



MyTalk

Norme di Progetto

Informazioni sul documento	
Nome file:	Norme di Progetto
Versione:	1.0
Data creazione:	2012-12-01
Data ultima modifica:	2012-12-04
Stato:	Approvato
Uso:	Interno
Redattori:	Marco Schivo
Approvato da:	Stefano Farronato
Verificatori:	Diego Beraldin

Storia delle modifiche

Versione	Descrizione intervento	Redattore	Data
1.0	Approvazione documento	Stefano Farronato	2012-12-04
0.6	Validazione documento	Diego Beraldin	2012-12-04
0.5	Fine stesura documento	Marco Schivo	2012-12-03
0.4	Stesura sezione Milestones e Ticketing, Analisi dei requisiti	Marco Schivo	2012-12-03
0.3	Stesura sezione ambiente documentale	Marco Schivo	2012-12-02
0.2	Stesura sezione ambiente di lavoro	Marco Schivo	2012-12-01
0.1	Stesura scheletro documento, sezione comunicazione	Marco Schivo	2012-12-01



Indice

1	Introduzione	1
1.1	Scopo del prodotto	1
1.2	Scopo del documento	1
1.3	Ambiguità	1
2	Relazioni interpersonali e comunicazione	2
2.1	Comunicazione interna	2
2.2	Comunicazione esterna	2
2.3	Incontri Interni	2
2.4	Incontri Esterni	2
3	Ambiente di lavoro	3
3.1	Repository	3
3.1.1	Registrazione	3
3.1.2	Installazione	3
3.1.3	Struttura	3
3.1.4	Sistema di versionamento	4
3.2	Sistema di tracciamento dei requisiti	4
3.3	Sistema operativo	5
3.4	Grafici UML	5
3.5	Grafici di Attività	5
4	Ambiente documentale	6
4.1	Software utilizzati	6
4.2	Nomi dei documenti	6
4.3	Impostazioni di base di un documento	6
4.4	Tipi di documento	7
4.5	Ciclo di vita di un documento	8
4.6	Template	9
4.7	Glossario	9
4.8	Norme tipografiche	9
4.8.1	Stili di testo	9
4.8.2	Punteggiatura	9
4.8.3	Composizione del testo	9
4.8.4	Formati ricorrenti	9
4.9	Immagini	10
4.9.1	Inserimento Immagini	10
4.10	Tabelle	11
5	Milestones e Ticketing	12
5.1	Milestones	12
5.1.1	Creazione nuova Milestone	12
5.1.2	Modifica Milestone	13
5.2	Ticketing	13
5.2.1	Creazione nuovo Ticket	13
5.2.2	Modifica Ticket	14
5.2.3	Ciclo di vita di un Ticket	14



6	Analisi dei Requisiti	16
6.1	Norme per i requisiti	16
6.2	Norme per i casi d'uso	17
6.3	Tracciamento	17
7	Norme di verifica	18
7.1	Verifica dei documenti	18

Elenco delle tabelle

1	Revisioni e Milestones	6
---	----------------------------------	---

Elenco delle figure

1	Creazione di una nuova Milestone	12
2	Modifica di una Milestone	13
3	Creazione di un nuovo <u>Ticket</u>	14
4	Modifica di un <u>ticket</u>	14

1 Introduzione

1.1 Scopo del prodotto

Con progetto “MyTalk” intendiamo un sistema software di comunicazione tra utenti mediante browser, utilizzando solo componenti standard, senza dover installare plugin o programmi esterni. L'utilizzatore dovrà poter chiamare un altro utente, iniziare la comunicazione sia audio che video, svolgere la chiamata e terminare la chiamata ottenendo delle statistiche sull'attività.

1.2 Scopo del documento

Questo documento viene redatto per definire le norme da adottare da parte di tutti i componenti del gruppo Software Synthesis durante il periodo di svolgimento del capitolato “MyTalk”, commissionato dall'azienda Zucchetti SPA. In particolare si andranno a definire le regole per:

- Relazioni interpersonali e comunicazione
- Definizione dell'ambiente di lavoro
- Redazione documenti
- Convenzioni e norme per l'analisi e la progettazione
- Convenzioni e norme per la verifica di file e documenti

1.3 Ambiguità

Al fine di evitare incomprensioni dovute all'uso di termini tecnici nei documenti, viene redatto e allegato il documento “Glossario.pdf” dove vengono definiti e descritti tutti i termini marcati con una sottolineatura.

2 Relazioni interpersonali e comunicazione

2.1 Comunicazione interna

La comunicazione tra i vari componenti del team avverrà principalmente durante gli incontri che si terranno in un luogo fisico comune. Per evitare problemi dovuti alla mancata presenza da parte di un membro, le notifiche dell'attività svolta durante l'incontro verranno scritte in un calendario comune messo a disposizione. Nei casi in cui il lavoro si svolgerà presso la propria abitazione, si potrà utilizzare *Skype* per avviare chat, chiamate e videoconferenze di gruppo. Qualora si evinca la necessità di un ritrovo fisico comune ma vi è l'impossibilità da parte di alcuni membri di recarsi nel luogo prefissato, allora si procederà suddividendo il gruppo in due sottogruppi, ciascuno con un luogo di ritrovo prefissato. I due team di sviluppo quindi potranno comunicare con l'altro team mediante software di comunicazione prestabilito.

