiettivi, che insieme rappresentano la Voce del Cliente.
- Il Product Owner può trarre un grande beneficio dagli orientamenti messi a disposizione dallo Scrum Guidance Body (con i documenti o gli standard sulla qualità, o tramite gli esperti della qualità).
- Questi specialisti dovrebbero lavorare con il Product Owner e il cliente per assicurare alle User Story l'adeguato livello di dettaglio e di informazioni, dal momento che le User Story rappresentano la base del successo di qualsiasi progetto Scrum.
- Da notare che gli stakeholder esterni non sono direttamente coinvolti a livello di Scrum Team, mentre interagiscono soprattutto con il Product Owner.

- Per qualsiasi progetto Scrum, **il cliente può essere indifferentemente**:
 - **Interno**, cioè appartenente alla stessa organizzazione;
 - **Esterno**, cioè al di fuori dell'organizzazione.
- In Scrum **la gestione della qualità consente ai clienti di prendere coscienza dei problemi del progetto fin dal suo inizio** e li aiuta a rendersi conto se un progetto sta funzionando o meno.
- In Scrum **la qualità riguarda la soddisfazione del cliente e un prodotto funzionante, non necessariamente la rispondenza a metriche arbitrarie.**
- Questa **distinzione assume molta importanza dal punto di vista dei clienti**, dal momento che sono loro che investono tempo e denaro nel progetto.
- La gestione della qualità in Scrum è facilitata da **tre attività fra loro correlate**:
 - **Pianificazione** della qualità;
 - **Controllo** della qualità;
 - **Garanzia** della qualità.

La gestione della qualità in Scrum – Pianificazione della qualità

- Uno dei principi guida di Scrum è quello di **sviluppare per prima la funzionalità che ha la priorità più alta** per il cliente.
- **Le funzionalità meno importanti** sono **sviluppate negli Sprint successivi** o possono essere **escluse** del tutto in base ai requisiti del cliente.
- Questo approccio **dà modo allo Scrum Team di avere tempo per concentrarsi sulla qualità della funzionalità essenziale.**
- Un **beneficio fondamentale** della pianificazione della qualità è la **riduzione del technical debt.**
- Il **technical debt** (debito tecnico)—chiamato anche debito di progettazione (design debt) o debito del codice (code debt)—si riferisce **al lavoro a cui il team dà priorità più bassa**, oppure omette o non completa, **in quanto lavora alla creazione dei deliverable principali** associati al prodotto del progetto.
- Il technical debt si **accumula e si paga nel futuro.**
- Di seguito sono elencate **alcune delle cause del technical debt:**
 - **Soluzioni tampone** e costruzione di **deliverable che non si conformano** agli standard di qualità, sicurezza, obiettivi architetturali di lungo termine, ecc.
 - **Attività di testing inadeguata** o incompleta;

La gestione della qualità in Scrum – Pianificazione della qualità

- Documentazione inappropriata o incompleta.
- Mancanza di coordinamento fra i vari membri del team, o situazioni in cui diversi Scrum Team iniziano a lavorare in modo isolato, con conseguente scarsa concentrazione sulla integrazione finale dei vari componenti, necessaria a garantire il successo del progetto o programma.
- Scarsa condivisione della conoscenza di business e di processo fra gli stakeholder e i team di progetto
- Eccessiva focalizzazione sugli obiettivi di breve termine del progetto invece che sugli obiettivi aziendali di lungo termine. Questa disattenzione può dar luogo a Deliverable Funzionanti di bassa qualità, che provocano significativi costi di manutenzione e di aggiornamento.
- Nei progetti Scrum, grazie a Criteri di Accettazione e Done Criteria chiaramente definiti, non viene trasferito alcun technical debt da uno Sprint al successivo.
- La funzionalità deve soddisfare questi criteri per essere considerata «Done».
- Dal momento che il Prioritized Product Backlog viene costantemente messo a punto e le User Story vengono prioritzate, il team crea regolarmente Deliverable Funzionanti, prevenendo così l'accumulazione di un significativo technical debt.

