

Expanded Guidance – Versione in italiano

User Story

Le User Story sono state originariamente sviluppate come mezzo per descrivere i requisiti di un prodotto dal punto di vista dell'utente, da qui il nome "User Story". La tecnica è stata resa popolare da Mike Cohn nel suo libro *"User Stories Applied"* ed è ora uno dei modi più popolari per esprimere gli elementi del Product Backlog in tutto il mondo.

Formato

Le User Story hanno un formato specifico:

Come <utente/beneficiario>

Voglio <questa caratteristica/capacità>

In modo che <possa raggiungere questo obiettivo>

Questo formato si basa sulla semplice affermazione di ciò che è necessario (il "Voglio...") fornendo contesto e motivazione per il bisogno dell'utente, che si esprime nelle affermazioni "Come un..." e "In modo che...", le quali aiutano a modellare la comprensione del bisogno e a dare forma a una soluzione per soddisfare tale bisogno.

La Scrum Guide afferma che *"Un prodotto è un mezzo per fornire valore. Ha un confine chiaro, parti interessate conosciute, utenti o clienti ben definiti. Un prodotto potrebbe essere un servizio, un prodotto fisico o qualcosa di più astratto"*. Afferma anche che *"Il Product Goal descrive uno stato futuro del prodotto che può servire come obiettivo per lo Scrum Team rispetto al quale pianificare"*.

Il formato di una User Story fornisce una descrizione chiara di un obiettivo o di un'esigenza all'interno di un confine (nell'affermazione "Voglio..."), utenti o clienti ben definiti (nell'affermazione "Come...") e un focus sulla fornitura di valore (nell'affermazione "In modo che..."). Unite, tali affermazioni permettono allo Scrum Team di comprendere la necessità per l'utente di avere quel prodotto e mostrano il percorso da compiere partendo dall'elemento del Product Backlog fino a giungere al prezioso Incremento (Increment).

User Story e Product Backlog

Gli elementi del Product Backlog possono essere espressi come storie. Ne è un esempio quello che segue:

Come *commesso viaggiatore*

Voglio *un autoveicolo per uso personale*

In modo che *possa trasportare me stesso e una varietà di illustrazioni e campioni da un potenziale cliente a un altro in un dato giorno*

Attraverso obiettivi più specifici associati a sottoinsiemi del prodotto generale, si può scendere più nel dettaglio:

Come *commesso viaggiatore*

Voglio *un sistema di navigazione satellitare nel mio veicolo a motore*

In modo che *possa pianificare un percorso che mi permetta di massimizzare il numero di clienti che posso visitare in un solo giorno*

Fino a uno specifico obiettivo ancora più dettagliato, come ad esempio:

Come *commesso viaggiatore*

Expanded Guidance – Versione in italiano

Voglio informazioni di viaggio in tempo reale disponibili per il sistema di navigazione satellitare del mio veicolo

In modo che possa pianificare il percorso più efficiente in termini di tempo da una sede del cliente all'altra e massimizzare il numero di clienti che posso visitare in un solo giorno

In Scrum, le storie nel Product Backlog vengono suddivise in storie sempre più dettagliate attraverso il processo di *backlog refinement*, fino a quando non saranno abbastanza piccole per poter essere realizzate all'interno di un singolo Sprint.

A storie di diversi livelli vengono spesso assegnate etichette diverse, come quelle mostrate nel diagramma sottostante. In termini Scrum, qualunque sia la loro etichetta, quelli rappresentati sono comunque Obiettivi di Prodotto e Elementi del Product Backlog, ma espressi con diversi livelli di granularità.



Composizione di una User Story

Un elemento del Backlog (Backlog Item) espresso in modo chiaro e conciso ed abbastanza piccolo da essere svolto entro i limiti di un singolo Sprint, è considerato **pronto** per essere realizzato in quello Sprint.

