



# Norme di Progetto

*Gruppo SWEetBIT — Progetto SWEDesigner*

## Informazioni sul documento

<b>Versione</b>	1.0.0
<b>Redazione</b>	Salvatore Pilò
<b>Verifica</b>	Da inserire
<b>Approvazione</b>	Da inserire
<b>Uso</b>	Interno
<b>Distribuzione</b>	Prof. Tullio Vardanega Prof. Riccardo Cardin Gruppo SWEetBITSWEetBIT

## Descrizione

Documento riguardante le norme stabilite dal gruppo SWEetBIT per la realizzazione di SWEDesigner

## Indice

<b>1</b>	<b>Introduzione</b>	<b>4</b>
1.1	Scopo del documento . . . . .	4
1.2	Scopo del Prodotto . . . . .	4
1.3	Glossario . . . . .	4
1.4	Riferimenti . . . . .	4
1.4.1	Informativi . . . . .	4
1.4.2	Normativi . . . . .	5
<b>2</b>	<b>Comunicazioni</b>	<b>5</b>
2.1	Comunicazioni esterne . . . . .	5
2.2	Comunicazioni interne . . . . .	6
2.3	Composizione e-mail . . . . .	6
2.3.1	Destinatario . . . . .	6
2.3.2	Mittente . . . . .	6
2.3.3	Oggetto . . . . .	6
2.3.4	Corpo . . . . .	6
2.3.5	Allegati . . . . .	7
<b>3</b>	<b>Riunioni</b>	<b>7</b>
3.1	Frequenza . . . . .	7
3.2	Convocazione riunione . . . . .	7
3.2.1	Interna . . . . .	7
3.2.2	Esterna . . . . .	8
3.3	Svolgimento riunione . . . . .	8
3.4	Verbale . . . . .	8
3.4.1	Riunione interna . . . . .	8
3.4.2	Esterna . . . . .	8
<b>4</b>	<b>Documenti</b>	<b>9</b>
4.1	Template . . . . .	9
4.2	Struttura del documento . . . . .	9
4.2.1	Prima pagina . . . . .	9
4.2.2	Diario delle modifiche . . . . .	9
4.2.3	Indici . . . . .	10
4.2.4	Formattazione generale delle pagine . . . . .	10
4.3	Norme tipografiche . . . . .	10
4.3.1	Punteggiatura . . . . .	10
4.3.2	Stile di testo . . . . .	11
4.3.3	Composizione del testo . . . . .	11
4.3.4	Formati . . . . .	12
4.3.5	Sigle . . . . .	12

4.4	Componenti grafiche . . . . .	13
4.4.1	Tabelle . . . . .	13
4.4.2	Immagini . . . . .	13
4.5	Classificazione dei documenti . . . . .	13
4.5.1	Documenti formali . . . . .	13
4.5.2	Documenti informali . . . . .	13
4.5.3	Versionamento . . . . .	14
4.6	Ciclo di vita . . . . .	14
<b>5</b>	<b>Analisi dei requisiti</b>	<b>14</b>
5.1	Studio di Fattibilità e Analisi dei Rischi . . . . .	14
5.2	Analisi dei requisiti . . . . .	15
5.2.1	Classificazione dei requisiti . . . . .	15
5.2.2	Modellazione concettuale del sistema e Allocazione . . . . .	16
<b>6</b>	<b>Codifica dei file e documentazione</b>	<b>17</b>
6.1	Codifica e convenzioni . . . . .	17
6.2	Documentazione . . . . .	17
<b>7</b>	<b>Glossario</b>	<b>18</b>
<b>8</b>	<b>Protocollo per lo sviluppo dell'applicazione</b>	<b>18</b>
8.1	Creazione di un nuovo progetto . . . . .	18
8.2	Creazione ticket . . . . .	19
8.2.1	Ticket di pianificazione . . . . .	19
8.2.2	Ticket di realizzazione e controllo . . . . .	20
8.2.3	Ticket di verifica . . . . .	21
8.2.4	Dipendenze temporali . . . . .	21
8.3	Aggiornamento ticket . . . . .	22
<b>9</b>	<b>Ambiente di lavoro</b>	<b>22</b>
9.1	Coordinamento . . . . .	22
9.1.1	Repository . . . . .	22
9.1.2	Gestione del progetto . . . . .	22
9.1.3	Versionamento . . . . .	23
9.1.4	Google Drive . . . . .	23
9.2	Ambiente documentale . . . . .	23
9.2.1	Pianificazione . . . . .	23
9.2.2	Stesura documenti . . . . .	24
9.2.2.1	L <sup>A</sup> T <sub>E</sub> X . . . . .	24
9.2.2.2	Strumentazione esterna . . . . .	24
9.2.2.3	Gestione Use Case e Requisiti . . . . .	24
9.2.2.4	Grafici UML . . . . .	24
9.3	Ambiente di verifica . . . . .	24

9.3.1 Documenti . . . . .	25
---------------------------	----

## 1 Introduzione

### 1.1 Scopo del documento

In questo documento sono definite le norme che i membri del gruppo SWEetBIT adotteranno durante lo sviluppo del progetto SWEDesigner. Tutti i membri sono tenuti a leggere e seguire le norme per migliorare l'uniformità del materiale prodotto, migliorare l'efficienza e ridurre il numero di errori oltre che, ovviamente, evitarli il più possibile. In particolare verranno definite norme riguardanti:

- Interazioni fra membri del gruppo.
- Stesura e convenzioni dei documenti.
- Modalità di lavoro durante le fasi di sviluppo del progetto.
- Ambiente di lavoro.

