

The background of the slide is a blue gradient with a pattern of binary code (0s and 1s) floating around. On the left side, there is a partial view of a laptop screen and keyboard.

Unità di apprendimento 5

A teal-colored rectangular box with a thin border, containing the text 'I modelli Agili'.

I modelli Agili

Metodologie agili

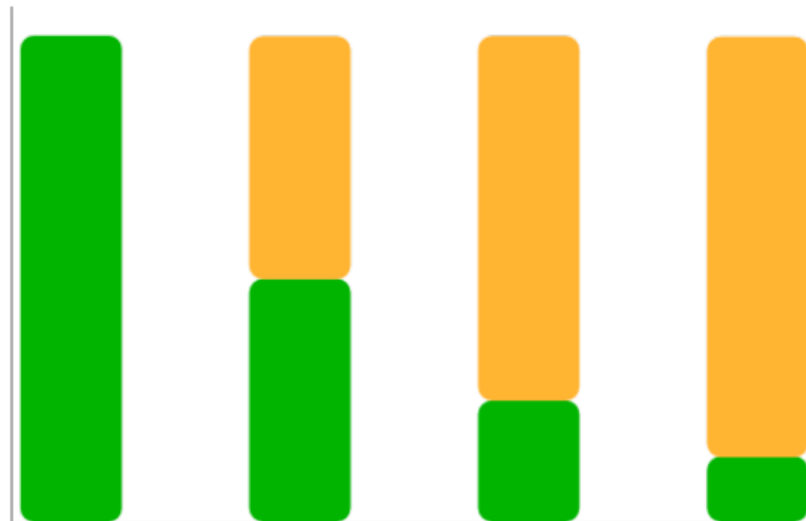
Certi sviluppi non si pianificano

- Alcuni problemi sono complessi, richiedono il contributo di molte persone
- La soluzione all'inizio non è chiara
- I requisiti del prodotto-soluzione durante lo sviluppo probabilmente cambieranno
- Il lavoro può essere articolato in incrementi
- È possibile una collaborazione stretta e un rapido feedback dagli utenti finali

Metodologie agili

Volatilità dei requisiti

Ogni sei mesi – o meno – metà dei requisiti di un prodotto software perdono di interesse



Metodologie agili

Migliorare i processi iterativi

- I processi iterativi devono produrre valore per gli stakeholder in modo incrementale e senza sprechi
- La filosofia **agile** ha lo scopo di evitare gli sprechi e le perdite inutili di tempo

Metodologie agili

Regola	Descrizione
Pianificazione realistica	I clienti devono prendere le decisioni sulla funzionalità, i programmatori devono prendere le decisioni tecniche. Aggiornate il piano di sviluppo quando è in conflitto con la realtà
Piccoli stati di avanzamento	Fornite velocemente un sistema utilizzabile, e fornite aggiornamenti in tempi brevi
Metafora	Tutti i programmatori dovrebbero condividere un racconto che illustri il sistema in fase di sviluppo
Semplicità	Progettate ogni cosa in modo che sia la più semplice possibile, invece di predisporre tutto per future complessità
Collaudo	Sia i programmatori sia i clienti devono preparare casi di prova. Il sistema deve essere collaudato continuamente
Riprogettazione	I programmatori devono continuamente ristrutturare il sistema per migliorare il codice ed eliminare parti duplicate
Programmazione a coppie	I programmatori devono lavorare a coppie e ciascuna coppia deve scrivere codice su un unico calcolatore
Proprietà collettiva	Tutti i programmatori devono poter modificare qualsiasi porzione di codice quando ne hanno bisogno
Integrazione continua	Non appena un problema è risolto, mettete insieme l'intero sistema e collaudatelo
Settimana di 40 ore	Non usate piani di lavoro poco realistici, riempiendoli di sforzi eroici
Cliente a disposizione	Un vero utilizzatore del sistema deve essere disponibile in qualsiasi momento per la squadra di progettazione
Standard per la scrittura del codice	I programmatori devono seguire degli standard di codifica che pongano l'accento sul codice autodocumentato

Metodologie agili

Etica del Movimento Agile

Stiamo scoprendo modi migliori di costruire il software facendolo e aiutando altri a farlo. Attribuiamo valore a:

Individui e interazioni più che a processi e strumenti
Software che funziona più che a documentazione completa
Collaborazione col cliente più che a negoziazione contrattuale
Reagire al cambiamento più che a seguire un piano