Scrum richiede che almeno un Incremento venga consegnato in ogni Sprint; ogni Incremento deve soddisfare la sua Definition of Done. Esprimere gli elementi del Product Backlog come User Story è un'attività che si allinea con i processi associati con la creazione, il perfezionamento e l'ordinamento del Backlog, ma per funzionare in modo ottimale è necessario che lo Scrum Team vada oltre la semplicità della struttura vista prima "Come... Voglio... In modo che".

Quando è stata creata la tecnica delle User Story, le storie erano considerate composte da tre elementi, noti come le **3 C**.

- **Card** su cui è scritta la storia
- **Conversazione** (Conversation) necessaria per raggiungere una comprensione collettiva della storia
- **Conferma** (Confirmation), fornisce una definizione di "Fatto" (Done) per la storia

Più recentemente alcuni praticanti hanno ritenuto utile considerare una quarta C:

- **Contesto** (Context) all'interno del quale esiste la storia, o l'Incremento che diventerà

Expanded Guidance – Versione in italiano

Card

Le User Story devono essere descritte su un foglio di dimensioni 15 cm per 10 cm. Limitare lo spazio disponibile limita intenzionalmente il livello di dettaglio del bisogno che si sta analizzando, assicurando che sia espresso in modo conciso. Inoltre, il foglio funge da "token per una conversazione" e viene spesso utilizzato come token nelle sessioni di pianificazione per aiutare a creare un ordine ragionevole per il Product Backlog.

Conversazione (Conversation)

Il Product Backlog è un *emergente* e *ordinato* elenco di ciò che è necessario per migliorare il prodotto. L'affinamento del Product Backlog (Backlog refinement) è l'attività che permette di far emergere dettagli, di definire l'ordine in cui gli elementi del Product Backlog dovrebbero essere affrontati e garantisce che tali elementi siano pronti per essere accettati nello Sprint successivo. È un'attività ricorrente e collaborativa che coinvolge l'intero Scrum Team sia per quanto riguarda l'ordinamento, sia per quanto riguarda la comprensione dei singoli elementi emersi durante la *conversazione*. Ha senso rimandare la discussione dei dettagli degli elementi del Product Backlog all'ultimo momento possibile, cioè prima dell'inizio del relativo Sprint, ciò per garantire che le decisioni siano basate su informazioni quanto più possibile aggiornate.

Conferma (Confirmation)

Quando un elemento del Product Backlog si dice *pronto* per essere realizzato in uno Sprint, la sua Definition of Done deve essere chiara. Ricordiamo che l'elemento di *Conferma* della User Story è la sua Definition of Done. Con la Definition of Done, si pongono domande con le quali il Product Owner, rispondendo, può "*confermare*" che la storia è "*Done*" nella misura in cui l'utente/beneficiario (*Come...*) potrebbe utilizzare la caratteristica/capacità dell'Incremento (*Voglio...*) per raggiungere il proprio obiettivo (*In modo che...*)

Contesto (Context)

Fornire un contesto per una storia significa descrivere in che modo quella User Story si relaziona ad altre storie e può rivelarsi utile per sviluppare una maggiore comprensione dell'elemento del Product Backlog e, quindi, dell'ordine in cui gli elementi dovrebbero essere affrontati. Può anche aiutare a comprendere come una serie di storie correlate possano aiutare a formare e/o realizzare uno Sprint Goal. Varie tecniche di modellazione possono essere impiegate nel processo di identificazione e perfezionamento delle storie e per aiutare a garantire la coerenza del prodotto stesso, mentre gli Incrementi vengono forniti "come trampolini di lancio concreti verso l'obiettivo del prodotto". In ambienti regolamentati tali modelli possono essere formalmente costituiti e controllati, in ambienti meno formali possono semplicemente riflettere il raggruppamento delle Story card su un tavolo o su una parete. Un esempio spesso utilizzato è la tecnica dello Story Mapping di Jeff Patton.