### 1.2 Scopo del Prodotto

Lo scopo del progetto è la realizzazione di una *Web App<sub>G</sub>* che fornisca all'utente un *UML<sub>G</sub> Designer<sub>G</sub>* con il quale riuscire a disegnare correttamente diagrammi delle classi e descrivere il comportamento dei metodi interni alle stesse attraverso l'utilizzo di -da decidere il tipo di schema-. La *Web App<sub>G</sub>* permetterà all'utente di generare codice Java o Javascript dal diagramma disegnato ed eventualmente andare a ritoccarne il risultato al fine di ottenere un codice eseguibile, funzionante e funzionale.

### 1.3 Glossario

Con lo scopo di evitare ambiguità di linguaggio e di massimizzare la comprensione dei documenti, il gruppo ha steso un documento interno che è il *Glossario v1.0.0*. In esso saranno definiti, in modo chiaro e conciso i termini che possono causare ambiguità o incomprensione del testo.

### 1.4 Riferimenti

#### 1.4.1 Informativi

- **Specifiche UTF-8:**  
[http://unicode.org/faq/utf\\_bom.html](http://unicode.org/faq/utf_bom.html)

- **ISO 8601:2004:**  
<https://www.iso.org/standard/40874.html>
- **Licenza MIT:** <https://opensource.org/licenses/MIT>
- **GitHUB:**  
<https://github.com/>
- **UML:**  
<http://www.uml.org/>
- **Atom:**  
<https://atom.io/>
- **TexLive:**  
<https://www.tug.org/texlive/>
- **L<sup>A</sup>T<sub>E</sub>X**  
<https://www.latex-project.org/>
- **Telegram:**  
<https://telegram.org/>
- INSERIRE ALTRI
- **Piano di progetto:** *Piano di progetto v1.0.0*
- **Piano di qualifica:** *Piano di qualifica v1.0.0*

#### 1.4.2 Normativi

- **Capitolato di appalto SWEDesigner (C6):**  
<http://www.math.unipd.it/~tullio/IS-1/2016/Progetto/C6.pdf>

## 2 Comunicazioni

### 2.1 Comunicazioni esterne

Per le comunicazioni esterne è stata creata un'apposita casella di posta elettronica:

[sweet.bit.group@gmail.com](mailto:sweet.bit.group@gmail.com)

Tale indirizzo deve essere l'unico canale di comunicazione esistente tra il gruppo di lavoro e l'esterno.

L'unico membro del gruppo ad avere accesso alla mail, e quindi alle comunicazioni con il committente, è il *Responsabile di Progetto*. Suo anche il compito di informare i membri del gruppo delle discussioni avvenute tramite la casella di posta elettronica e inoltrare, qualora dovesse ritenerlo necessario, tali discussioni alle caselle postali dei vari membri del gruppo.

## **2.2 Comunicazioni interne**

Per tutte le comunicazioni interne è stato creato un gruppo **Telegram** in cui ogni membro può comunicare i propri impegni e il proprio stato dei lavori.

Il gruppo serve, inoltre, per la comunicazione delle date delle riunioni, per comunicare tempestivamente le decisioni prese e per scambiare materiale utile al progetto.

## **2.3 Composizione e-mail**

### **2.3.1 Destinatario**

Il destinatario delle mail esterne può variare a seconda che ci si debba riferire al *committente<sub>G</sub>* del progetto, al Prof. Tullio Vardanega o al Prof. Cardin Riccardo.

### **2.3.2 Mittente**

L'unico indirizzo utilizzabile è [sweet.bit.group@gmail.com](mailto:sweet.bit.group@gmail.com) e deve essere usato solamente dal *Responsabile di Progetto*.

### **2.3.3 Oggetto**

L'oggetto di una comunicazione deve essere chiaro, stringato e possibilmente diverso da altri soggetti preesistenti.

Nel caso in cui si dovesse comporre un messaggio di risposta vi è l'obbligo di utilizzare la particella "RE:" prima dell'oggetto in modo tale da identificare il livello di risposta; Qualora si trattasse invece di un inoltrato è obbligatorio utilizzare la particella "I:".

In ogni caso l'oggetto di una comunicazione già avviata non potrà essere cambiato.

### **2.3.4 Corpo**

Il corpo di un messaggio deve contenere tutte le informazioni necessarie a rendere facilmente comprensibile l'argomento trattato a tutti i destinatari. Se alcune parti del messaggio hanno uno o più destinatari specifici, il loro nome dovrà essere aggiunto all'inizio del paragrafo tramite la seguente segnatura: *@destinatario*. In caso di risposta od

inoltre del messaggio, il contenuto aggiunto deve essere sempre messo in testa (per non costringere gli altri membri a dover scorrere tutta la mail). È caldamente consigliato, inoltre, la non cancellazione delle precedenti parti del messaggio in modo tale da rendere ogni partecipante alla discussione consapevole del contesto.