2.2 Comunicazione esterna

I contatti verso l'esterno saranno a cura del responsabile di progetto. A tale scopo è stato creato un indirizzo di posta elettronica tramite il quale la persona incaricata tratterà a nome dell'intero gruppo Software Synthesis. L'email sopra descritta è *info@softwaresynthesis.org* e l'inoltro di risposte a tutti gli altri membri sarà sempre a carico del responsabile di progetto che dovrà utilizzare i metodi descritti nella sezione 2.1 "Comunicazione Interna".

2.3 Incontri Interni

Saranno fissate delle riunioni formali decise dal responsabile di progetto che provvederà rendere noto a tutti luogo, data e ora della seduta mediante email personale con almeno tre giorni di anticipo. Nell'email sarà contenuta anche la motivazione per cui si è reso necessario l'incontro. Ai membri è richiesta la conferma di partecipazione tramite risposta sempre via email. Nel caso un componente del team non possa essere presente alla riunione decisa, dovrà specificarne il motivo nell'email di risposta al responsabile. Ogni membro del gruppo ha la possibilità di richiedere un incontro interno: tale domanda dovrà venir indirizzata sempre al responsabile di progetto, il quale la visionerà e pianificherà un eventuale incontro. Ad ogni riunione verrà redatto un verbale dove verranno annotate tutte le decisioni prese e i punti salienti della discussione.

2.4 Incontri Esterni

Sarà il responsabile di progetto a prendere accordi per incontri con il committente o con i proponenti. Ogni membro del gruppo può richiedere un incontro esterno al responsabile, presentando una motivazione. Questa necessità verrà presentata all'intero team e solo nel momento in cui ci saranno tre conferme, con relativa presenza all'incontro, il responsabile provvederà a prendere appuntamento con la parte esterna. In caso contrario la proposta sarà bocciata.

3 Ambiente di lavoro

3.1 Repository

Per un corretto svolgimento del progetto si è reso necessario adottare un luogo dove poter cercare e salvare tutti i documenti da redigere, nonché i file di codifica. Per questo si è deciso di adottare un repository ospitato da *github*. Il motivo principale di questa scelta sta nel fatto che Git, rispetto ad altri sistemi quali Sourceforge ad esempio, è un sistema di controllo di versione distribuito. Rientra infatti nei repository di tipo Distributed Control Version System (DVCS) mentre sourceforge rientra nei repository di tipo Centralized Control Version System. La differenza sta nel fatto che in Git ognuno di noi avrà in locale l'intera copia del repository, quindi tutti i vari commit, rami ecc. mentre con sourceforge in locale si ha solo la situazione dell'ultimo commit e se si deve eseguire operazioni di rollback si ha bisogno della connessione di rete per potersi connettere al server centrale.

3.1.1 Registrazione

Per utilizzare github occorre registrarsi presso:

<https://github.com>.

per far ciò basta inserire un username, un indirizzo email e una password nell'apposito form proposto nella pagina sopra indicata.

3.1.2 Installazione

Per aiutare i componenti del gruppo nella fase di installazione di github, è stata creata appositamente una guida che passo passo spiega come installare e configurare nel modo giusto il software. La guida è reperibile all'indirizzo

<https://github.com/SoftwareSynthesis/SoftwareEngineeringProject/blob/master/Manuals/GuidaGit/MasterDocument.pdf>.

3.1.3 Struttura

- **Progetto:** l'indirizzo web a cui fa riferimento l'intero progetto è:

<https://github.com/SoftwareSynthesis/SoftwareEngineeringProject>

- **Documentazione:** tutti i documenti saranno reperibili all'indirizzo

<https://github.com/SoftwareSynthesis/SoftwareEngineeringProject/Documents>

Ogni documento redatto dovrà essere contenuto in una sotto-cartella della directory "Documents" nominata come il nome del documento stesso: questa conterrà i file $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$. E' inoltre presente una cartella "Revisioni", composta a sua volta da quattro sotto-cartelle ("RR", "RP", "RQ", "RA") che conterranno i documenti formali consegnati alle quattro revisioni del progetto ed una cartella "Pics" che conterrà tutte le immagine utilizzate nei documenti.

- **Codice:** tutti i file sorgente saranno reperibili all'indirizzo

<https://github.com/SoftwareSynthesis/SoftwareEngineeringProject/Code>

Le regole da adottare per utilizzare questa cartella saranno redatte non appena si verificherà la necessità.

- **Others:** per qualsiasi altro file di utilità per il progetto, è stata predisposta una cartella "Others" utilizzabile da tutti i membri.

3.1.4 Sistema di versionamento

Come sistema di controllo di versione è stato adottato *git*. Questa scelta è stata decisa dopo aver individuato diversi pregi, tra i quali:

- velocità;
- design semplice;
- forte supporto allo sviluppo non-lineare (possibilità di migliaia di rami paralleli);
- completamente distribuito;
- capacità di gestire, in modo efficiente (velocità e dimensione dei dati), grandi progetti come il kernel Linux.

