

Immagine che contiene testo, logo, simbolo, emblema

Descrizione generata automaticamente

**Software Engineering**

Project Planning - Gruppo 17

Immagine che contiene logo, Carattere, Elementi grafici, grafica

Descrizione generata automaticamente

**COMPONENTI DEL PROGETTO:**

Emmanuele Iovino - matr. 061270 5985

Miriana Molaro - matr. 061270 5824

Daniele Miranda - matr. 061270 5847

Caterina Nugnes - matr. 061270 6767

**BREVE DESCRIZIONE DEI REQUISITI E DEI RISULTATI DEL PROGETTO SOFTWARE**

L’obiettivo è quello di realizzare una calcolatrice che effettua le seguenti operazioni sui numeri

complessi: addizione, sottrazione, moltiplicazione, divisione, radice quadrata e inversione del segno.

La calcolatrice sarà implementata attraverso l’utilizzo dello stack. È inoltre prevista sia la manipolazione dello stack tramite vari comandi che l’utilizzo di variabili.

**DEFINIZIONE DELLE FASI DEL PROGETTO**

Il modello di processo che verrà seguito è il modello a cascata.

* Project Planning
* Requisiti Ingegneristici
* Design
* Testing e Implementazione

**RISORSE**

**Risorse software**

* **Linguaggi di sviluppo:**

Java: La versione utilizzata è JDK 8.

Questo linguaggio verrà utilizzato per l’implementazione del codice.

UML: PlantUML verrà utilizzato per l’implementazione dei diagrammi

NetBeans: IDE utilizzato per i linguaggi scelti

* **Tools:**

JUnit: È un framework che sarà utilizzato per testare il codice.

OnlineGantt: tool online sarà utilizzato per la realizzazione del diagramma di Gantt.

<https://www.onlinegantt.com/#/gantt>

GitHub: utilizzo di una piattaforma online per la condivisione e la gestione di progetti software.

È un sistema di gestione di repository, di cui utilizzeremo le funzionalità di hosting.

<https://github.com/Emma-nuele/group17_software_eng>

JavaFX: framework software per la realizzazione di interfacce utente grafiche (GUI).

Microsoft Teams: piattaforma online per i meeting.

**Risorse umane**

**Team**: Emmanuele Iovino, Miriana Molaro, Daniele Miranda, Caterina Nugnes.

**SVILUPPO FASI DEL PROGETTO E PARTIZIONAMENTO DELLE RISORSE UMANE** \*

* **Pianificazione:**

Pianificazione attività: tre risorse

Gantt chart: una risorsa

* **Requisiti:**

Use Cases Definition: team

Use Cases Diagram: due risorse

Interface Mock-up: due risorse   
Traceability Matrix: due risorse

* **Design:**

Class Diagram: team

Sequence Diagrams: team

Updated version of the Traceability Matrix: due risorse

* **Testing e Implementazione:**

Functional test Cases: due risorse

Unit Test Cases: due risorse

System Source Code: due risorse

Unit Test Cases con Junit: due risorse

Updated version of the Traceability Matrix: due risorse

\*Il team si occuperà di revisionare tutto il lavoro svolto

Le risorse umane non impegnate in una task verranno ridistribuite temporaneamente in modo dinamico.

Al termine di ogni fase, sarà rilasciato un documento che contiene il prodotto di ogni sotto-task.

**Organizzazione risorse umane**

Gli incontri di organizzazione e sviluppo avverranno in modalità blended.

Gli incontri frontali saranno svolti in sede universitaria, mentre quelli da remoto saranno svolti su apposite piattaforme.

**RISCHI**

**IDENTIFICAZIONE DEI RISCHI**

1. Ritardi nella schedule: le fasi potrebbero richiedere più tempo di quello previsto, rallentando a catena le fasi successive del progetto.
2. Imprevisti di progetto: l’utilizzo dei vari tools potrebbe causare rallentamenti in quanto i membri del team non hanno una formazione completa per ognuno di essi.
3. Modifiche impreviste di progetto: modifiche dei requisiti e di implementazione del codice.
4. Inadeguatezza dei test: analisi superficiale dei casi critici e dei casi limite.
5. Incomprensione dei Requisiti: stesura dei requisiti non soddisfacente, con ridondanze e ambiguità. Ciò potrebbe causare problemi nelle fasi successive.

**MITIGAZIONE RISCHI**

1. Pianificazione realistica della schedule. Monitoraggio e revisione dei progressi.
2. Impiegare risorse per la formazione all’utilizzo dei vari tools.
3. Focalizzazione sulla manutenibilità del codice con l’obiettivo di fronteggiare gli imprevisti.
4. Dedicare sufficiente spazio alla fase di testing, verificarne la robustezza tramite test automatizzati.

Implementare una strategia di testing rigorosa e completa ed effettuare ricorrenti test al rilevamento di errori.

1. Maggiore attenzione alle specifiche dei requisiti, avvalendosi anche ad un approccio schematico.

**Immagine che contiene testo, schermata, diagramma, Diagramma

Descrizione generata automaticamenteDIAGRAMMA DI GANTT**

Il progetto avrà inizio il 20 Novembre e terminerà il 15 Dicembre.

Pianificazione: dal 20 al 22 Novembre.

Requisiti Ingegneristici: dal 23 al 28 Novembre.

Design: dal 29 Novembre al 4 Dicembre.

Testing e Implementazione: dal 5 al 15 Dicembre.