### 2.3.5 Allegati

L'uso di allegati è permesso qualora si ritenga necessario.

Un esempio è il resoconto di un incontro con il *proponente<sub>G</sub>* o il *committente<sub>G</sub>*.

## 3 Riunioni

### 3.1 Frequenza

Tutte le riunioni interne, salvo casi eccezionali, si svolgeranno settimanalmente mentre tutte quelle esterne saranno convocate solo qualora se ne sentisse il bisogno.

### 3.2 Convocazione riunione

#### 3.2.1 Interna

Il *Responsabile di Progetto* ha il compito di convocare le riunioni generali valutando, di volta in volta, la possibilità di anticipare o posticipare la data designata per la riunione del gruppo.

Qualora un membro del gruppo lo ritenesse necessario potrà fare richiesta, attraverso il gruppo **Telegram**, di una riunione extra.

È auspicabile, infine, che diversi membri del gruppo possano organizzarsi fra di loro per svolgere alcuni compiti che non richiedono la presenza del gruppo di lavoro al completo. Ad esempio è interessante e utile la collaborazione fra *Progettista* e *Analista* senza che vengano coinvolte altre persone esterne ai compiti da loro svolti.

Il responsabile avviserà tutti i membri del gruppo attraverso un messaggio sul gruppo **Telegram** che verrà fissato in alto e conterrà luogo, data ed ora della riunione.

Ogni membro del gruppo è tenuto a confermare o meno la sua presenza nelle 24h successive. Il responsabile è tenuto ad avvertire telefonicamente tutti i membri che non hanno ancora risposto al messaggio.

Ogni cambiamento nell'orario di convocazione deve essere comunicato per tempo dal responsabile attraverso le modalità sopra elencate.



### 3.2.2 Esterna

Concordando con gli altri membri del gruppo la necessità di effettuare una riunione con il *proponente<sub>G</sub>* o con il *committente<sub>G</sub>*, il *Responsabile di Progetto* si metterà in contatto con i diretti interessati per fissare una data che metta tutti quanti d'accordo.

## 3.3 Svolgimento riunione

All'apertura della riunione, verificata la presenza dei membri previsti, viene scelto un segretario che avrà il compito di annotare ogni argomento trattato e di redigere il verbale dell'assemblea, che dovrà poi essere inviato ai restanti componenti del gruppo.

Tutti i partecipanti devono osservare un comportamento consono al miglior svolgimento della riunione e al raggiungimento degli obiettivi della stessa. Il segretario deve inoltre controllare che venga seguito l'ordine del giorno in modo da non tralasciare alcun punto.

## 3.4 Verbale

### 3.4.1 Riunione interna

Il verbale di una riunione interna è un documento informale che traccia semplicemente tutti gli argomenti trattati all'interno della riunione.

Verrà redatto dal segretario della riunione, ruolo scelto di volta in volta e a rotazione fra i presenti, e dovrà essere condiviso attraverso il gruppo **Telegram** per essere a disposizione, in qualsiasi momento (**Telegram** offre la possibilità di tracciare istantaneamente i media condivisi) da ogni membro del gruppo.

Il verbale dovrà essere inoltre inviato via e-mail ad ogni componente del gruppo il quale avrà cura di mantenerlo localmente al fine di avere sempre a disposizione gli argomenti trattati nel corso di una riunione.

### 3.4.2 Esterna

Il verbale generato da una riunione esterna con il committente o il proponente è un documento ufficiale che può assumere il valore di normativo quindi deve essere redatto seguendo dei criteri specifici.

Per agevolarne la scrittura è stato creato un template  $\text{\LaTeX}$  che ne definisce la struttura. Vi è, ovviamente, l'obbligo di seguire tale schema per la stesura del verbale di una riunione esterna che dovrà poi essere inviato a tutti i membri del gruppo seguendo le stesse regole del verbale per una riunione interna.

## 4 Documenti

Questo capitolo descrive tutte le convenzioni scelte ed adottate da SWEetBIT riguardo alla stesura, verifica e approvazione della documentazione da produrre.

### 4.1 Template

Per agevolare la redazione di un documento è stato prodotto un template e delle regole da seguire per la stesura degli stessi.

Tale modello e tali regole sono inseriti all'interno di una cartella `documents\template` sulla *repository<sub>G</sub>*

### 4.2 Struttura del documento

#### 4.2.1 Prima pagina

Ogni documento è caratterizzato da una prima pagina che contiene le seguenti informazioni sul documento:

- Nome del gruppo;
- Nome del progetto;
- Logo del gruppo;
- Titolo del documento;
- Versione del documento;
- Cognome e nome dei redattori del documento;
- Cognome e nome dei verificatori del documento;
- Cognome e nome del responsabile approvatore del documento;
- Destinazione d'uso del documento;
- Lista di distribuzione del documento;
- Breve descrizione del documento;

#### 4.2.2 Diario delle modifiche

La seconda pagina di ogni documento contiene il diario delle modifiche del documento. Ogni riga del diario delle modifiche contiene:

- Un breve sommario delle modifiche svolte;
- Cognome e nome dell'autore;
- Data della modifica;
- Versione del documento dopo la modifica;

La tabella è ordinata per data in ordine decrescente, in modo che la prima riga corrisponda alla versione attuale del documento.

