**Software Engineering**

Project Planning - Gruppo 17

**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**

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

L’obiettivo è quello di realizzare una calcolatrice.

Essa effettua le seguenti operazioni sui numeri complessi: addizione, sottrazione, moltiplicazione, divisione, radice quadrata e inversione del segno.

La calcolatrice sarà implementata attraverso l’utilizzo di uno stack.

**DEFINIZIONE DELLE FASI DEL PROGETTO**

* Project Planning
* Stesura dei requisiti
* Design
* Implementazione
* Testing

**STIMA DEL TEMPO PER OGNI FASE**

* Pianificazione: tre giorni.

-Pianificazione attività

-Gantt chart

* Requisiti: sette giorni.

-Use Cases Definition

-Use Cases Diagrams

-Interface Mock-up

-Traceability Matrix

* Design: sette giorni.

-Class Diagram

-Sequence Diagrams

-Activity Diagram

-Updated version of the Traceability Matrix

* Implementazione: cinque giorni.

-System Source Code

-Unit Test Cases con Junit

-Updated version of the Traceability Matrix

* Test: cinque giorni.

-Functional test Cases

-Unit Test Cases

-Test Results

-Updated version of the Traceability Matrix

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

**RISORSE**

**I linguaggi di sviluppo**:

-Java:

La versione utilizzata è JDK 8.

Come IDE sarà utilizzato NetBeans.

Questo linguaggio verrà utilizzato per l’implementazione del codice.

-UML: PlantUML

Come IDE sarà utilizzato NetBeans.

Il linguaggio sarà utilizzato per l’implementazione dei diagrammi.

**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.

//link al github

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

//-tool per mock up

//-Moodle

**Coding Conventions:**

Nel codice si adotterà lo stile CamelCase per le variabili.

Per favorire la leggibilità del codice si seguiranno convenzioni decise anticipatamente nel gruppo di comune accordo. Si aggiungeranno commenti ai metodi, evitando ridondanze, specificandone le finalità.

**Risorse umane:**

Emmanuele Iovino, Miriana Molaro, Daniele Miranda, Caterina Nugnes.

**//Organizzazione risorse umane:**

**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. Mente aperta al cambiamento.

2. Impiegare risorse per la formazione all’utilizzo dei vari tools.

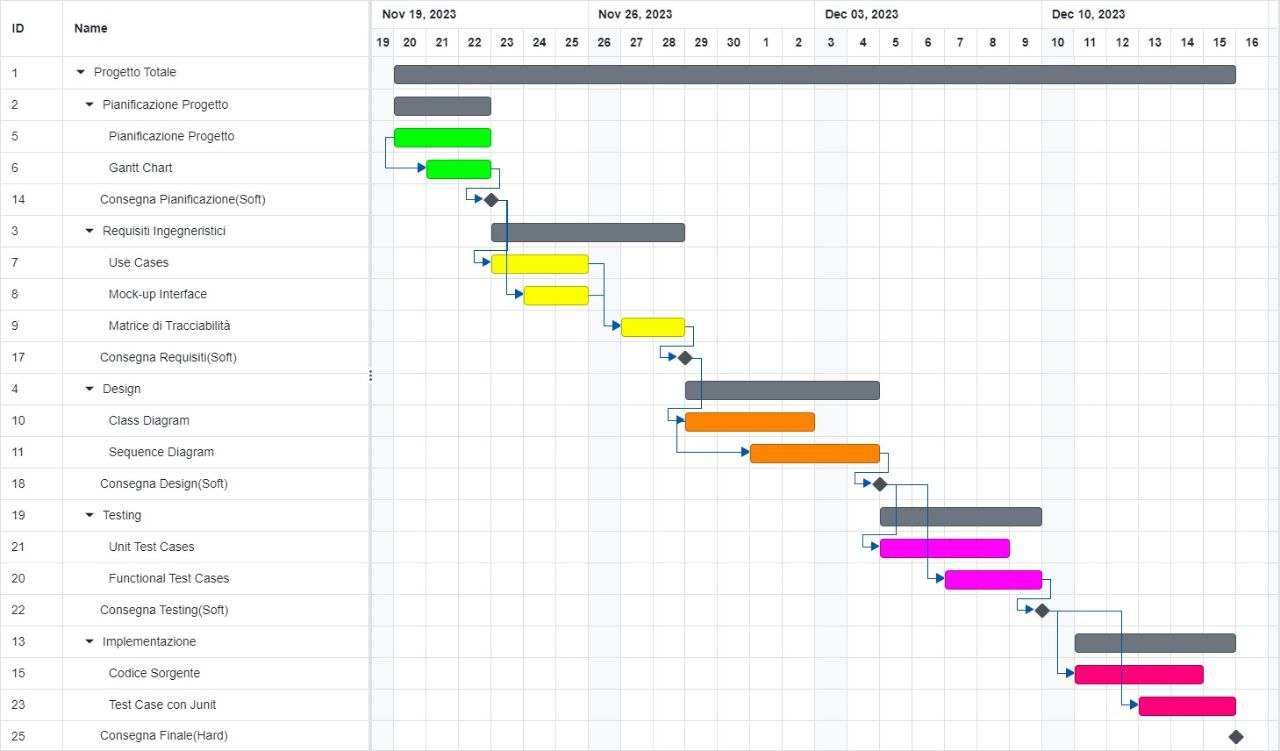
3. Focalizzazione sulla manutenibilità del codice con l’obiettivo di fronteggiare gli imprevisti.

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.

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

**DIAGRAMMA DI GANTT**



Pianificazione: dal 20 al 22 novembre

Requisiti: dal 23 al 29 Novembre.

Design: dal 30 novembre al 6 dicembre.

Implementazione: dal 7 al 10 dicembre.

Testing: dall’11 al 15 dicembre.

Domande:

1. Stesura della suddivisione dei lavori
2. Festività
3. Testing e implementazione: chi va prima?