

# UW@ SOFTWARE PROCESS

## 0.0 Introduzione

L'adozione di un processo di produzione standard consente a un'organizzazione di dare ripetibilità al processo e di raggiungere uno standard di qualità del prodotto. In particolare la produzione di software è un'attività dove la soddisfazione dei requisiti, l'affidabilità e l'efficienza giocano un ruolo fondamentale e delicato. Un software è un prodotto intangibile, difficile da valutare a priori, l'attività di sviluppo è particolarmente complessa da organizzare e gestire per la sua natura iterativa ed evolutiva, difficilmente predicibile. Il mercato moderno richiede inoltre che i prodotti vengano sviluppati rapidamente pur presentando tutti gli attributi di qualità del caso, ecco dunque che poter contare su un processo produttivo controllato ed efficiente risulta fondamentale se non si vuole andare incontro a perdite materiali sia in termini di tempo che di denaro.

## 0.1 Linee Guida

Il processo di produzione adottato si ispira fortemente al già affermato modello XP (eXtreme Programming) formulato da Kent Beck, tale modello assicura una produzione agile, di tipo adattivo, improntata sulle risorse umane e su una stretta collaborazione col cliente. I principi fondamentali ripresi dal modello stesso e riadattati alle esigenze e alle risorse di cui la nostra organizzazione dispone sono i seguenti:

- ✓ Planning: identificazione dei requisiti attraverso casi d'uso definiti insieme al cliente che li classifica in base alla priorità.
- ✓ Simple Design: attitudine a scegliere sempre la soluzione più semplice in fase di progettazione.
- ✓ Testing: definizione di test sistematici da effettuare al termine della codifica e dopo qualsiasi modifica del codice effettuata.
- ✓ Refactoring: modificare il codice ogni volta che si ritiene di poter renderlo migliore o di poterlo semplificare.
- ✓ Collective Code Ownership: proprietà collettiva del codice da parte dell'intero gruppo di lavoro, chiunque può modificare il codice per migliorarlo.
- ✓ Coding Standards: adozione di stili di codifica simili e tool di sviluppo compatibili.
- ✓ Stand Up Meetings: incitamento ai rapporti personali tra sviluppatori, brevi riunioni in piedi per confrontarsi e definire le attività.

- ✓ Stay agile: le decisioni implementative importanti vengono rimandate il più possibile e si cerca di mantenere un'architettura aperta del sistema, così da avere un costo di modifica del codice costante con l'avanzare del processo di produzione.

## **1.0 Planning**

La prima fase prevede la definizione assieme al cliente dei casi d'uso principali, ovvero quelli giudicati dal cliente ad alta priorità realizzativa. I casi d'uso vengono rappresentati tramite UML 2.0 (Unified Modeling Language). Ne segue la stesura di un documento dove vengono riportate le funzionalità e i requisiti richiesti dal cliente. Si passa ad un gantt di progetto minimale, vengono concordate la data di rilascio del sw in versione beta, e la data entro la quale sarà disponibile un primo prototipo da visionare. Vengono gestite le risorse sia in termini di personale e ruoli svolti, sia in termini di linguaggi e strumenti di sviluppo adottati.

## **1.1 Iteration**

Inizia una fase iterativa che porta alla implementazione dei casi d'uso definiti all'inizio di ogni ciclo. Prima della codifica vengono definiti i test e si effettua un'eventuale refactoring di parti del codice ove ritenuto opportuno. Al termine dell'attività di codifica vengono lanciati i test sulle singole unità del sistema e se non si riscontrano anomalie si passa al ciclo successivo, si definiscono i nuovi casi d'uso da implementare e si progettano i nuovi test da effettuare. Le varie funzionalità del software saranno divise in unità autonome, dotate di interfaccia per il dialogo tra unità diverse e testate separatamente. Tale tecnica consente di poter effettuare cambiamenti anche consistenti all'interno di una singola unità senza modificare il funzionamento complessivo del sistema e senza dover modificare altre unità.

## **1.2 Integration**

Quando l'implementazione delle singole unità del sistema è terminata si passa all'integrazione e ai relativi test sul sistema completo. Se non si verificano anomalie di funzionamento si passa al rilascio della versione beta che il cliente può provare per un periodo di tempo. Qualora il cliente voglia aggiungere o modificare funzionalità al software si ritorna alla fase iterativa (1.1).

### **1.3 Release**

Nel momento in cui il cliente accetta la versione beta del software si passa ad affrontare i problemi di installazione e distribuzione del prodotto. Inizia un periodo preventivamente concordato in cui è possibile effettuare interventi di manutenzione correttiva e/o migliorativa.