3.2 Sistema di tracciamento dei requisiti

Al fine di rendere automatico e sistematico la gestione del tracciamento dei requisiti, è stato creato un sistema che si appoggia al sito aziendale. Ogni volta che un membro del gruppo necessita di interagire con tale sistema, si dovrà autenticare alla pagina

<http://www.softwaresynthesis.org/Login.php>

Il sistema di tracciamento offre le seguenti 5 opzioni:

- **Gestione requisiti:** permette l'inserimento, modifica e la cancellazione di un requisito.
- **Gestione fonti:** permette l'inserimento, modifica e la cancellazione di una fonte da intendersi come ad esempio il capitolato d'appalto.
- **Gestione casi d'uso:** permette l'inserimento di un caso d'uso con relativi scenari e requisiti associati.
- **Gestione attori:** permette l'inserimento, modifica e la cancellazione di un attore utilizzabile in seguito per lo sviluppo di un caso d'uso.
- **Stampa:** crea in modo automatico un file di estensione .tex contenente la lista dei requisiti analizzati, i casi d'uso studiati e il tracciamento tra i requisiti e le fonti. Successivamente alla stampa tale file sarà da includere all'intero del documento "*analisi_dei_requisiti.tex*"

Il sistema è basato su tecnologie MySQL con creazione del database sotto engine `innodb`. Per sopperire ad alcune mancanze del sistema di tracciamento, lo sviluppatore si potrà appoggiare alla piattaforma phpMyAdmin fornita dal web host, al fine di modificare e gestire alcuni dati del database. In particolare, se ne obbliga l'utilizzato nei seguenti casi:

- modifica di un caso d'uso;
- cancellazione di un caso d'uso.

Quest'obbligo di gestione è dovuto a mancanze temporali nella fase di sviluppo del sistema, nel periodo antecedente alla consegna dei capitolati, e anche in parte per la complessità di implementazione.

Ogni altra modifica attuata tramite phpmyadmin è assolutamente vietata per mansioni che non siano riportate al punto precedente.

3.3 Sistema operativo

L'intero progetto verrà portato avanti attraverso sistemi Unix e Windows, più in particolare Mac OSX 10.8.2, Fedora 17, Windows 7, Windows 8. Questa scelta deriva dal fatto che, essendo il progetto MyTalk un applicativo web, non si sono riscontrate dipendenze alcune da librerie di sistema.

3.4 Grafici UML

Per la produzione di grafici UML si adotterà il software ConceptDraw. Questo scelta perchè il tool si presenta molto semplice da usare, è molto potente e permette la creazione di moltissimi grafici UML adottando lo standard 2.0. La qualità dei grafici prodotti inoltre è molto buona.

3.5 Grafici di Attività

Come supporto alla pianificazione del progetto si è scelto di utilizzare Project Libre nella versione 1.5.2, software molto leggero e facile da installare. E' un tool opensource e multi piattaforma. Permette la creazione di diagrammi di Gantt in modo molto semplice con possibilità di esportarli in formato jpg.

Per il download e l'installazione si rimanda al seguente link:

<http://sourceforge.net/projects/projectlibre/>.

4 Ambiente documentale

In questa sezione verranno illustrati i vari standard utilizzati dal team per la produzione di documenti durante l'intero progetto.

4.1 Software utilizzati

Tutti i documenti dovranno essere scritti in italiano con \LaTeX , un software free di composizione testuale che comprende una serie di caratteristiche atte alla produzione di documentazione tecnica e scientifica.

Per attuare questa scelta, l'azienda Software Synthesis ha deciso di utilizzare come editor \LaTeX *TexMaker*, software disponibile sia per ambienti Unix che Windows. Si presenta come un software molto flessibile che include al suo interno la correzione ortografica automatica, l'autocompletamento e un visualizzatore di PDF integrato.

La versione da utilizzare è l'ultima disponibile in tale momento ed è la 3.5.2. Per il download seguire il seguente link:

<http://www.xmlmath.net/texmaker/download.html>.

Per l'installazione, una volta scaricato, basta seguire le indicazioni presenti nel file scaricato precedentemente.

4.2 Nomi dei documenti

Il nome dei documenti dovrà rappresentare in modo univoco la natura del file e sarà composto nel seguente modo:

nome_del_file.X.Y.estensione

dove:

- **nome_del_file**: il nome del file non dovrà contenere lettere maiuscole né caratteri speciali. Nel caso sia composto da più parole, queste vanno separate con il carattere underscore.
- **Variabile X**: Indica il raggiungimento di uno stato formale rispetto ad una Milestone. Assumerà quindi principalmente 4 valori, dall'1 al 4, attribuiti come segue:

Milestone	X
Revisione dei Requisiti (RR)	1
Revisione dei Requisiti (RP)	2
Revisione dei Requisiti (RQ)	3
Revisione dei Requisiti (RA)	4

Tabella 1: Revisioni e Milestones

- **Variabile Y**: parte dal valore 0 e viene incrementata ogni qualvolta, senza limite superiore, si apportano modifiche sostanziali al documento, come ad esempio la stesura di una nuova sezione o correzione di errori.

4.3 Impostazioni di base di un documento

Ogni documento dovrà attenersi alle seguenti regole per la sua redazione:

- **Intestazione:** composta dalla sezione del documento nella parte sinistra e dal logo nella parte destra.
- **Piè di pagina:** composto dal nome del documento con relativa versione sulla sinistra e dal numero di pagina sulla destra.