Metodologie agili

Individui e interazioni

meglio
di

Processi e strumenti

Software che
funziona

meglio
di

Documentazione
completa

Collaborazione
del cliente

meglio
di

Negoziazione
contratto

Rispondere al
cambiamento

meglio di

Seguire un piano

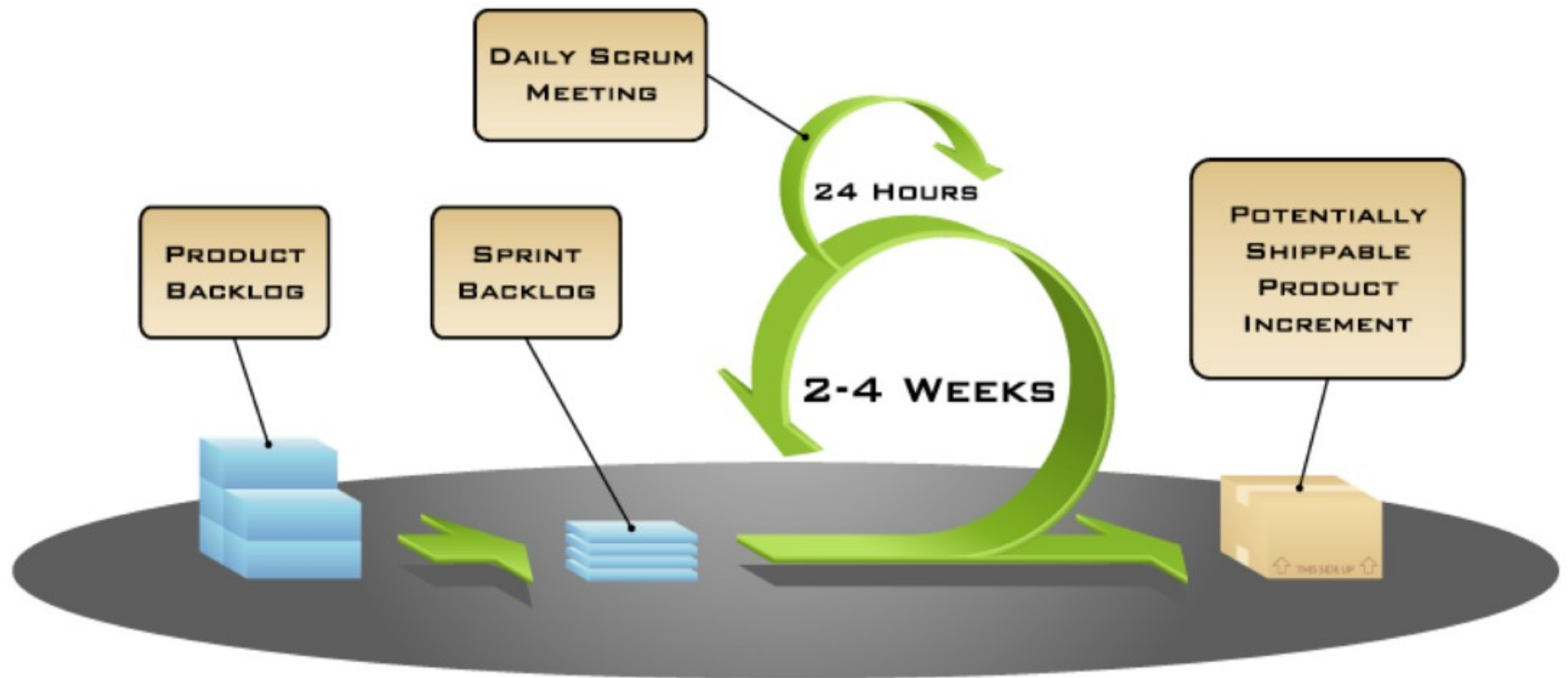
Metodologie agili

Metodi agili

I metodi agili sono una famiglia di metodi di sviluppo che hanno in comune:

- **Rilasci frequenti** del prodotto sviluppato
- **Collaborazione** continua del team di progetto col cliente
- **Documentazione** di sviluppo **ridotta**
- **Valutazione** sistematica e continua di **valori** e **rischi** dei **cambiamenti**

Modello Scrum



Modello Scrum

Scrum in 100 parole

- Scrum è un modello di processo per produrre software ottenendo il massimo valore utile nel minor tempo
- Permette al cliente di ispezionare rapidamente e ripetutamente ogni 3-4 settimane versioni funzionanti del software
- Il cliente definisce funzioni da realizzare e loro priorità. Il team di sviluppo decide quotidianamente il modo migliore di produrre le funzioni di più alta priorità.
- Ogni 3-4 settimane nasce una nuova versione che viene esaminata per decidere se continuarne lo sviluppo con un altro sprint o produrne un rilascio

Punti chiave dello Scrum

- Team di sviluppatori auto organizzante
- Metodo agile, parzialmente pianificato
- I requisiti sono catalogati nel “product backlog”, che è un insieme ordinato di user stories
- Sviluppo agile guidato da storie e test
- Il prodotto cresce in “sprint” di durata fissa
- Per ogni sprint, ogni persona del team sceglie i requisiti da realizzare da uno “sprint backlog”
- Meetings: reviews e “retrospettive”