



Metodologie Agili per lo sviluppo di applicazioni Internet Distribuite

**Agile Group
DIEE, Università di Cagliari
www.agile.diee.unica.it**

Agile Group

- ▶ **Agile Group**, gruppo di ricerca su Ingegneria del SW, Metodologie Agili ed Open Source dell'Università degli Studi di Cagliari.
- ▶ L'Agile Group si pone ai vertici italiani ed internazionali nello studio e nell'applicazione delle **Tecnologie ad Oggetti** e delle **Metodologie Agili** per lo sviluppo del software,



AgileGroup

Gruppo di Ricerca su Metodologie Agili e Ingegneria del Software
Dipartimento di Ingegneria Elettrica ed Elettronica - Università degli Studi di Cagliari

Responsabile: Prof. **Michele Marchesi**

Componenti: Prof. Giulio Concas

📧 Sandro Pinna	📧 Sabrina Ecca
📧 Ivana Turnu	📧 Federico Scarpa
📧 Roberta Quaresima	📧 Guido Porruvecchio
📧 Leonardo Paschino	📧 Alessandro Murgia
📧 Katuscia Mannaro	📧 Walter Ambu
📧 Selene Uras	📧 Stefano Marras
📧 Mario Locci	

📧 AgileGroup - Dipartimento di Ingegneria Elettrica ed Elettronica
Università degli Studi di Cagliari

<http://agile.diee.unica.it>

infoagile@diee.unica.it

Tel. +39 070 675 5781
Fax. +39 070 675 5782

Agile Group

FlossLab Srl
Il primo spinoff Universitario di Cagliari,
nato dall'AgileGroup

Metodologie Agili

Le Metodologie Agili di sviluppo software stanno acquisendo sempre maggior popolarità grazie alla loro capacità di rispondere velocemente alle variazioni di un mercato a fortinamicità come è quello ICT.

Si pongono in antitesi rispetto alle metodologie tradizionali e predittive che sono caratterizzate da processi pesanti sia in termini di costi che di tempi, in quanto tendono a voler prevedere e definire tutte le possibili evoluzioni che un sistema software potrebbe avere nel futuro.

Le MA, invece, propongono delle modalità di lavoro orientate al raggiungimento di piccoli risultati a breve termine, mantenendo il processo di sviluppo flessibile e pronto ad inseguire le inevitabili variazioni delle condizioni e dei requisiti...e dei clienti!

Il processo si basa su rilasci brevi, iterazioni di sviluppo e test molto frequenti che permettono di costruire, a piccoli passi, un sistema complesso in modo incrementale, incentivando al contempo il feedback del cliente.

Open Source

I numerosi vantaggi derivanti dall'utilizzo di software proprietario hanno portato ad un'apertura verso il mondo Open Source, che negli ultimi anni sta diventando protagonista sia nelle Imprese che nelle PA.

Infatti il mondo Open Source offre rilevanti potenzialità di carattere economico, culturale e sociale, oltre alla fondamentale caratteristica di garantire elevata qualità e sicurezza dei dati trattati e conservati. Il software libero sta sperimentando un'attenzione senza precedenti da parte dello Stato italiano che per la prima volta sta incominciando ad incentivare l'uso del sw libero.

Sono numerosi i vantaggi che l'Open Source può offrire:

- Nessun costo di acquisto del software e minori costi generali
- Il Software è sempre e costantemente aggiornato, di alta qualità e conforme agli standard.
- L'assistenza e l'evoluzione del prodotto sono garantite dalla comunità
- Indipendenza dal produttore e dai fornitori.
- Apertura e disponibilità del codice sorgente.
- Modello di business ormai consolidato

Tecnologie ad Oggetti

Metodologie di analisi e progetto orientate agli oggetti.
Linguaggio UML.
Linguaggi ad oggetti: Java, C#, C++, Smalltalk, Ruby.
Testing automatico di sistemi a oggetti.

Modellazione e simulazione di sistemi

Modelli di sistemi ad eventi discreti.
Sviluppo di simulatori ad oggetti (traffico ferroviario e veicolare; reti dati e Internet; sistemi economici; processi aziendali).

Finanza computazionale

Modelli e simulatori di mercato.
Tecniche automatiche di trading per la gestione di "hedge fund".
Gestione ottima di portafogli.
Gestione del rischio.
Pricing di derivati.

Sistemi complessi

Analisi di reti complesse, con applicazioni a: motori di ricerca, data mining, data ware-housing, sistemi economico-sociali, modellazione della rete Internet.

E-government

Architetture di cooperazione applicativa.
Standard di interoperabilità.
Metodologie di gestione progetti per PP. AA.
Integrazioni di servizi/sistemi.