La prima pagina di ogni documento dovrà contenere come prima cosa il logo dell'azienda Software Synthesis, il nome del documento e una tabella che riporta le seguenti informazioni sul file:

- nome del file;
- versione;
- data creazione;
- data ultima modifica;
- stato;
- uso;
- redattori;
- approvato da;
- verificatori.

Nella seconda pagina invece dovrà essere presente la tabella delle modifiche apportate. Queste dovranno essere inserite in ordine cronologico dalla più recente alla creazione, specificando:

- versione del documento;
- descrizione dell'intervento;
- redattore;
- data modifica.

Seguirà su una nuova pagina l'indice dell'intero documento e in caso di presenza di tabelle e immagini sarà presente anche l'indice delle tabelle e l'indice delle immagini a seguire. La presenza di questi due indici va specificata settando a "true" due flag presenti nel documento "Template".

4.4 Tipi di documento

- **Informali:** verranno "etichettati" come informali tutti i documenti in fase di redazione, oppure completati ma non ancora approvati dal responsabile di progetto. Vanno utilizzati solo internamente e quindi non verranno distribuiti all'esterno. Sono reperibili nella cartella "Documents" del repository e sono organizzati secondo quanto previsto nella sezione 3.1.3 "Struttura repository".
- **Formali:** tutti i documenti approvati dal responsabile di progetto sono da considerarsi formali e quindi pronti alla distribuzione verso l'esterno. Sono reperibili nella cartella "Revisioni" del repository e sono organizzati secondo quanto previsto nella sezione 3.1.3 "Struttura repository".
- **Verbali:** documenti redatti dal responsabile di progetto dopo un avvenuto incontro esterno. Servono come promemoria dell'incontro avvenuto e delle tematiche trattate. Dovranno essere denominati

verbale_incontro_{dataincontro}

dove con {dataincontro} intendiamo la data dell'incontro espressa nel formalismo descritto nella sezione 4.8.4 "Formati Ricorrenti".

Ogni verbale dovrà contenere le seguenti informazioni riportate come seguono:

- **Data:** indica la data dell'incontro
- **Ora:** indica l'ora dell'incontro espressa in "HH:MM" dove HH sarà compresa tra 00 e 23 ed indicherà le ore mentre MM sarà compreso tra 00 e 59 ed indicherà i minuti.
- **Luogo:** indica il luogo dell'incontro e dovrà essere espresso nel seguente formalismo:
provincia, città, via, numero civico
- **Durata:** indica la durata dell'incontro espressa in minuti
- **Partecipanti:** indica tutti i partecipanti presenti alla riunione. Deve essere un elenco di nomi e cognomi suddivisi per membri interni ed esterni.
- **Oggetto:** indica l'argomento trattato all'incontro. Seguono i punti salienti e le decisioni prese.

Il verbale dovrà essere approvato da tutti i componenti firmandolo entro e non oltre il successivo incontro. L'approvazione è ufficializzata automaticamente nel momento in cui sono poste tutte le firme dei presenti. In via del tutto eccezionale, nel qual caso un componente non possa approvare il documento nei termini prestabiliti, è compito del responsabile assicurarsi che ciò avvenga il prima possibile. Nel caso in cui un membro del gruppo non abbia potuto partecipare alla riunione, esso dovrà comunque prendere visione del verbale e dovrà firmarlo per accertare la presa visione dello stesso.

4.5 Ciclo di vita di un documento

1. Il redattore di un nuovo documento dovrà redigere in locale la prima impostazione del documento e solo allora potrà caricarlo nel repository secondo le regole indicate nella sezione 3.1 "Repository".
2. Ad ogni sostanziale modifica verrà incrementato il contatore Y (vedi sezione 4.2 "Nomi di documenti") e seguirà un commit nel repository.
3. Quando il redattore finisce la stesura o revisione del documento, questo verrà proposto al verificatore il quale avrà il compito di controllare che siano state seguite le norme riguardanti la stesura di documenti (vedi sezione 7.1 "Verifica dei documenti"). Il verificatore potrà dare due esiti, positivo o negativo.
4. In caso di esito negativo, il documento viene riproposto al redattore evidenziandone gli errori riscontrati. Se invece l'esito è positivo, il documento è stato redatto secondo le norme e non presenta errori, passerà quindi al responsabile di progetto che provvederà all'approvazione formale.
5. Anche il responsabile di progetto può dare due giudizi, idoneo o non idoneo. Nel caso di idoneità, il documento è da considerarsi formale e può essere inserito nella cartella corrispondente alla Milestone presente nella directory "Revisioni".
6. In caso di non idoneità, il documento verrà rimandato al redattore notificando il motivo di tale decisione e si ripartirà dal punto 3.

4.6 Template

Ogni documento dovrà essere generato includendo il template \LaTeX presente nella cartella “Documents”. Questo file è stato redatto prima dell’inizio di stesura di ogni altro documento sotto comune accordo. La modifica perciò di tale template deve essere richiesta all’amministratore di progetto che analizzerà l’esigenza e darà una risposta positiva o negativa. In caso positivo, si procederà all’attuazione delle modifiche e verrà informato l’intero team al fine che ogni membro modifichi il documento su cui sta lavorando.