#### 4.2.3 Indici

In ogni documento è presente un indice delle sezioni, un indice delle figure e un indice delle tabelle. Nel caso non siano presenti figure o tabelle i rispettivi indici verranno omessi.

#### 4.2.4 Formattazione generale delle pagine

L'intestazione di ogni pagina contiene:

- Logo del gruppo;
- Nome del gruppo;
- Nome del progetto;
- Sezione corrente del documento;

A piè di pagina invece è presente:

- Nome e versione del documento;
- Pagina corrente nel formato  $N$  di  $T$  dove  $N$  è il numero di pagina corrente e  $T$  è il numero di pagine totali.

### 4.3 Norme tipografiche

Questa sezione racchiude le convenzioni riguardanti tipografia, ortografia e uno stile uniforme per tutti i documenti.

#### 4.3.1 Punteggiatura

- **Parentesi:** Il testo racchiuso tra parentesi non deve aprirsi o chiudersi con un carattere di spaziatura e non deve terminare con un carattere di punteggiatura;

- **Punteggiatura:** Un carattere di punteggiatura non deve mai seguire un carattere di spaziatura;
- **Lettere maiuscole:** Le lettere maiuscole vanno poste solo dopo il punto, il punto di domanda, il punto esclamativo e all'inizio di ogni elemento di un elenco puntato, oltre che dove previsto dalla lingua italiana. È inoltre utilizzata l'iniziale maiuscola nel nome del team, del progetto, dei documenti, dei ruoli di progetto, delle fasi di lavoro e nelle parole Proponente e Committente.

#### 4.3.2 Stile di testo

- **Corsivo:** Il corsivo deve essere utilizzato nei seguenti casi:
  - **Citazioni:** Quando si deve citare una frase questa sarà scritta in corsivo;
  - **Nomi particolari:** Il corsivo deve essere utilizzato quando ci si riferisce a figure particolari (es. *Analista*);
  - **Documenti:** Il corsivo deve essere utilizzato quando ci si riferisce a documenti particolari (es. *Glossario*);
  - **Altri casi:** Il corsivo sarà utilizzato in tutte quelle situazioni in cui è necessario dare rilievo ad una parola o passaggio significativo;
- **Grassetto:** Il grassetto deve essere utilizzato nei seguenti casi:
  - **Elenchi puntati:** In questo caso il grassetto può essere utilizzato per mettere in evidenza i punti sviluppati nella loro continuazione;
  - **Altri casi:** Il grassetto dovrà essere sempre utilizzato per evidenziare passaggi o parole chiave;
- **\path:** Il comando `\path` deve essere utilizzato per indicare i percorsi all'interno di directory;
- **Maiuscolo:** L'utilizzo di parole completamente in maiuscolo è riservato solo ed esclusivamente alle sigle o alle macro `\LaTeX` riportate nei documenti;
- **\LaTeX:** Ogni riferimento a `\LaTeX` deve essere scritto utilizzando la macro `\LaTeX`;

#### 4.3.3 Composizione del testo

- **Elenchi puntati:** Ogni punto dell'elenco puntato deve essere scritto in grassetto e con la prima lettera in maiuscolo.  
Nella definizione del punto la prima lettera dovrà essere maiuscola ad eccezione di casi isolati (es. nome di file) e dovrà terminare sempre con un ";" ;

- **Note a piè di pagina:** Ogni nota dovrà cominciare con l'iniziale della prima parola maiuscola e non deve essere preceduta da alcun carattere di spaziatura. Ogni nota deve terminare con un punto.

#### 4.3.4 Formati

- **Percorsi:** Per tutti gli indirizzi e-mail e web completi dovrà essere utilizzato il comando `\url` mentre per i percorsi relativi si utilizzerà il comando `\path`;
- **Date:** Tutte le date presenti all'interno della documentazione devono seguire la notazione definiti nello standard *ISO<sub>G</sub> 8601:2004*:

AAAA-MM-GG

dove:

- AAAA: rappresenta l'anno utilizzando quattro cifre;
- MM: rappresenta il mese utilizzando due cifre;
- GG: rappresenta il giorno utilizzando due cifre;
- **Nomi propri:** l'utilizzo dei nomi propri dei membri del team (e non) deve seguire la notazione "Cognome Nome";
- **Nome gruppo:** ci si riferirà al gruppo solo come "SWEetBIT";
- **Nome del proponente:** ci si riferirà al proponente come "Zucchetti s.r.l" o semplicemente come "Proponente";
- **Nome del committente:** ci si riferirà al committente come "prof. Vardanega Tullio" o semplicemente come "Committente";
- **Nome del progetto:** ci si riferirà al progetto solo come "SWEDesigner";

#### 4.3.5 Sigle

Le sigle dei documenti potranno essere utilizzate solo ed esclusivamente all'interno di tabelle o diagrammi. Sono previste le seguenti sigle:

- **AdR** = Analisi dei Requisiti;
- **GL** = Glossario;
- **NdP** = Norme di Progetto;
- **PdP** = Piano di Progetto;
- **PdQ** = Piano di Qualifica;

- **SdF** = Studio di Fattibilità;
- **ST** = Specifica Tecnica;
- **RA** = Revisione d'Accettazione;
- **RP** = Revisione di Progettazione;
- **RQ** = Revisione di Qualifica;
- **RR** = Revisione dei Requisiti;

## 4.4 Componenti grafiche

### 4.4.1 Tabelle

Ogni tabella presente all'interno dei documenti dev'essere accompagnata da una didascalia, in cui deve comparire un numero identificativo incrementale per la tracciabilità della stessa all'interno del documento.