Servizi Offerti

Una elevata conoscenza acquisita nel campo delle metodologie di sviluppo software ed in particolare delle "Metodologie Agili" e l'esperienza maturata nello sviluppo di applicazioni software e realizzazione di progetti Open Source, consente all'Agilegroup di assicurare elevati standard di qualità dei servizi offerti finalizzati alla soddisfazione del cliente.

L'Agilegroup fornisce servizi di consulenza:

- Metodologie di sviluppo software Agili, in particolare Scrum e Extreme Programming (XP)
- Sviluppo e gestione di progetti Open Source, in tutte le sue fasi: da quella **tecnica**, fornendo valutazioni sulle tecnologie Open Source che meglio si prestano allo sviluppo, a quella della **gestione** della comunità di sviluppo.
- Servizi di migrazione a soluzioni Open Source, incluso analisi di costi e benefici, gestione del processo di migrazione, attività di supporto e consulenza
- Gestione di progetti di sviluppo con team distribuiti
- Supporto alla progettazione ed alla realizzazione di attività di innovazione e ricerca

Agilegroup inoltre organizza corsi avanzati sulla gestione progetti **Open Source**, **Metodologie Agili** di sviluppo software e **tecnologie J2E**.

I percorsi di formazione saranno personalizzati e mirati a soddisfare le vostre esigenze.

Dati essenziali:

- Fondata nel marzo 2007
- Capitale sociale di 20.000 €
- Ha come soci anche l'Università degli Studi di Cagliari e la SIRS, Società Finanziaria per la Rinascita Sarda (finanziaria della Regione Autonoma della Sardegna)
- Mission: sviluppo software con tecnologie agili avanzate e fornitura di servizi legati a prodotti software Open Source, sia sviluppati a FLOSSLab che da comunità terze.
- Mercato di riferimento: Pubbliche Amministrazioni Locali, PMI, imprese in genere, in Sardegna, Italia e nei paesi del Mediterraneo.
- Progetti finanziati:
 - "Migrazione a Open Source e Integrazione Applicativa per PP.AA.", finanziamento MIUR art. 11, DM 593 del 8/9/2000, per 755K€
 - "Metodologie di "forking" da progetti FLOSS e loro applicazione ad un DMS", finanziamento Regione Autonoma della Sardegna, POR 2000-2006, Mis. 3.1.3.B.2, per 380K€

Consulenza e Formazione:

- Assistenza all'adozione di FLOSS di vario tipo da parte di PA e altre organizzazioni; analisi costi-benefici, maturità dei prodotti Open Source, processo di migrazione.
- Processi e certificazione di qualità per la produzione del software
- Integrazione sistemi e sviluppo software su commessa
- Review di documenti di analisi, di progetto e di codice Java, C, C++, Ruby, Smalltalk
- Condulenza informatica
- Formazione, mentoring, scouting tecnologico su analisi e progetto di sistemi ad oggetti, linguaggi Java, C++, Ruby, Metodologie Agili per la produzione del software, Testing automatico, Software Open Source, ingegneria del software in generale.

Prodotti sviluppati:

FLOSS-HIS

"Fork" italiana del HIS (sistema informativo ospedaliero) "Care2K", in grado di gestire l'amministrazione di un ospedale o di un centro clinico. Basato interamente su tecnologia Web, è un potente strumento molto usato in vari paesi del mondo per gestire strutture ospedaliere. La fork italiana aggiunge la gestione a norma di privacy e sicurezza, la classificazione ICD9 usata dalla Sanità italiana, l'integrazione con strumentazione di laboratorio e l'aggiunta di numerosi tipi di esami comunemente utilizzati in Italia.

FLOSS-AR

Portale Web per la gestione dell'Anagrafe della Ricerca di un'Università o di un ente di ricerca. Basato sulla piattaforma Java Open Source "JAFS", mette a disposizione ampie funzionalità sia per la gestione interna dei prodotti della ricerca a scopo di valutazione, che per la presentazione a utenti esterni di prodotti, servizi, uso di strumentazione offerti.

FLOSS-DMS

Sistema integrato Open Source per la gestione del protocollo informatico, della documentazione e del workflow in organizzazioni ed Enti Pubblici.

FLOSS-LMS

Piattaforma eLearning Open Source per la formazione

Il progetto DART

- ▶ **Sono state definite delle nuove Metodologie Agili orientate allo sviluppo distribuito nell'ambito del progetto Dart per la realizzazione di un Motore di Ricerca Semantico**

Le applicazioni distribuite

- ▶ Un'applicazione costituita da diversi componenti, che viene eseguita in ambienti separati, normalmente su diversi calcolatori connessi in rete.
- ▶ Le architetture tipiche sono quelle:
 - ▶ two-tier,
 - ▶ three-tier (in cui viene operata una separazione della logica di presentazione da quella di business e dai dati),
 - ▶ multi-tier (con client multipli).