4.7 Glossario

Il glossario è un documento nel quale vengono riportate tutte le parole difficilmente comprensibili o dal significato ambiguo presenti nei documenti. E’ un documento un po’ particolare in quanto non rispetta le regole stilistiche degli altri documenti. Non sarà presente l’indice dei contenuti, ma solo una pagina iniziale contenente le informazioni del documento, la tabella delle modifiche, una breve introduzione e a seguire le voci ordinate alfabeticamente.

4.8 Norme tipografiche

Questa sezione descrive le norme da utilizzare per l’utilizzo dell’ortografia, tipografia e l’assunzione di uno stile uniforme nel redigere i documenti.

4.8.1 Stili di testo

- **Grassetto:** usato per evidenziare passaggi importanti. Si utilizza anche su elementi immediatamente seguenti agli elenchi per evidenziare l’oggetto trattato nel paragrafo.
- **Corsivo:** usato per dare enfasi a parole o frasi su cui prestare attenzione.
- **Sottolineato:** le parole sottolineate sono riportate nel glossario. Ogni occorrenza di tali parole va sottolineata.
- **Maiuscolo:** le parole in maiuscolo vanno usate solo per acronimi.

4.8.2 Punteggiatura

Qualsiasi segno di punteggiatura non deve mai seguire uno spazio ma deve precederne uno. Dopo un punto fermo deve esserci uno spazio seguito da una lettera maiuscola.

Le parentesi vanno usate per raggruppare un periodo, non devono aprirsi con un carattere di spaziatura e non devono chiudersi con un carattere di punteggiatura o di spaziatura.

4.8.3 Composizione del testo

Gli elenchi, sia puntati che numerati, vanno usati per elencare una serie di elementi e solitamente hanno una breve lunghezza con la quale descrivono l’elemento in questione. Per questo, ogni punto elenco deve finire con un punto e virgola (;) tranne l’ultimo elemento che finirà con un punto (.), mentre nel caso di punti elenco più discorsivi la frase finirà con un punto ogni volta. Per quanto concerne le note a piè di pagina invece, ogni nota dovrà cominciare con la prima lettera della prima parola maiuscola, senza spaziatura davanti e finire con un punto.

4.8.4 Formati ricorrenti

- **Path:** per scrivere indirizzi web e url si adotterà il comando prestabilito da \LaTeX `\url`.
- **Date:** per scrivere una data si dovrà utilizzare il formato AAAA-MM-GG (standard ISO 8601) dove:

- AAAA: sta ad indicare l'anno;
- MM: sta ad indicare il mese;
- GG: sta ad indicare il giorno.
- **Sigle:** di seguito sono elencate una serie di sigle, da usare principalmente all'interno di diagrammi o tabelle per risparmiare spazio.
 - AR: documento "Analisi dei Requisiti"
 - PdP: documento "Piano di Progetto"
 - PdQ: documento "Piano di Qualifica"
 - NdP: documento "Norme di Progetto"

4.9 Immagini

Tutte le immagini utilizzate nei documenti devono avere estensione .png. Nel caso l'immagine abbia una grande dimensione, va compressa con il formato .jpg. Devono risiedere tutte nella cartella "pics" (<https://github.com/SoftwareSynthesis/SoftwareEngineeringProject/Documents/pics>).

Nel caso di immagini riguardanti diagrammi UML, si deve rispettare la seguente regola per il nome file:

tipologia_x1.x2....xn

dove "tipologia" dovrà essere una di queste sigle:

- **UC:** diagrammi casi d'uso;
- **DC:** diagrammi delle classi;
- **DO:** diagrammi degli oggetti;
- **DS:** diagrammi di sequenza;
- **DA:** diagrammi di attività;
- **DP:** diagrammi di package.

e "x1.x2...xn" rappresenteranno la numerazione gerarchica dei diagrammi, dove il numero più a sinistra rappresenta il padre e quelli più a destra i figli.

Nel caso di immagine generiche, il loro nome dovrà rispecchiare il contenuto di tale immagine.

4.9.1 Inserimento Immagini

L'inserimento di immagini all'interno del documento verrà fatto tramite l'utilizzo del comando `\begin{figure}` e `\includegraphics[parametri]{riferimento}` dove:

- **parametri:** indica una serie di parametri da utilizzare per formattare al meglio l'immagine, come una scala di ridimensionamento (`[scale=0.8]` indica un ridimensionamento del 20%), l'altezza e larghezza (`[width=2cm, heigh=4cm]` indica un'immagine larga 2 cm e alta 4 cm) oppure parametri del tipo `[width=\textwidth]` per indicare che la figura corrisponda esattamente alla larghezza del testo;
- **riferimento:** va messo il nome dell'immagine presente nella cartella "pics" (vedi sezione 3.1.3) che si vuole inserire.

Prima della chiusura (`\end{figure}`) vanno inseriti anche i comandi `\caption{didascalia}` e `\label{riferimento}` per permettere l'indicizzazione e l'identificazione delle immagini tramite un codice.