### 4.4.2 Immagini

Le immagini da includere all'interno del documento devono avere preferibilmente il formato Portable Network Graphics (*PNG<sub>G</sub>*).

## 4.5 Classificazione dei documenti

### 4.5.1 Documenti formali

Un documento viene definito formale quando viene approvato dal *Responsabile di Progetto* ed è quindi pronto per essere inviato ai richiedenti.

Per raggiungere questo stato il documento deve seguire l'iter descritto nel *Piano di Qualifica* e nel paragrafo 4.6 riguardante il ciclo di vita dei documenti.

### 4.5.2 Documenti informali

Un documento è definito informale fino a quando non approvato dal *Responsabile di Progetto*, fino ad allora il suo uso è da considerarsi unicamente interno.

### 4.5.3 Versionamento

La documentazione prodotta deve essere corredata dal numero di versione attuale utilizzando la codifica:

$$v.X.Y.Z$$

dove:

- X: indica il numero crescente di uscite formali del documento;
- Y: indica il numero crescente di modifiche sostanziali al documento;
- Z: indica il numero crescente di modifiche minori apportate al documento;

## 4.6 Ciclo di vita

Ogni documento prodotto segue un preciso iter che scandisce le fasi in cui si trova in ogni istante. Un documento può trovarsi in tre stati diversi:

- **In lavorazione:** Un documento entra in questa fase nel momento della sua creazione e vi rimane per tutto il periodo della sua stesura o per eventuali successive modifiche;
- **Da verificare:** Un documento entra in questa fase alla fine della sua stesura quando entra in possesso dei verificatori che avranno il compito di individuare e correggere eventuali errori sintattici o semantici;
- **Approvato:** Un documento entra in questa fase una volta che il *Responsabile di Progetto* lo ha approvato dopo la fase di verifica.  
L'approvazione sancisce la fine del ciclo di vita del documento per la data versione;

Ogni fase del ciclo di vita può essere affrontata anche più volte da parte di un documento.

## 5 Analisi dei requisiti

### 5.1 Studio di Fattibilità e Analisi dei Rischi

Alla pubblicazione dei capitolati è compito del *Responsabile di Progetto* convocare un numero di riunioni tale da consentire al gruppo un confronto su tutti i capitolati disponibili.

Tali riunioni saranno d'aiuto agli *Analisti* per farsi un'idea delle conoscenze e preferenze di ogni membro del gruppo così da poter redigere uno *Studio di Fattibilità* dei capitolati disponibili basandosi su:

- **Dominio tecnologico e applicativo:** conoscenza delle tecnologie richieste, esperienze precedenti con le problematiche poste dal capitolato, conoscenza del dominio applicativo;
- **Rapporto Costi/Benefici:** competitori e prodotti simili già presenti sul mercato, quantità di requisiti obbligatori, costo della realizzazione rapportato al risultato previsto;
- **Individuazione dei rischi:** Comprensione dei punti critici della realizzazione, individuazione di eventuali lacune tecniche o di conoscenza del dominio applicativo dei membri del gruppo, analisi delle difficoltà nell'individuazione dei requisiti e loro verificabilità;

Un'ulteriore riunione a *Studio di Fattibilità* concluso determinerà la scelta del capitolato.

## 5.2 Analisi dei requisiti

La stesura del documento di *Analisi dei Requisiti* è compito degli *Analisti* e si divide nelle fasi di seguito riportante.

Il documento dovrà seguire, inoltre, le norme specificate di seguito.

### 5.2.1 Classificazione dei requisiti

È compito degli *Analisti* stilare una lista dei requisiti emersi dal capitolato e da eventuali riunioni con il Proponente. Questi dovranno essere classificati per tipo e per importanza utilizzando la seguente codifica:

R[importanza][tipo][codice]

- **Importanza** può assumere i seguenti valori:
  - 0 : Requisito obbligatorio;
  - 1 : Requisito desiderabile;
  - 2 : Requisito opzionale;
- **Tipo** può assumere i seguenti valori:
  - F : Funzionale;
  - Q : Di Qualità;
  - P : Prestazionale;
  - V : Vincolo;
- **Codice** è il codice univoco di ogni requisito espresso in modo gerarchico.



Ogni requisito è poi esplicito nel seguente modo:

- **Relazioni** di dipendenza con altri requisiti;
- **Descrizione** sintetica del requisito;

### 5.2.2 Modellazione concettuale del sistema e Allocazione

Successivamente al riconoscimento e definizione del requisiti emersi dal capitolato si procede all'analisi dei casi d'uso, denominati anche come *use case* o con l'acronimo UC. È richiesta agli analisti l'identificazione dei vari casi d'uso, procedendo dal generale al particolare che verranno inseriti nel software di tracciamento *Trender<sub>G</sub>*.