Collegare il Lean thinking con le User Story, i Backlog, gli Sprint e gli Incrementi

Il Lean thinking incoraggia a *ridurre gli sprechi e concentrarsi sull'essenziale*, nonché sul migliorare continuamente i processi. Il Lean Thinking incoraggia a concentrarsi su:

Valore – ad esempio identificando specificamente il valore negli obiettivi e negli elementi da affrontare nel Product e nello Sprint Backlog; **Flusso** – ad esempio assicurando che le fasi di

Expanded Guidance – Versione in italiano

creazione del valore avvengano in stretta sequenza (dal Backlog item all'Incremento consegnato); **Tirare** – ad esempio attraverso l'estrazione degli elementi di maggior valore dal Backlog, giusto in tempo per essere convertiti in prodotti di valore.

Un'interpretazione comune che si ritrova nelle versioni precedenti della Scrum Guide è che un Incremento (spesso indicato come un *“incremento del prodotto potenzialmente rilasciabile”*) è qualcosa consegnato alla fine di uno Sprint, dopo l'ispezione avvenuta nello Sprint Review, necessaria per verificare che abbia soddisfatto la sua Definition of Done.

Sebbene Scrum supporti ancora questa modalità di lavoro, e molti potrebbero scegliere di applicarla, in realtà non abbraccia completamente tutti i concetti tipici del Lean Thinking.

I pilastri empirici della trasparenza, dell'ispezione e dell'adattamento sono incorporati nel Lean Thinking e nella teoria di Scrum nel modo che segue:

- estrarre il lavoro dal Backlog – sia nel caso in cui sia l'intero Scrum Team che estrae gli elementi del Product Backlog per lo Sprint, sia nel caso in cui siano i singoli sviluppatori ad estrarre gli elementi su cui lavorare dallo Sprint Backlog
- incoraggiare il valore attraverso la costruzione della qualità – Scrum fa questo assicurandosi che lo Scrum Team abbia tutte le capacità necessarie per raggiungere il *“Done”*

Altre caratteristiche del Lean thinking riguardano il *“flusso”*. Scrum prevede che il Team sia focalizzato sullo sviluppo di un singolo prodotto e sul flusso di lavoro. L'idea che un Incremento nasca non appena viene completato un elemento del Backlog, affronta chiaramente e in maniera risolutiva il problema dello *“spreco”* nel sistema, che deriva dal ritardare la consegna del valore necessariamente alla fine di uno Sprint.

La guida di Scrum afferma chiaramente che *“un Incremento può essere consegnato agli stakeholder prima della fine dello Sprint”* e che *“lo Sprint Review non dovrebbe mai essere considerato un passaggio obbligato per rilasciare valore”*. Ciò separa esplicitamente la consegna dell'Incremento dallo Sprint Review. L'obiettivo è fornire valore il più presto e il più spesso possibile, eliminando così i ritardi nella realizzazione del valore. Lo Sprint Review consente l'empirismo ispezionando *“la somma degli incrementi”* forniti nello Sprint in modo che l'apprendimento possa fornire gli strumenti per assumere decisioni informate sul prossimo Sprint. Laddove richiesto dalla normativa o anche solo dalle consuetudini e dalle pratiche all'interno di un'organizzazione, lo Sprint Review può essere utilizzato per riconoscere formalmente ciò che è stato realizzato, fornendo un regolare controllo qualora ritenuto utile dall'organizzazione.

Il Product Backlog è *“un elenco emergente e ordinato di ciò che è necessario per migliorare il prodotto”*. Quando le User Story vengono create, probabilmente non saranno completamente formate, ad esempio potrebbero contenere solo poche parole che descrivono il *“Voglio...”* ma non il *“Come...”* e il *“In modo che...”*. Il concetto di *“emergenza”* supporta l'idea che il dettaglio degli elementi del Product Backlog dovrebbe emergere nel tempo e l'empirismo suggerisce che *“l'ultimo momento responsabile”* è il momento giusto per realizzare questa evoluzione incrementale. Di conseguenza, man mano che una Storia si avvicina alla cima dell'elenco ordinato, diventerà sempre più dettagliata. Sarebbe uno spreco esplorare tutti i dettagli di tutte le Storie, semplicemente perché potrebbero non arrivare mai in cima al Product Backlog. C'è anche il rischio che i dettagli esplorati troppo presto perdano la loro validità nel tempo man mano che il prodotto stesso si evolve e gli obiettivi vengono