4.10 Tabelle

L'inserimento di tabelle all'interno del documento verrà fatto tramite l'utilizzo del package `tabularx` che permette la creazione di colonne a larghezza dinamica. Vanno inseriti prima della chiusura della tabella (quindi prima del comando `\endtabularx`) i comandi `\caption{didascalia}` e `\label{riferimento}` per permettere l'indicizzazione e l'identificazione delle tabelle tramite un codice.

5 Milestones e Ticketing

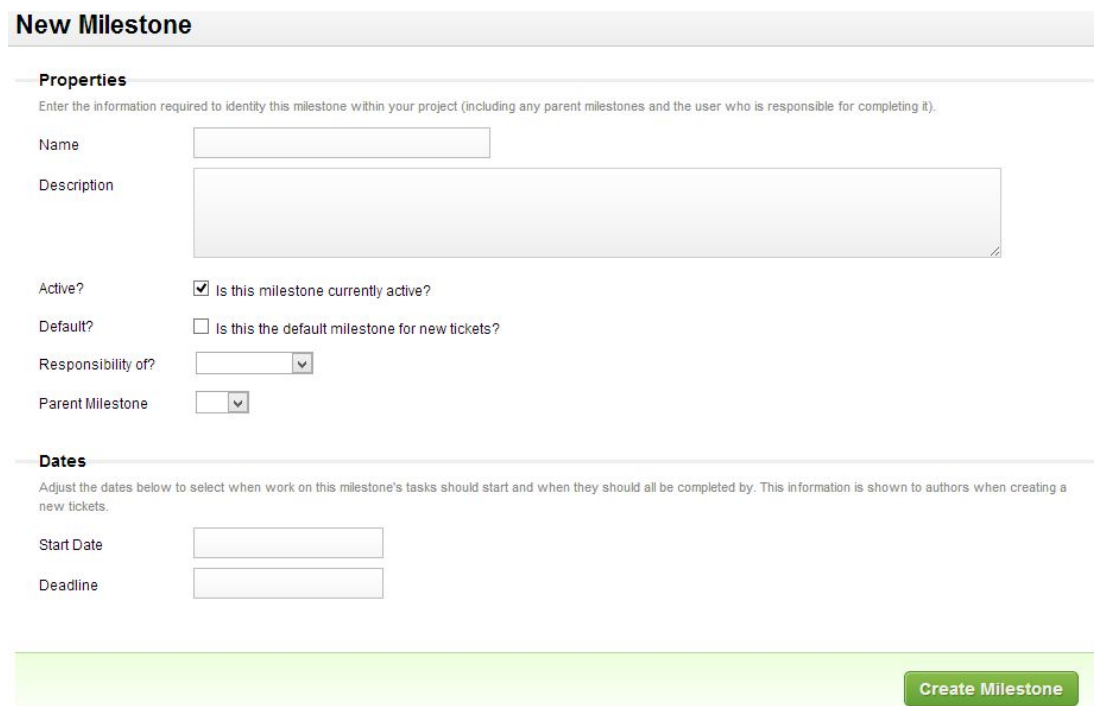
Al fine di aiutare lo svolgimento delle operazioni di assegnazione dei lavori e revisione durante l'intero ciclo di vita del progetto, l'azienda Software Synthesis userà il servizio *Codebase*. Questo permette la creazione di Milestones e tickets, strumenti molto utili per tracciare il lavoro del team e anche per delineare una roadmap degli obiettivi a breve e lungo termine che l'azienda si prefigge di raggiungere.

5.1 Milestones

Come prima cosa, il responsabile di progetto dovrà inserire tutte le Milestones previste dal progetto, ossia le quattro revisioni.

5.1.1 Creazione nuova Milestone

Per inserire una nuova Milestone ci si dovrà posizionare nella pagina web dedicata alle Milestone e cliccare su “New Milestone”. Il form va completato nel seguente modo:



New Milestone

Properties

Enter the information required to identify this milestone within your project (including any parent milestones and the user who is responsible for completing it).

Name

Description

Active? ☒ Is this milestone currently active?

Default? ☐ Is this the default milestone for new tickets?

Responsibility of?

Parent Milestone

Dates

Adjust the dates below to select when work on this milestone's tasks should start and when they should all be completed by. This information is shown to authors when creating a new tickets.

Start Date

Deadline

Create Milestone

Figura 1: Creazione di una nuova Milestone

- **Name:** inserire il nome della Milestone;
- **Description:** inserire una breve descrizione della Milestone;
- **Active?:** flag da spuntare se si vuole che la Milestone creata sia quella attiva;
- **Default?:** flag da spuntare se si vuole che i futuri tickets creati siano riferiti a questa Milestone;
- **Responsibility of?:** selezionare la persona responsabile;

- **Parent Milestone:** selezionare se la Milestone che si va a creare è figlia di qualche altra Milestone;
- **Start Date:** inserire la data da cui far iniziare la Milestone;
- **Deadline:** inserire la data di chiusura delle Milestone.

Nel caso non si conosca qualche informazione, si può lasciare il relativo campo bianco e modificarlo successivamente (vedi paragrafo seguente) quando sarà resa pubblica l'informazione.

5.1.2 Modifica Milestone

Per modificare una Milestone, ci si deve posizionare nella scheda "Edit Milestone". Per raggiungerla, basta entrare nella Milestone che si vuole modificare e cliccare sul bottone "Edit Milestone" situato nella parte alta dello schermo a destra.



