

Per lo sviluppo dell'applicazione web Smart Learning si è utilizzato un processo di sviluppo di tipo **UP**, o **Unified Process**. Infatti, tale metodologia riconosce lo sviluppo del software come un'attività guidata dalla definizione dei requisiti funzionali, espressi attraverso i casi d'uso. L'analisi dei casi d'uso permette di definire le caratteristiche dell'architettura software che li realizza in modo integrato. Da questo punto di vista, l'UP è centrato sull'architettura, in quanto, una volta stabilita quella di base, il progetto procede per incrementi che integrano le funzionalità individuate nell'analisi.

Il progetto, seguendo lo Unified Process, attraversa diverse fasi, le quali possono essere reiterate più volte:

Fase	Numero di iterazioni	Inizio	Fine
Ideazione	1	Settimana 1	Settimana 2
Elaborazione	1	Settimana 2	Settimana 3
Costruzione	2	Settimana 3	Settimana 5
Transizione	2	Settimana 4	Settimana 6

Tabella 1: Distribuzione temporale del processo di sviluppo

Ogni fase è a sua volta attraversata da diversi flusso di lavoro, che vengono eseguiti ad ogni iterazione, ed inoltre essa si deve concludere con la realizzazione di determinati obiettivi (**milestone**):

Fase	Descrizione	Milestone
Ideazione	Analisi di fattibilità, modellazione dei primi casi d'uso (20%), definizione dei requisiti, risk assessment	Studio di fattibilità, Use Case Diagram, Specifica dei requisiti
Elaborazione	Identificazione attori e casi d'uso (80%), definizione dei requisiti non funzionali, definizione piano di progettazione e sviluppo, analisi del dominio, progettazione dell'architettura	Prototipo architetturale, Piano di progettazione e sviluppo
Costruzione	Sviluppo software, integrazione e testing, documentazione dello sviluppo	Prodotto software stabile e utilizzabile da end-users
Transizione	Installazione, esecuzione ed analisi, bug fixes	Utenti soddisfatti

Tabella 2: Descrizione del processo di sviluppo

La suddivisione in iterazioni presenta numerosi vantaggi, quali:

- Valutare progressivamente l'avanzamento del progetto;
- Gestire al miglior modo i fattori rischio;
- Fornire rapidamente una versione operativa del sistema;
- Avere maggiore flessibilità.

Si è quindi deciso di definire un piano di progettazione e sviluppo sufficientemente dettagliato da mostrare le iterazioni e i criteri di valutazione per ciascuna di esse:

Fase	Iterazione	Descrizione	Criteri di valutazione
Ideazione	Iterazione preliminare	Analisi di fattibilità (architettura a microservizi e monolitica, tecnologie per scambio flussi multimediali - Janus WebRTC); Definizione requisiti e casi d'uso	Il contenuto dei casi d'uso principali è una rappresentazione accurata di ciò che il software fornirà; Fattibilità
Elaborazione	E1 Iteration – Develop Architectu- ral Prototype	Analisi, design e sviluppo prototipo architetturale per R1	Prototipo architetturale stabile e approvato; Principali elementi di rischio identificati e risolti in modo credibile; Progetti e piani di sviluppo sufficientemente dettagliati, accurati e credibili
Costruzione	C1 Iteration – Develop R1 Release	Sviluppo R1, integrazione e testing; Analisi e design R2	R1 stabile e utilizzabile da end-users; Piano di sviluppo per R2 sufficientemente dettagliato ed approvato
	C2 Iteration – Develop R2 Release	Sviluppo R2, integrazione e testing;	R2 stabile e utilizzabile da end-users
Transizione	T1 Iteration – R1 Release	Deploy e bug fixes R1	Soddisfamento per il set di funzionalità e il loro funzionamento
	T2 Iteration – R2 Release	Deploy e bug fixes R2	Soddisfamento per il set di funzionalità e il loro funzionamento

Tabella 3: Descrizione del processo di sviluppo con iterazioni

0.1 Releases

Il documento descrive le prime due releases del sistema. Per la gestione delle release è stato utilizzato **GitHub**. Le funzionalità di base sono state implementate nella **prima release (R1)**, la quale include:

- La registrazione di un utente;
- L'upgrade di un profilo utente ad un profilo di tipo docente;
- Il login;
- La creazione di una lezione da parte di un docente;
- L'aggiunta di una programmazione per una determinata lezione da parte di un docente;
- La possibilità per un docente di visualizzare le lezioni create e le relative programmazioni;
- La ricerca di una lezione basata sul cognome del docente o sul topic della lezione.
- La prenotazione di una lezione da parte di un utente;
- La possibilità di visualizzare le lezioni prenotate da parte di un utente;

La **seconda release (R2)**, d'altra parte, include le seguenti funzionalità aggiuntive:

- L'avvio della videochiamata reattiva ad una lezione da parte del docente;
- La possibilità di entrare in una videochiamata avviata da parte di un utente prenotato.

-
- La possibilità di terminare una lezione da parte di un docente (tale comando termina la videochiamata per tutti i partecipanti);
 - La possibilità di terminare la videochiamata da parte di un utente generico;
 - La condivisione simultanea di più schermi all'interno della stessa videochiamata;
 - La possibilità di spegnere il microfono e la videocamera all'interno di una videochiamata.

Per quanto riguarda le **releases future**, invece, si è ipotizzato di progettare le seguenti funzionalità:

- Includere il sistema di pagamento PayPal per effettuare le prenotazioni;
- La possibilità da parte di un utente di recensire una lezione alla quale ha partecipato;
- Associare ad ogni docente uno “score” basato sulle recensioni ricevute ed una pagina contenente tali recensioni;
- Fornire all'utente la possibilità di esprimere una preferenza per un “topic” (tale funzionalità è stata implementata solo parzialmente, ovvero sono state implementate le API a livello Data Layer);
- Fornire ai docenti, quando creano la lezione, la possibilità di visualizzare i topic preferiti dagli utenti (tale funzionalità è stata implementata solo parzialmente, ovvero sono state implementate le API a livello Data Layer);
- Un sistema di notifiche tramite e-mail che:
 - Invi una notifica all'utente nel caso in cui un docente crei una lezione relativa ad uno dei topic per i quali ha espresso la propria preferenza;
 - Invi una notifica all'utente quando un docente avvia la videochiamata relativa ad una lezione prenotata;
- Aggiunta di un meccanismo di autenticazione e autorizzazione basato su JWT Token.