La gestione della qualità in Scrum – Pianificazione della qualità

- Inoltre, lo **Scrum Guidance Body** può includere documentazione e definizioni dei processi che contribuiscono a ridurre il technical debt.
- Per **mantenere una quantità minima di technical debt**, è importante **definire il prodotto** che si richiede a uno Sprint e all'intero progetto **insieme ai Criteri di Accettazione**, agli eventuali metodi di sviluppo da seguire, e alle responsabilità **chiave** attribuite ai membri dello Scrum Team **in materia di qualità**.
- La **definizione dei Criteri di Accettazione** è una parte importante della pianificazione della qualità e consente di mettere in atto un controllo di qualità efficace durante il progetto.
- Il **technical debt** è una sfida molto impegnativa per alcune tecniche di project management tradizionale, nelle quali lo sviluppo, il testing, la documentazione, ecc. sono **eseguite in sequenza** e molto spesso da **persone diverse**, senza che **nessuno sia responsabile per un particolare Deliverable** Funzionante.
- Il risultato è che **il technical debt si accumula**, con conseguenti **costi di mantenimento, di integrazione e di rilascio del prodotto** significativamente più alti nelle fasi finali della consegna di un progetto.
- Inoltre, in questi casi **il costo delle modifiche è molto alto** poiché **i problemi emergono solo nelle ultime fasi del progetto**.

La gestione della qualità in Scrum – Pianificazione della qualità

- Il framework **Scrum** previene le questioni collegate al **technical debt** assicurando che i **deliverable** classificabili come «**Done**» con i relativi Criteri di Accettazione siano **definiti come parte dello Sprint Backlog** e che anche le attività fondamentali fra cui lo sviluppo, il testing e la documentazione siano eseguite nell'ambito del medesimo Sprint e dallo stesso Scrum Team.
- Il **mantenimento di un ritmo sostenibile** è uno dei principali cardini di Scrum.
- Il ritmo sostenibile si traduce **nell'aumento della soddisfazione del personale**, nella stabilità e in una **maggiore accuratezza delle stime**, fattori che portano tutti in definitiva ad una **maggiore soddisfazione del cliente**.
- Per sviluppare un prodotto veramente di alta qualità e mantenere un ambiente di lavoro sano, è importante **eseguire regolarmente attività di carattere integrativo** piuttosto che rinviare fino alla fine il lavoro di integrazione.
- Per fornire valore a intervalli frequenti, **il team deve sviluppare, testare e integrare continuamente le funzionalità di ciascun elemento del Prioritized Product Backlog (PBI) in ogni Sprint**, facendo uso di tecniche come l'integrazione continua e i test automatizzati del prodotto.
- È inoltre **importante**, dal punto di vista del team, fare in modo **che l'impegno profuso nello Sprint corrente sia analogo a quello dello Sprint precedente**, al fine di mantenere un **ritmo costante** per tutti gli Sprint del progetto.



La gestione della qualità in Scrum – Pianificazione della qualità

- Questo aiuta il team ad evitare fasi di intensi periodi di lavoro, mettendolo in condizione di riuscire sempre a approfondire il livello di sforzo necessario a compiere il lavoro che deve essere svolto.