Figura 2: Modifica di una Milestone

Per la compilazione dei campi, valgono le stesse regole riportate nella sezione 5.1.1 "Creazione nuova Milestone".

Da notare che si può eliminare anche una Milestone da tale scheda, basta cliccare sul bottone "Delete Milestone" presente nella parte destra della pagina.

5.2 Ticketing

Un ticket rappresenta un compito da svolgere.

5.2.1 Creazione nuovo Ticket

Per inserire un nuovo ticket ci si dovrà posizionare nella pagina web dedicata ai Tickets e cliccare su "New Ticket". Il form va completato compilando obbligatoriamente i seguenti campi:



- **Summary:** inserire il nome del compito da assegnare;
- **Status:** da settare su "New";
- **Priority:** selezionare la priorità effettiva del compito;
- **Ticket Type:** da settare su "Task";
- **Assigned User:** selezionare la persona che dovrà svolgere il compito;
- **Milestone:** selezionare la Milestone corrispondente del compito.

Tutte le altre informazioni possono essere completate se si ritiene opportuno dare maggiori dettagli.

New Ticket

Summary — enter a brief description of this ticket (up to 250 characters)

Additional information — you may enter optional information for this ticket in the field below (this is optional)


B I  

Ticket Properties **Milestone & Dates** **Attachments** **Watchers**

STATUS **PRIORITY** **CATEGORY** **TICKET TYPE**

New Normal General Feature

ASSIGNED USER **TAGS**


 This ticket will be added to the **RR** milestone. To change this select the Milestone and Dates tab above.

Create Ticket

Figura 3: Creazione di un nuovo Ticket


5.2.2 Modifica Ticket


Durante l'intero ciclo di vita di un ticket, questo viene modificato molte volte come avviene ad esempio quando dobbiamo modificare lo stato del ticket. Per far ciò, basta entrare nel ticket che si vuole modificare e appare subito la scheda di modifica.


Iniziare casi d'uso 


TYPE **STATUS** **PRIORITY** **CATEGORY** **ASSIGNEE**

Task Accepted Critical General Marco S



 Reported by Marco S on December 16th, 2012 13:14

 Belongs to the RR milestone.

 Status changed from New to Accepted

 Posted by Marco Schivo on December 16th, 2012 13:16

Update this ticket

B I  

Change Properties **Milestone & Dates** **Related Tickets** **Add Time** **Attachments** **Watchers**

SUBJECT

Iniziare casi d'uso


STATUS **PRIORITY** **CATEGORY** **TICKET TYPE**

Accepted Critical General Task

ASSIGNED USER **TAGS**

Marco Schivo

Post Ticket Update

 You are currently receiving updates by e-mail when this is updated by others.

[Stop receiving updates](#)

[View all 1 watchers](#)

If you'd like to delete this ticket, you can use the button below. This is not reversible.

Delete Ticket

Figura 4: Modifica di un ticket

5.2.3 Ciclo di vita di un Ticket

1. **Creazione:** come prima cosa, il ticket va creato come descritto nella sezione 5.2.1 “Creazione nuovo Ticket”.

2. **Esecuzione:** ogni membro dovrà prendere atto dei tickets a lui pervenuti e una volta visionati impostare lo stato ad “Accepted” o ad “Invalid”. Una volta accettati, lo stato va modificato in “In Progress” quando si inizia a lavorarci e “Completed” quando si è portato a termine il compito assegnato.
3. **Verifica:** una volta che il compito è stato completato, si procede nella creazione di un nuovo ticket di tipo “Verification” il quale verrà assegnato ad un verificatore. Il nuovo ticket va compilato nel seguente modo:
 - **Summary:** dovrà contenere il nome del documento o file da verificare;
 - **Status:** da settare su “New”;
 - **Priority:** selezionare la priorità effettiva del compito;
 - **Ticket Type:** da settare su “Verification”;
 - **Assigned User:** selezionare il verificatore che dovrà svolgere il compito;
 - **Milestone:** selezionare la Milestone entro cui il documento dovrà essere verificato.
4. **Esecuzione verifica:** per ogni errore (esclusi quelli grammaticali) dovrà essere creato un nuovo ticket nel seguente modo:
 - **Summary:** dovrà contenere il nome del documento o file da modificare;
 - **Additional information:** dovrà contenere la descrizione dell’errore presente nel documento;
 - **Status:** da settare su “New”;
 - **Priority:** selezionare la priorità effettiva del compito;
 - **Ticket Type:** da settare su “Bug”;
 - **Assigned User:** selezionare il Responsabile del progetto;
 - **Milestone:** selezionare la Milestone entro cui il documento dovrà essere verificato.

Sarà compito del Responsabile approvare o invalidare il ticket. In caso di approvazione, il Responsabile dovrà decidere a chi far eseguire la modifica. Il membro selezionato dovrà quindi risolvere l’errore. Una volta completata la modifica il Responsabile dovrà riapplicare la procedura a partire dal punto 3.

Quando tutti i tickets di una Milestone avranno stato “Completed”, significa che tutte le attività sono state svolte e il Responsabile potrà procedere alla chiusura della Milestone.