Problematiche dello sviluppo di applicazioni distribuiti

- ▶ Lo sviluppo di applicazioni distribuite non può fare affidamento su pratiche metodologiche di sviluppo collaudate e consolidate come avviene per i sistemi software.
- ▶ Talvolta lo sviluppo di applicazioni distribuite comporta lo sviluppo in team distribuiti
- ▶ La loro crescente diffusione e complessità impone l'adozione di approcci che supportino adeguatamente le persone coinvolte nel processo di sviluppo.

Aspetti dello sviluppo Web distribuito

- ▶ **Team di sviluppo:** team multidisciplinari che vanno coordinati definendone compiti e responsabilità
- ▶ **Supporto per diverse piattaforme:** deve essere in grado di essere eseguita su una varietà di ambienti differenti
- ▶ **Qualità:** progetti web di bassa qualità per mancanza di refactoring e la manutenzione del codice
- ▶ **Testing e validazione:** sono necessarie particolari pratiche, che testino aspetti spesso secondari in altri contesti
- ▶ **Sviluppo rapido:** è necessario quindi un approccio per facilitare continue integrazioni e gestire al meglio le nuove release

Le Metodologie Agili

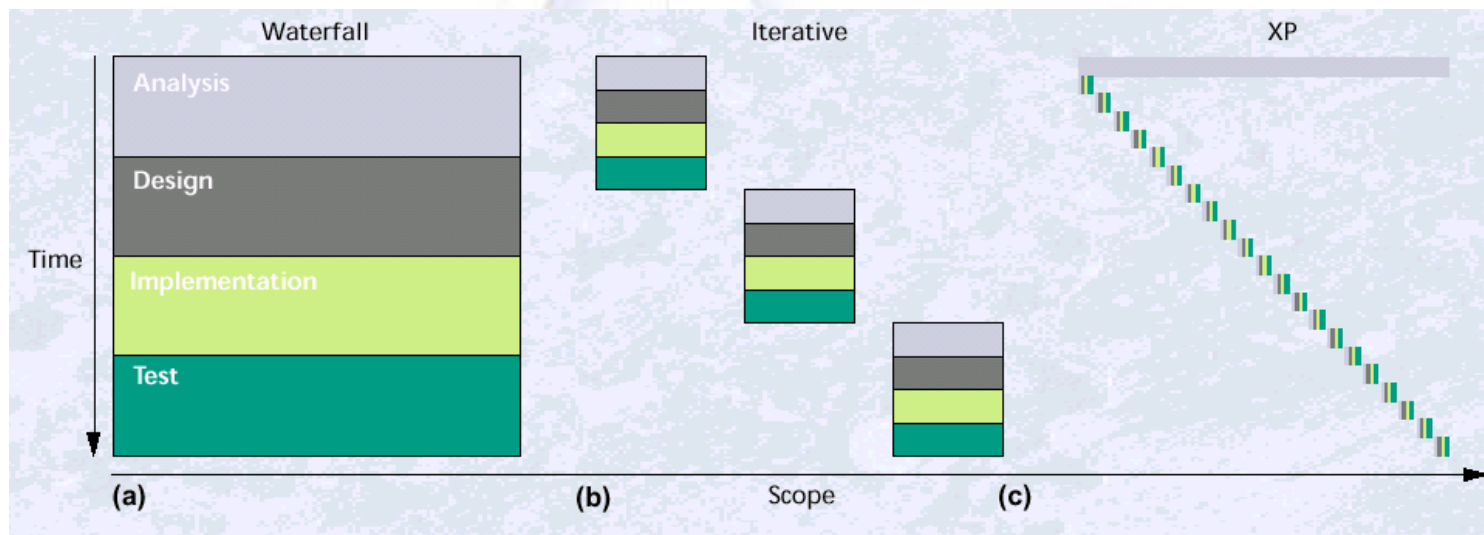
- ▶ Le Metodologie Agili si propongono di superare i limiti degli approcci tradizionali allo sviluppo di sistemi software.
- ▶ La constatazione che un numero molto limitato di progetti software ha successo (circa il 10%), ha messo in moto, nel corso degli anni '90, una spinta verso l'elaborazione di metodologie che meglio si adattassero ai nuovi tipi di sistemi.