Expanded Guidance – Versione in italiano

affinati per soddisfare le opportunità e le sfide poste dal mondo reale in cui esisterà il prodotto. L'ultimo momento responsabile affinché una storia sia "pronta" per uno Sprint (cioè completamente dettagliata in termini di *Come...*, *Voglio...*, *In modo che...* e completamente formata attraverso la Conversazione e in termini di Conferma e Contesto) è, quindi, durante lo Sprint immediatamente precedente a quello in cui la Storia sarà realizzata.

Storie, Task, Progressi e Daily Scrum

Durante lo Sprint Planning, come parte fondamentale dell'argomento numero 3 (*Come verrà svolto il lavoro scelto?*), gli sviluppatori spesso trovano utile identificare le attività necessarie per trasformare la User Story in un Incremento.

Ad esempio, una User Story che ha come obiettivo "Concordare le specifiche per il retrofit del vagone ferroviario" potrebbe essere così strutturata: "Come passeggero del treno, voglio per poter ricaricare i miei dispositivi elettronici portatili *in modo che* possa lavorare o divertirmi durante il mio viaggio".

Durante la pianificazione dello Sprint questo potrebbe suddividersi in una serie di attività (task) come:

- organizzare una sessione di focus group con i passeggeri per comprenderne meglio le esigenze;
- eseguire una sessione di focus group e condividere quanto appreso con l'intero Scrum Team;
- esaminare la realizzabilità delle opzioni in base alle esigenze identificate dal focus group;
- collaborare con un esperto di sicurezza elettrica per assicurarsi di poter fornire energia in sicurezza;
- stimare i costi.

Queste attività, insieme alla User Story a cui si riferiscono e allo Sprint Goal, costituiscono lo Sprint Backlog, ossia il piano per il lavoro creato dagli sviluppatori per gli sviluppatori. Si noti che lo Sprint Goal potrebbe emergere come risultato dell'identificazione degli elementi del Product Backlog più importanti e da portare in uno Sprint o potrebbe essere definito in anticipo selezionando gli elementi del Backlog appropriati per raggiungere l'obiettivo. In ogni caso, lo Sprint Goal, quale impegno per lo Sprint, deve essere definito nello Sprint Planning.

L'avanzamento verso l'obiettivo e l'analisi del lavoro svolto per raggiungere tale obiettivo, possono essere monitorati attraverso l'uso di una lavagna KanBan, fisica o virtuale. Una lavagna fisica è lo strumento migliore per visualizzare i progressi dei Team co-localizzati, mentre una lavagna virtuale è utile per coordinare l'attività dei Team distribuiti.

Le Card e/o le note adesive che rappresentano le attività da svolgere, saranno collocate nella colonna "Da fare". Quando un membro del Team seleziona un'attività su cui lavorare, la card sarà spostata nella colonna "In corso". L'attività completata sarà quindi spostata nella colonna "Fatto". Si noti che il numero di colonne e le etichette utilizzate dipenderanno sia dalla natura del prodotto che si sta costruendo sia dall'approccio di sviluppo seguito.

L'utilizzo della lavagna KanBan permette di visualizzare intuitivamente chi sta lavorando su cosa e i progressi verso lo Sprint Goal.

Le lavagne, fisiche e virtuali, forniscono *trasparenza*.

Expanded Guidance – Versione in italiano

Il Daily Scrum dovrebbe idealmente svolgersi in un luogo in cui è possibile visualizzare la lavagna KanBan, al fine di rendere possibile l'ispezione dei progressi verso lo Sprint Goal e quindi l'adattamento dei piani per la prossima giornata di lavoro. La lavagna KanBan dovrebbe essere aggiornata durante il Daily Scrum per mantenere la trasparenza.