Per ogni UC è richiesto l'inserimento, all'interno del software, di:

- **Titolo:** Nome del Caso d'Uso;
- **Descrizione:** Descrizione breve e concisa dell'UC;
- **Precondizione:** Condizione d'accesso al Caso d'Uso;
- **Postcondizione:** Condizione d'uscita al Caso d'Uso;

Gli altri campi da compilare, da non ritenere obbligatori ma desiderabili, sono:

- **Padre:** Indicare codice univoco del Caso d'Uso padre;
- **Tipo:** Può essere di tre tipi differenti:
  - **Inclusione**
  - **Estensione**
  - **Gerarchia**
- **Scenario:** Descrizione dello scenario rappresentato;
- **Scenario alternativo:** Descrizione scenari alternativi, se presenti;
- **Percorso immagine:** Il percorso dell'immagine rappresentate l'UML;
- **Descrizione immagine:** Descrizione breve e sommaria dell'immagine;

Una volta creato l'UC, selezionarlo dalla lista per:

- Modificare i campi dati inseriti;
- Osservare i figli dell'UC corrente;
- Associare un **Attore** selezionandolo dal menù a tendina;
- Associare un **Requisito**;

Il caso d'uso dovrà essere accompagnato da un grafico riassuntivo in  $UML_G2.x$ , titolato come il caso d'uso in questione.

È compito del software di tracciamento tracciare gli UC con un codice univoco e gerarchico nella forma:

UC[codice univoco del padre].[codice univoco del figlio]

Il software provvederà a generare anche i file  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ corrispondenti.

## 6 Codifica dei file e documentazione

### 6.1 Codifica e convenzioni

Tutti i file contenenti codice o documentazione dovranno essere in codifica  $UTF-8_G$  senza  $BOM_G$ .

È ammessa la possibilità di effettuare modifiche alle convenzioni stabilite in seguito ad una decisione del *Responsabile di Progetto*.

L'unica lingua ammessa per i nomi di variabili, classi e funzioni è l'inglese.

### 6.2 Documentazione

I file contenenti codice dovranno essere provvisti di un'intestazione contenente:

```
/*!  
* \file Nome del file  
* \author Autore (indirizzo e-mail dell'autore)  
* \date Data di creazione  
* \brief Breve descrizione del file  
*  
* Descrizione dettagliata del file  
*/
```

Prima di ogni classe dovrà esserci un commento contenente:

```
/*!  
* \class Nome della classe  
* \brief Breve descrizione della classe  
*/
```

Prima di ogni metodo dovrà essere inserito un commento contenente:

```
/*!  
* \brief Breve descrizione della funzione  
* \param Nome del primo parametro  
* \param Nome del secondo parametro  
* \return Valore ritornato dalla funzione  
*/
```

## 7 Glossario

Il Glossario conterrà tutte le parole presenti negli altri documenti che fanno parte del contesto dell'applicazione o che possono essere fraintese. Le definizioni, presentate in ordine alfabetico, dovranno essere concise e comprensibili.

I termini verranno inseriti nel glossario parallelamente al processo di stesura degli altri documenti, in modo da limitare l'errore umano.

È preferibile inserire un termine inizialmente privo di definizione, piuttosto che rimandare la stesura del glossario.

## 8 Protocollo per lo sviluppo dell'applicazione

Per procedere con una stesura controllata dei documenti e con uno sviluppo controllato del codice si è scelto di adottare il sistema di ticketing **Redmine**.

La scelta di tale piattaforma è spiegata all'interno del capitolo 9. In questa sezione si faranno molti riferimenti impliciti al *Piano di Progetto* e al *Piano di Qualifica*.

### 8.1 Creazione di un nuovo progetto

La creazione di un progetto è un compito del *Responsabile di Progetto*.

Un nuovo progetto è una macro-attività formata da molte sotto-attività coordinate da un responsabile.

Per la creazione di un nuovo progetto la prassi da seguire è la seguente:

- Aprire **Progetti**;
- Selezionare **Nuovo Progetto**;
- Assegnare un **Nome** breve ma significativo;
- Nel caso in cui è necessario creare un sotto-progetto, indicare il nome del progetto padre nell'omonimo campo;

- **Identificativo:** scrivere in minuscolo ed indicare il nome della fase a cui si riferisce (es. SdF-rr);

## 8.2 Creazione ticket

I ticket vengono creati da:

- **Responsabile di Progetto:** crea i ticket più importanti che rappresentano le macro-fasi evidenziate all'interno della pianificazione;
- **Responsabile di Sotto-progetto:** crea i ticket per i processi non pianificati inizialmente ma che si rivelano necessari per l'avanzamento del sotto-progetto assegnato;
- **Verificatore:** crea i ticket per segnalare errori emersi durante il processo di verifica;

I ticket possono essere di tre categorie:

