***UNIBite – Gestione del progetto***

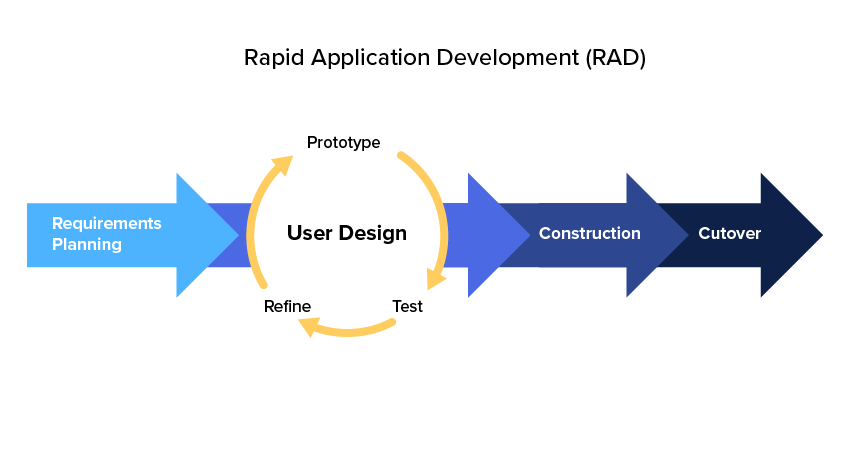
Università degli studi di Bergamo  
Ingegneria Informatica

Beccarelli Raissa Matr. 1086785  
Locatelli Giacomo Matr. 1086262  
Valceschini Marco Matr. 1086356

***Immagine che contiene testo, Carattere, Elementi grafici, simbolo

Descrizione generata automaticamente***

**1 Software Life Cycle**Il modello di processo che viene utilizzato è il metodo RAD (Rapid Application Development). Tramite esso saremo in grado di suddividere il ciclo di vita del progetto in 4 fasi principali: pianificazione dei requisiti, progettazione (con eventuali fasi di prototipazione), implementazione e taglio (collaudo del software).



Questo tipo di processo permette appunto di sfruttare la prototipazione e i principi dei processi agili, aiutando nella definizione dei requisiti di progetto e consentendo un'elevata interazione con gli utenti. La prototipazione ci sarà utile nelle prime fasi dello sviluppo del progetto per elaborare i requisiti già individuati e per arricchirlo con requisiti nuovi.  
Tramite questo metodo sarà utilizzato il concetto di timebox, ovvero degli intervalli di tempo in cui i membri svolgeranno attività lavorative al fine di raggiungere gli obiettivi prefissati per quell’intervallo di tempo.  
I nostri timebox saranno così definiti:

|  |  |
| --- | --- |
| 5-11 novembre | Project Plan |
| 12-18 novembre | Gestione del progetto e specifica dei requisiti |
| 19-26 novembre | Diagrammi UML (casi d’uso, macchina a stati, sequence, class) |
| 27 nov - 4 dic | Creazione DB e implementazione interfaccia grafica |
| 5-12 dicembre | Creazione classi e implementazione jooq  Architettura e design |
| 13 dic – 15 gen | Scrittura codice |
| 16-24 gennaio | Testing e Manutenzione, UML (activity, package, component, communication) |

Sarà fondamentale anche l'uso di Papyrus strumento per la creazione dei diagrammi UML, alcuni dei quali saranno tradotti tramite il metodo Model Driven Architecture (MDA). Esso si basa su modelli, ovvero prima di scrivere subito il codice, si costruiscono vari livelli di rappresentazione del sistema che possono poi essere trasformati in codice eseguibile.

**2 Configuration Management**

Durante lo sviluppo del progetto sarà utilizzato lo strumento di GitHub e GitHub desktop per favorire la collaborazione e per poter lavorare autonomamente, ognuno sul proprio dispositivo. Tramite GitHub potremo lavorare attraverso l’utilizzo di branch e issues, questi ultimi assegnati ed implementati anche grazie all’utilizzo della Kanban Board aggiornata in tempo reale.

Il branch principale del repository è composto da due cartelle:

* UNIBite, al cui interno sarà presente il codice;
* Documents, al cui interno ci sarà tutta la documentazione (project plan, gestione del progetto, specifica dei requisiti, design, testing e manutenzione) e i diagrammi UML relativi al progetto;
* UML, al cui interno saranno presenti i diagrammi UML relativi.

**3 People Management**

Il team è composto da tre membri: Beccarelli Raissa, Locatelli Giacomo e Valceschini Marco.

Il gruppo è basato su una struttura SWAT, con tecniche simili ad un team agile, questo perché le competenze possedute non possono essere definite di alto livello, ma possono essere considerate sufficienti per lo svolgimento del progetto.

Sulla base di questa struttura i membri collaborano ponendo il focus principale sulla realizzazione del progetto, prendendo decisioni e suddividendo i vari compiti all’interno dei timebox in modo collettivo. Per questo motivo non viene definito un leader a capo del progetto e non si pone una gerarchia tra membri, in quanto essi sono tutti collaboratori allo stesso livello.