Per fornire trasparenza e supporto verso l'ispezione, molti Scrum Team trovano utili i grafici di burn-down. I grafici di burn-down tengono traccia dell'impegno stimato per il completamento e sono generalmente utilizzati all'interno degli Sprint per visualizzare i progressi rispetto alla previsione di consegna degli elementi dello Sprint Backlog.



Possono essere utili anche grafici di burn-up, che mostrano il lavoro completato nel tempo e i diagrammi di flusso cumulativi, che mostrano lo stato degli elementi di lavoro nel tempo (in termini di "da fare", "in corso" e "completato"). È importante notare che lo scopo principale di questi grafici è supportare l'empirismo, ovvero aiutare la trasparenza, focalizzare l'ispezione e, nella successiva iterazione, valutare se l'adattamento è efficace. Attenzione: poiché il lavoro in Scrum non si basa sull'analisi e la pianificazione dettagliata, è pericoloso utilizzare tali grafici come meccanismo per valutare le prestazioni del Team.

Expanded Guidance – Versione in italiano

Stima comparativa

La stima comparativa (comparative estimating) è la stima dell'impegno espresso come un multiplo dell'impegno previsto per un lavoro a basso costo, idealmente noto (qualcosa che è già stato fatto) o di un lavoro poco impegnativo ma ben compreso.

Si utilizza una serie di valori simile alla sequenza di Fibonacci, non lineare perché all'aumentare delle stime, queste diventano solitamente meno precise a causa di un aumento dell'incertezza. La sequenza di Fibonacci (1, 2, 3, 5, 8, 13, 21...) inizia con 1 e 2, con ogni numero successivo che è la somma dei due precedenti, quindi 3 (=2+1), 5 (=3+2), 8 (=5+3) ecc.

La serie tipicamente usata per la stima comparativa è invece una versione revisionata di quella di Fibonacci: (0, ½, 1, 2, 3, 5, 8, 13, 20, 40, 100, ∞)

Dove:

- 0 – indica uno sforzo minimo o inesistente;
- ½, 1, 2, 3, 5, 8, 13, 20, 40, 100 – sono valori crescenti associabili ad impegno che a sua volta ha una dimensione crescente;
- ∞ – quando l'impegno richiesto stimato è maggiore di 100, si utilizza il simbolo dell'infinito (∞) per indicare che la User Story richiede uno sforzo troppo grande per essere stimato. In questo caso, si dovrà procedere con una preventiva scomposizione.

Planning Poker

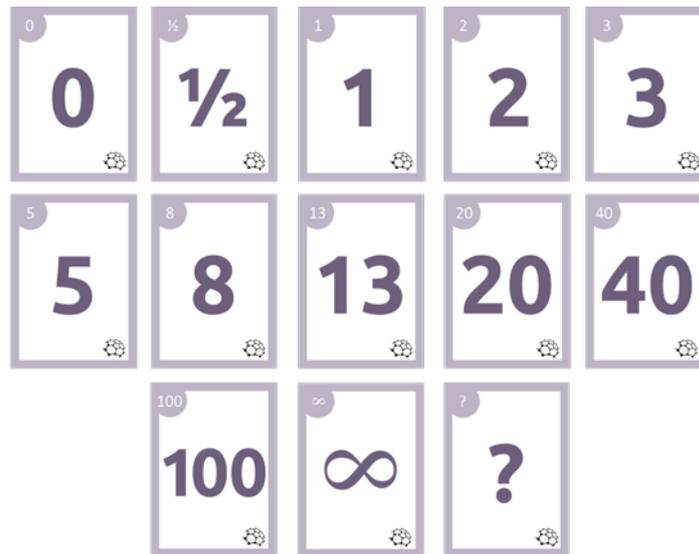
Il Planning Poker è un metodo di stima collaborativo perché incoraggia il dialogo e la discussione tra i membri del Team, ed è utile a prevenire la sottostima o, al contrario, la sovrastima delle attività, che potrebbero influire negativamente sulla pianificazione del progetto. Inoltre, il Planning Poker aiuta a rafforzare la comprensione comune delle attività e a rendere più trasparente il processo di stima. La tecnica fa uso di un mazzo di carte da gioco e utilizza come base per la stima la sequenza di numeri "adattata" vista in precedenza.