La gestione della qualità in Scrum – Controllo di Qualità e Garanzia di Qualità

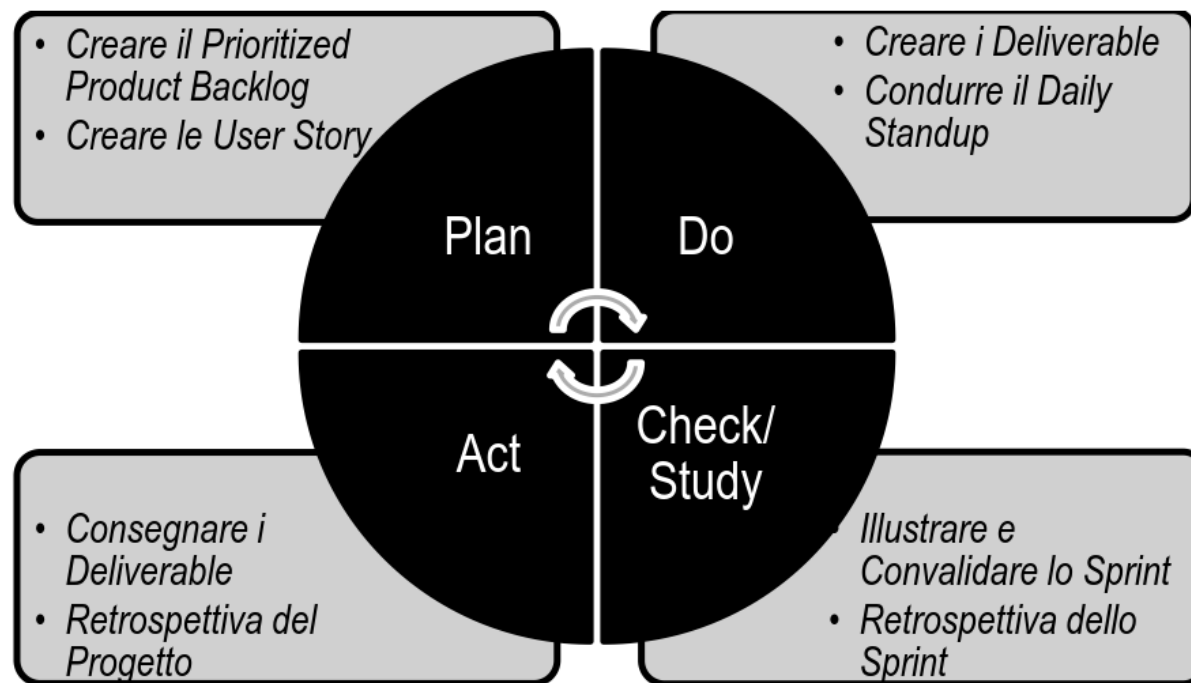
- Il **controllo di qualità** si riferisce alla **esecuzione delle attività pianificate relative alla qualità** da parte dello Scrum Team **nel processo di creazione di deliverable** che sono potenzialmente consegnabili.
- Tale controllo **include anche l'apprendimento da ciascun insieme di attività completate** al fine di realizzare il **miglioramento continuo**.
- È **importante possedere**, all'interno di un team interfunzionale, **le capacità necessarie ad eseguire le attività di controllo di qualità**.
- Durante lo **Sprint Retrospect Meeting**, i membri del team **discutono le lezioni apprese**.
- Queste lezioni fungono da **input per il miglioramento continuo** e contribuiscono al **miglioramento del controllo di qualità sempre in corso**.
- La **qualità** è richiesta **non solo nei prodotti ma anche nei processi**.
- La **garanzia di qualità** si riferisce alla **valutazione dei processi e degli standard che governano la gestione della qualità in un progetto per confermare che continuano ad essere pertinenti**.
- Le attività di **garanzia della qualità** sono eseguite come parte del lavoro di progetto.

La gestione della qualità in Scrum – Controllo di Qualità e Garanzia di Qualità

- In effetti, la **garanzia di qualità** è un fattore **importante** della definizione di «Done».
- Il **deliverable non è completato** se non è stata eseguita l'opportuna garanzia di qualità.
- **Spesso**, la garanzia di qualità è **dimostrata durante lo Sprint Review Meeting**.
- I **Product Owner** dei rispettivi progetti, programmi e portfolio possono **monitorare e valutare le attività di garanzia di qualità** per assicurarsi che ogni team continui ad essere in accordo e a conformarsi agli standard di qualità fissati.
- Una **garanzia di qualità completa** può **riguardare il test finale di un prodotto, un Rilascio o uno Sprint**.
- Si può eseguire un **confronto fra il numero di questioni emerse e il numero di User Story completate**.
- I **componenti del prodotto che presentano difetti** possono essere **incorporati come elementi del Prioritized Product Backlog (PBI)**, ai quali possono lavorare l'intero team o una singola persona in determinati momenti dello Sprint, a seconda del numero dei difetti.
- A volte, lo **Scrum Guidance Body** può **definire i processi e i documenti a cui gli Scrum Team devono fare riferimento nell'eseguire i loro progetti**, per garantire che **tutti i progetti dell'azienda seguano le medesime norme di qualità**.