- **Ticket di pianificazione:** rappresentano le macro-attività di maggiore importanza e sono organizzati in una gerarchia basata sul livello di importanza. Tali attività vengono create da:
  - *Responsabile di Progetto* che durante la pianificazione individua le attività più importanti da svolgere;
  - *Responsabile di Sotto-progetto* che durante lo svolgimento dell'attività principale può scomporla in sotto-problemi;
- **Ticket di realizzazione e controllo:** ogni documento, durante la sua stesura, passa attraverso due stadi:
  - **Realizzazione:** un redattore realizzerà la prima stesura dell'intero documento;
  - **Controllo:** un redattore, diverso dal precedente, eseguirà un primo controllo di tutta la parte scritta;
- **Ticket di verifica:** rappresentano gli errori evidenziati dai *Verificatori* durante l'operazione di controllo dell'intero documento;

### 8.2.1 Ticket di pianificazione

- Selezionare **Nuova segnalazione** dal menù principale;
- **Tracker:** indicare la natura del ticket:
  - **Documento:** attività legata alla stesura di un documento;

- **Codifica:** attività legata alla codifica del software;
- **Verifica:** macro-attività legata alla verifica del prodotto delle macro-attività;
- **Oggetto:** descrizione breve e significativa della natura del ticket;
- **Descrizione:** descrizione comprensibile dell'attività da svolgere;
- **Stato:** Plan;
- **Attività principale:** se si vuole identificare una **sotto-attività** indicare l'id del ticket padre;
- **Categoria:**  $PDCA_G$  se e solo se il ticket viene generato dal *Responsabile di Progetto*;
- **Assegnato a:** indicare il nome del responsabile;
- **Osservatori:** aggiungere eventuali collaboratori;

### 8.2.2 Ticket di realizzazione e controllo

- Selezionare **Nuova segnalazione** dal menù principale;
- **Tracker:** indicare la natura del ticket:
  - **Documento:** attività legata alla stesura di un documento;
  - **Codifica:** attività legata alla codifica del software;
  - **Verifica:** macro-attività legata alla verifica del prodotto delle macro-attività;
- **Oggetto:** descrizione breve e significativa della natura del ticket;
- **Descrizione:** descrizione comprensibile dell'attività da svolgere;
- **Stato:** New;
- **Attività principale:** se si vuole identificare una **sotto-attività** indicare l'id del ticket padre;
- **Inizio:** dare una presunta data di inizio;
- **Scadenza:** dare una presunta data di fine;
- **Assegnato a:** indicare il nome del responsabile;
- **Osservatori:** aggiungere eventuali collaboratori;

### 8.2.3 Ticket di verifica

Un *Verificatore* per creare un *ticket di verifica* deve:

1. Assicurarsi che esista all'interno del progetto l'attività *Verifica*.  
Su questa attività devono essere presenti due sotto-attività: *Verifica - realizzazione* e *Verifica - approvazione*.  
Tutti i ticket devono essere creati come sotto-attività di *Verifica - realizzazione*.
2. Creare il ticket seguendo le seguenti direttive:
  - Selezionare **Nuova segnalazione** dal menù principale;
  - **Tracker:** Bug;
  - **Oggetto:** descrizione breve e significativa della natura del ticket;
  - **Descrizione:** descrizione comprensibile dell'attività da svolgere;
  - **Stato:** New;
  - **Attività principale:** se si vuole identificare una **sotto-attività** indicare l'id del ticket padre;
  - **Assegnato a:** indicare il nome del responsabile;

Tutti i campi non segnalati sono da lasciarsi vuoti.

Il compito di assegnare la correzione dell'errore è dato al Responsabile del progetto padre.

### 8.2.4 Dipendenze temporali

Dopo la creazione dei ticket è necessario assegnare le **dipendenze temporali** fra gli stessi.

La procedura da seguire è la seguente:

- Spostarsi su **Segnalazioni**;
- Aprire il link alla segnalazione a cui aggiungere la dipendenza;
- Nella sezione **Segnalazioni correlate** premere su **Aggiungi**;
- Scegliere **segue** ed indicare il numero della segnalazione bloccante con eventuali giorni di slack;

Tutti i campi non segnalati sono da lasciarsi vuoti.

### 8.3 Aggiornamento ticket

L'aggiornamento dei ticket avviene tramite il cambiamento del loro stato da:

- **In Progress:** segnala che uno o più membri del gruppo stanno lavorando al completamento di quel ticket.  
In questo caso la percentuale di completamento deve essere compresa fra 0% e 90%;
- **Closed:** segnala che l'attività è stata conclusa.  
La percentuale di completamento in questo caso è fissata a 100%;

## 9 Ambiente di lavoro

### 9.1 Coordinamento

Il coordinamento del gruppo avviene tramite:

- Repository su *GitHub<sub>G</sub>*;
- *Google Drive<sub>G</sub>*;
- *Telegram<sub>G</sub>*

#### 9.1.1 Repository

Sulla repository di GitHub, raggiungibile all'indirizzo <https://github.com/SWEetBIT>, sono caricati i vari template da utilizzare durante la stesura dei documenti e la strumentazione da utilizzare per la formattazione dei termini del *Glossario*.