Lo Scrum Team apre la sessione individuando di comune accordo una User Story già stimata in passato e da utilizzare come base per il processo di stima che si sta avviando. Questa User Story dovrà essere "a basso sforzo" (o "low-effort", come si usa dire nella pratica) e pertanto avrà un valore assegnato, in termini di Story point, pari a 2.

Se non dovesse esistere una User Story già precedentemente stimata, il Team potrà selezionarne una tra quelle ancora da stimare che, a suo avviso, è ben definita e a basso sforzo e a cui assegna un valore pari a 2.

A questo punto, ogni membro del Team riceve un set di carte i cui valori rispecchiano la sequenza prima descritta.

Expanded Guidance – Versione in italiano



E se il Team è geograficamente dislocato o lavora in modalità smart working? In tutti quei casi in cui la sessione di Planning Poker non può avvenire in un luogo fisico, è possibile utilizzare App o tool online.

A questo punto il Team è pronto per entrare nel vivo della sessione di stima, che si apre con l'intervento del Product Owner, che selezionerà un elemento del Product Backlog e lo descriverà allo Scrum Team. Il Team avvia una discussione per confrontarsi circa le caratteristiche della User Story, per cercare di esplorare tutti gli elementi utili per valutare nel modo più preciso possibile, l'impegno necessario per completarla. Anche in questa fase, può essere utile il contributo del Product Owner, se necessario. Al termine del confronto, ogni membro del Team seleziona in piena autonomia la carta il cui valore rappresenta la propria stima. Per evitare reciproche influenze, è importante che tutte le carte siano rivelate contemporaneamente. Bene! Cosa accade ora? Se tutti sono concordi circa il valore assegnato alla User Story, quel valore ne rappresenterà senza dubbio la stima. In caso contrario, si procede con un ulteriore confronto, che coinvolgerà principalmente coloro i quali hanno attribuito i valori più alti e quelli più bassi. Seguirà quindi una nuova stima con la rivelazione contemporanea delle carte. Questo processo verrà ripetuto fin a quando non si raggiungerà il pieno accordo tra tutti i membri del Team. Solo a quel punto, il Product Owner selezionerà la User Story successiva.

È chiaro che se i membri del Team utilizzano criteri di stima molto diversi tra loro, raggiungere l'accordo potrebbe essere difficile e questo allungherebbe inevitabilmente i tempi della sessione. In questi casi, tutt'altro che rari, gioca un ruolo chiave lo Scrum Master in quanto moderatore del processo. Solitamente, se la stima oscilla tra due valori, ad esempio 5 e 8, lo Scrum Master interverrà associando alla User Story il valore più alto (nel nostro esempio il valore sarà 8). Quando tutte le User Story sono state stimate, è consigliabile ripetere il processo per ricalibrare le stime. In questo secondo passaggio il Team riesamina e confronta

Expanded Guidance – Versione in italiano

tutte le User Story che hanno un valore simile, con l'obiettivo di intercettare eventuali errori nella stima.

Il Planning Poker è quindi una tecnica di stima efficace e facile da utilizzare, che trova ampia applicazione soprattutto nei Team Scrum che lavorano su progetti di sviluppo software. La tecnica, se utilizzata con il giusto approccio, aiuta a garantire una pianificazione precisa e realistica del progetto, favorendo la collaborazione e la comunicazione tra i membri del Team.

Affinity Estimating

L'Affinity Estimating è una variazione della tecnica del Planning Poker. Prevede l'utilizzo di una superficie di lavoro, che può essere fisica o virtuale, organizzata in colonne, a ciascuna delle quali sarà assegnato un valore della sequenza (0, ½, 1, 2, 3, 5, 8, 13, 20, 40, 100, ∞).