Le Metodologie Agili

- ▶ Le MA lasciano al cliente la **libertà di decidere** in qualunque momento quali funzionalità realizzare e di richiedere modifiche delle funzionalità esistenti
- ▶ La parola **Agile** si riferisce alla possibilità di far fronte al continuo cambiamento dei requisiti
- ▶ Identificare i requisiti in modo rigoroso e formale è un processo lungo e costoso, e risulta spesso poco utile perché i requisiti possono cambiare molto velocemente

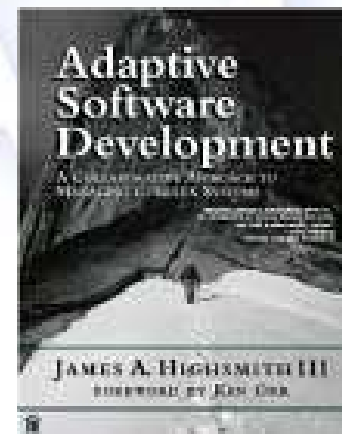
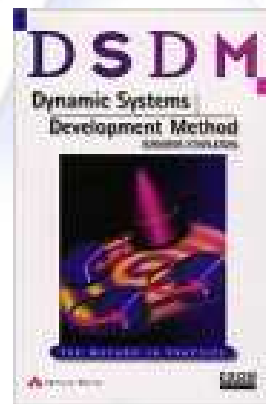
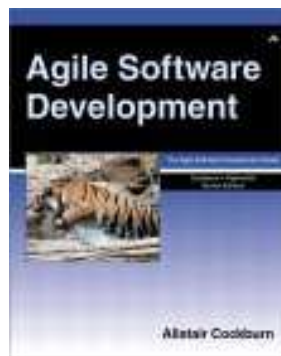
Le Metodologie Agili

- ▶ Adattative e non predittive
- ▶ Orientate alle persone e non al processo
- ▶ Mantenuite semplici per poter reagire in fretta
- ▶ Incrementali e iterative, con iterazioni molto corte
- ▶ Ruoli intercambiabili degli sviluppatori
- ▶ Basate su testing e codifica e non su analisi e progetto



Le Metodologie Agili

- ▶ **SCRUM**
- ▶ **Extreme Programming (XP)**
- ▶ Adaptive Software Development (ASD) e Agile Project Management (APM)
- ▶ Crystal Methods
- ▶ Lean Software Development
- ▶ Feature Driven Development (FDD)
- ▶ Dynamic Systems Development Method (DSDM)



Le Pratiche Agili per applicazioni Internet

- ▶ **Pair Programming:** utile per comprendere a fondo il dominio del problema da affrontare. Esistono esempi di adozione del pair programming in ambiente distribuito
- ▶ **Gruppo completo:** avere varietà di diverse competenze; questo implica particolare attenzione nello scegliere tutte le capacità necessarie per il successo del progetto.
- ▶ **Sit Together:** tutte le figure coinvolte nel processo dovrebbero lavorare in un ambiente open space per massimizzare la comunicazione. Questa pratica è facilmente gestibile e si può applicare in ambiente distribuito mediante l'uso di un wiki o una mailing list al fine di favorire il senso del team fra gli attori coinvolti.

Le Pratiche Agili per applicazioni Internet

- ▶ **Qualità del codice e Object Orientation:** adottare le “buone abitudini” date dalla programmazione ad oggetti, rendendo il codice di qualità superiore.
- ▶ **Iterazioni brevi :** avere iterazioni lunghe al massimo qualche settimana.
- ▶ **User stories:** nella prassi XP le storie sono scritte dal cliente, ma nel nostro caso questo non è sempre vero.
- ▶ **Pratiche di codifica:** codice il più possibile semplice, uso delle CRC cards, convenzioni sui nomi consistenti; riduzione del rischio mediante l'uso di prototipi.
- ▶ **Testing :** test unitari e HTTPUnit. I test devono essere indipendenti dal browser, dato che possono esserci differenze anche da una versione all'altra.

Le Metodologie Agili per applicazioni Internet

<i>Pratica</i>	<i>Molto adatta</i>	<i>Applicabile senza ostacoli</i>	<i>Sconsigliata o non testata</i>
Pair Programming	X		
Gruppo Completo	X		
Ambiente di lavoro		X	
Sit Together		X	
User Stories		X	
Test Driven Development		X	
Cliente sul posto		X	
Pratiche e convenzioni di codifica		X	
Semplicità		X	
Proprietà collettiva del codice			X
Refactoring	X		
Brevi cicli di rilascio	X		
Progetto incrementale	X		
Iterazioni brevi	X		
Integrazione Continua		X	