#### 9.1.2 Gestione del progetto

La piattaforma che è stata scelta per la gestione del progetto è **Redmine** che fornisce:

- Un sistema flessibile di gestione dei ticket;
- Il grafico Gantt delle attività;
- Un calendario per l'organizzazione e la distribuzione dei compiti;
- La visualizzazione della repository relativa al progetto;
- Un sistema di rendicontazione del tempo;

Sono state analizzate altre alternative a **Redmine** che, dopo una fase di analisi iniziale, non sono risultate idonee allo scopo:

- **Teamworks:** si tratta del software probabilmente più adatto per il project management vista la sua grande versatilità e la strumentazione offerta. Purtroppo i suoi costi non hanno permesso un suo utilizzo in ambito universitario;
- **Zohoo:** A differenza di Redimine o di Teamworks, questa piattaforma non offre un servizio di rendicontazione del tempo e la generazione di grafici Gantt;

### 9.1.3 Versionamento

Dopo aver preso in considerazione diverse opzioni per il versionamento ma alla fine si è scelto di utilizzare **GitHub** per via della sua enorme flessibilità e per via delle esperienze pregresse di tutti i membri del gruppo che hanno manifestato una certa familiarità con lo strumento.

È stata creata una sola repository, alla quale si aggiungeranno le altre legate alle fasi successive, contenente tutte le cartelle necessarie alla stesura dei documenti  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ .

Una volta terminato il lavoro di redazione dei documenti sarà creato un *branch* di verifica per permettere ai *Verificatori* di lavorare in parallelo agli altri membri del gruppo.

### 9.1.4 Google Drive

Lo strumento di cloud storage di Google è stato utilizzato principalmente per tenere traccia di verbali interni e di documentazione interna non formale che può essere utile a tutti i membri del gruppo durante le varie fasi di lavorazione del progetto.

## 9.2 Ambiente documentale

### 9.2.1 Pianificazione

Per pianificare le attività legate allo sviluppo del progetto e la gestione delle risorse si è scelto di utilizzare **ProjectLibre**.

Si tratta di un ottimo software open-source basato su Java per il project management.

La scelta è ricaduta su questo software principalmente per quattro motivi:

- Si tratta di un software portabile essendo basato su Java;
- È open-source;
- Genera automaticamente digrammi di Gantt;
- Il salvataggio dei file è in XML, quindi un formato testuale che permette di utilizzare i merge senza causare troppi conflitti;



### 9.2.2 Stesura documenti

**9.2.2.1 L<sup>A</sup>T<sub>E</sub>X** Per la stesura dei documenti si è scelto di utilizzare il sistema L<sup>A</sup>T<sub>E</sub>X poiché permette una netta separazione fra contenuti e formattazione: con L<sup>A</sup>T<sub>E</sub>X è possibile definire i template di layout in file condivisi da ogni documento rendendo la lavorazione degli stessi altamente più flessibile e ottimale.

L<sup>A</sup>T<sub>E</sub>X consente poi l'utilizzo di funzioni e variabili locali definiti dall'utente in maniera tale da semplificare ulteriormente il lavoro di stesura dei documenti una volta definita la struttura degli stessi.

Per la scrittura dei documenti L<sup>A</sup>T<sub>E</sub>X l'editor utilizzato è **TexMaker**.

**9.2.2.2 Strumentazione esterna** Per ridurre al minimo gli errori di calcolo di alcuni indici o di formattazione del testo si è scelto di utilizzare alcuni strumenti automatici che si occupano di svolgere alcuni semplici compiti di calcolo e formattazione:

- TOOL INDICE DI GULPEASE
- GLOSSARIZZAZIONE DEI TERMINI
- **Aspell**: si tratta di un tool offerto dall'editor stesso o presente in forma stand-alone che effettua un controllo ortografico su tutto il documento;

**9.2.2.3 Gestione Use Case e Requisiti** Per semplificare il tracciamento degli Use Case e dei Requisiti si è scelto di utilizzare **Trender**, uno strumento open-source che permette di gestire al meglio entrambi gli elementi e le realizzazioni fra gli stessi.

Il tool permette l'esportazione di questi direttamente in L<sup>A</sup>T<sub>E</sub>X così da semplificare la scrittura degli stessi all'interno della documentazione.

**9.2.2.4 Grafici UML** Per il disegno dei grafici UML si è scelto di utilizzare **Astah Professional** per via del suo enorme potenziale e della mole di strumenti offerti rispetto agli altri editor presenti sul mercato.

Le alternative analizzate sono state Dia, LuchiChart e Papyrus che si sono rivelati però troppo deboli in confronto ad **Astah Professional** che offre diverse funzionalità aggiuntive raggruppandone diverse dei tre software citati al suo interno.

## 9.3 Ambiente di verifica

Vengono qui elencati e sommariamente descritti gli strumenti automatizzati per effettuare la verifica dei documenti redatti e del codice prodotto.

Le metriche ed i metodi per effettuare verifica sono ampiamente e dettagliatamente descritti nel *Piano di Qualifica*.

A tale documento si fa inoltre riferimento per le caratteristiche di fondamentale importanza per la verifica degli strumenti qui riportati.

### 9.3.1 Documenti

- **TexMaker:** Per la stesura dei documenti è stato utilizzato TexMaker per via dell'intrazione con i dizionari di OpenOffice.org che integra che consentono un controllo ortografico in real-time;
- **Aspell:** strumento per la correzione tipografica dei documenti redatti in  $\text{\LaTeX}$ ;