- Il **Ciclo Plan-Do-Check-Act** — noto anche come **Ciclo di Deming o di Shewhart** — fu sviluppato dal Dott. W. Edwards Deming, considerato il padre del moderno controllo di qualità e dal Dott. Walter A. Shewhart.
- Quelli che seguono sono **alcuni punti importanti della filosofia di Deming**:
 - Le **linee guida del management** definiscono la qualità.
 - Quando il management è in grado di **mettere a disposizione un ambiente favorevole** ed è capace di **motivare i propri dipendenti** per migliorare la qualità su base continua, **ciascun dipendente sarà in grado di dare il proprio fattivo contributo** alla realizzazione di un prodotto di qualità superiore.
 - La «**Teoria della Conoscenza Approfondita**» di Deming **raccomanda ciò che il management dovrebbe fare** per creare un ambiente nel quale ogni dipendente può contribuire in modo significativo al miglioramento della qualità.
- Deming **modificò successivamente il ciclo Plan-Do-Check-Act in Plan-Do-Study-Act (PDSA)**, perché ritenne che il **termine «Study» – Esamina – mettesse in luce il concetto di analisi piuttosto che semplicemente quello di ispezione**, come suggerito dal termine «Check» – Controlla.
- Sia **Scrum** che il **Ciclo PDCA** di Deming/Shewhart sono **metodi iterativi che si focalizzano sul miglioramento continuo**.

- La seguente figura illustra le fasi del Ciclo PDCA e la loro correlazione con i vari processi di Scrum.



Confronto fra Scrum e il Project Management Tradizionale

- Anche se ci sono delle **analogie fra Scrum e i metodi tradizionali di project management riguardo alla definizione di «qualità»** (vale a dire, la capacità del prodotto di soddisfare i Criteri di Accettazione concordati e di realizzare il valore di business atteso dal cliente), ci sono delle **differenze riguardo a come i due approcci affrontano l'implementazione e la realizzazione dei livelli di qualità richiesti.**
- Nei **metodi tradizionali di project management**, gli utenti chiariscono le proprie aspettative; il project manager **definisce quelle aspettative in termini misurabili e ottiene l'approvazione da parte degli utenti.**
- Dopo una pianificazione dettagliata, **il team di progetto sviluppa il prodotto in un periodo di tempo concordato.**
- Se deve essere modificato qualcuno dei criteri concordati, **le modifiche possono avvenire solo attraverso un sistema formale di gestione dei cambiamenti** nel quale viene **stimato l'impatto delle modifiche** e il Project Manager ottiene **l'approvazione da parte di tutti gli stakeholder pertinenti.**
- In **Scrum**, tuttavia, **il Product Owner collabora con lo Scrum Team e definisce i Criteri di Accettazione delle User Story relative al prodotto da consegnare.**
- Lo **Scrum Team poi sviluppa il prodotto in una serie di brevi iterazioni chiamate Sprint.**

Confronto fra Scrum e il Project Management Tradizionale

- Il Product Owner può operare cambiamenti ai requisiti per stare al passo con i bisogni dell'utente e questi cambiamenti possono essere presi in esame dallo Scrum Team terminando anticipatamente lo Sprint corrente oppure includendo i requisiti modificati nello Sprint successivo, dal momento che ogni Sprint è di durata molto breve (vale a dire, da una a sei settimane).
- Uno dei maggiori vantaggi di Scrum è il risalto dato alla creazione di deliverable potenzialmente consegnabili alla fine del ciclo di ogni Sprint, invece che alla fine dell'intero progetto.
- Quindi, il Product Owner e i clienti ispezionano, approvano e accettano costantemente i Deliverable dopo ogni Sprint.
- Inoltre, anche se un progetto Scrum viene terminato anticipatamente, prima dell'interruzione si è comunque generato del valore, che è costituito appunto dai Deliverable creati nei singoli Sprint.