Come per il Planning Poker, si parte da una User Story con valore pari a 2, che sarà inserita nella relativa colonna.

Per l'Affinity Estimating si può procedere in tre modi alternativi:

- 1) i membri del Team selezionano in autonomia le User Story dal Product Backlog e, sulla base della propria opinione, le inseriscono nella colonna che ha valore pari alla loro stima. Una volta esaurito il Backlog, il Team prenderà qualche minuto per rivedere le stime e spostare le carte che, secondo l'opinione individuale, sono state posizionate in modo errato. Lo Scrum Master, che funge anche in questo caso da moderatore, identifica le User Story che sembrano muoversi molto tra le colonne e le estrae. Queste storie vengono poi discusse collettivamente con il contributo del Product Owner, per ottenere il consenso sulla stima finale;
- 2) il Product Owner legge la User Story e chiunque può porre una domanda per chiarire i propri dubbi. Successivamente, la storia viene posizionata chiedendo l'accettazione da parte del gruppo. In caso di disaccordo, la User Story viene discussa ulteriormente ma entro un periodo di tempo prestabilito, ad esempio 60 secondi. Se il disaccordo persiste, la storia viene messa da parte e il processo viene ripetuto per la storia successiva. Quando tutte le User Story sono state stimate, quelle messe da parte vengono discusse fino al raggiungimento del consenso;
- 3) ogni membro del Team, a turno, legge una User Story e propone una stima. In caso di dissenso la storia viene discussa fino al raggiungimento del consenso e così via.

Il Planning Poker e l'Affinity Estimating garantiscono un corretto bilanciamento fra il processo decisionale di gruppo e quello individuale, garantendo l'indipendenza di quest'ultimo ma promuovendo al tempo stesso una maggiore interazione e una migliore comunicazione fra i partecipanti e ciò le rende tecniche applicabili anche al di fuori di contesti agili.

Expanded Guidance – Versione in italiano

Commitment Based Planning

È una tecnica semplice che può essere facilitata dallo Scrum Master per aiutare gli sviluppatori a raggiungere l'argomento 2 dello Sprint Planning.

Dal Product Backlog, ordinato dal Product Owner, lo Scrum Master seleziona il primo elemento del Product Backlog, quindi pone la domanda agli sviluppatori: " *Se questa storia fosse l'unica da realizzare nello Sprint, potresti impegnarti a realizzarla?*"

Se gli sviluppatori risponderanno "Sì, siamo felici di impegnarci in questo", lo Scrum Master seleziona l'elemento successivo dall'elenco ordinato e pone di nuovo la domanda: " *Se questa storia fosse l'unica da realizzare nello Sprint, potresti impegnarti a realizzarla?*"

Se la risposta a questa domanda è "Sì", il passaggio precedente viene ripetuto fino a quando la risposta non è un deciso "Sì". Se la risposta è "No" o anche "Non sono sicuro...", allora si può tenere una breve discussione per capire se ci sono altre storie in cima al Backlog che potrebbero essere incluse nell'impegno.

Nota: se combinata con la tecnica di stima comparativa sopra descritta, questa tecnica può essere utilizzata nel primo Sprint per *prevedere* la velocità per lo Scrum Team.

- La velocità è il numero di Story point associati alle User Story "Done" dello Sprint precedente.
 - In alcuni casi, gli Scrum Team potrebbero preferire utilizzare come indicazione una media dei punti degli Sprint precedenti.
- Anche in presenza di una velocità stabilita, questa tecnica può essere utilizzata nei casi in cui avviene un cambiamento nel Team che potrebbe influire sulle prestazioni.
 - La performance passata è l'unità di misura migliore su cui basare le stime della performance futura.
- Se Story point e Velocity non vengono utilizzati, il Commitment based planning è una buona tecnica per affrontare l'argomento 2 dello Sprint Planning.