6 Analisi dei Requisiti

L'analisi dei requisiti dovrà essere redatta dagli analisti. Il documento si presenterà suddiviso principalmente in tre parti: una prima parte riguardante i requisiti, una seconda parte riguardante i casi d'uso e la parte finale riguardante il tracciamento dei due elementi appena elencati.

6.1 Norme per i requisiti

La parte riguardante i requisiti dovrà elencare ordinatamente tutti i requisiti evinti dal capitolato d'appalto o dagli incontri con il proponente. Per far ciò, *ogni requisito avrà nome univoco* e dovrà essere indicato nel formalismo:

$$R\{\text{Ambito}\}\{\text{Tipo}\}\{\text{Priorità}\}\{\text{Gerarchia}\}$$

dove:

- **Ambito:** indicherà l'ambito del requisito e potrà assumere i seguenti valori
 - U per indicare “Utente”;
 - S per indicare “Sistema”.
- **Tipo:** indicherà il tipo del requisito e potrà assumere i seguenti valori
 - F per indicare “Funzionale”, ossia offre una funzione all'utente;
 - Q per indicare “di Qualità”, ossia offre una qualità al software;
 - P per indicare “Prestazionale”;
 - D per indicare “Dichiarativo”.
- **Priorità:** indicherà la priorità del requisito e potrà assumere i seguenti valori
 - O per indicare “Obbligatorio”, ossia necessario per la realizzazione del progetto;
 - D per indicare “Desiderabile”, ossia non obbligatorio ma, se implementato, aumenta il valore del progetto;
 - F per indicare “Facoltativo”, ossia un requisito da implementare se avanzano risorse al termine del progetto.
- **Gerarchia:** indicherà il grado di parentela dei requisiti e andrà specificato nella sintassi

$$x.y.z$$

dove il numero più a sinistra rappresenta il padre e quelli più a destra i figli.

Ogni requisito dovrà essere presente nel documento attraverso una “scheda”, dove oltre i dettagli ricavabili dal nome, dovranno essere specificati anche una breve descrizione, la fonte (ossia la loro origine) e il motivo.

Esempio

“RUFO2.3.1” indica un requisito utente, funzionale obbligatorio con padre RUFO2.3.0 che a sua volta è figlio di RUFO2.0.0.

6.2 Norme per i casi d'uso

Il nome di ogni caso d'uso dovrà essere anch'esso, come per i requisiti, univoco e nella forma:

$$UC\{n1.n2...n3\}$$

dove con “n1.n2...n3” intendiamo la gerarchia, ossia il grado di parentela. Come per i requisiti, il numero più a sinistra rappresenta il padre e quelli più a destra i suoi figli.

Ogni caso d'uso dovrà avere i seguenti dettagli:

- Nome;
- Grafico UML;
- Scopo;
- Pre;
- Post;
- Attori;
- Scenario Principale;
- Scenari Secondari.

Esempio

“UC1.2.3.0” indica un caso d'uso con padre UC1.2.0 che a sua volta è figlio di UC1.0.0.

6.3 Tracciamento

Durante l'intero ciclo di vita del progetto, sono previste misure per il tracciamento dei casi d'uso con i relativi requisiti. Questo avviene grazie all'utilizzo del Sistema di Tracciamento dei requisiti descritto nella sezione 3.2 “Sistema di tracciamento dei requisiti”. Tutta la gestione è automatica e in output sarà visualizzata una tabella con tre colonne: Codice UC, Nome UC, Requisito. Questa soluzione facilita notevolmente la gestione di grandi quantità di dati e un notevole risparmio di tempo.

7 Norme di verifica

La verifica è un'attività molto importante svolta dal verificatore. Lo scopo è di accertare che i prodotti ottenuti, qualunque sia il loro tipo, siano conformi le norme e convenzioni riportate su questo documento. Oltre a questo, il verificatore dovrà anche assicurare sotto la propria responsabilità che tali prodotti rispettino i requisiti di qualità descritti nel documento Piano di Qualifica (piano_di_qualifica.1.0.pdf).

7.1 Verifica dei documenti

La verifica di un documento avviene nel momento in cui il redattore ha finito di redigere completamente il documento. Tale processo avverrà principalmente in tre fasi:

1. Si procederà con un'analisi ortografica, compreso il corretto uso della punteggiatura, degli accenti e degli apostrofi. Per far ciò il verificatore potrà aiutarsi con lo strumento "Verifica ortografia" presente in TexMaker raggiungibile dalla scheda "Modifica->Verifica Ortografia". Successivamente controllerà l'assenza di errori grammaticali, logici e sintattici nei periodi ed infine la coerenza del documento con le norme previste nella sezione 4 "Ambiente documentale".
2. Se è la prima volta che il documento viene verificato, il verificatore procederà con un metodo di tipo walkthrough, ossia analizzerà l'intero documento dall'inizio alla fine. Se invece il documento era già stato verificato e rimandato per via di errori, si adopererà un metodo di tipo inspection, ossia analizzerà il documento solo dove sono state apportate le modifiche accertandosi che siano stati corretti gli errori rilevati e che non ne siano stati introdotti nuovi.
3. Il verificatore prenderà nota in modo ordinato di tutti gli errori riscontrati e procederà con la creazione dei tickets relativi che saranno rivolti al redattore del documento come descritto nella sezione 5.2.3 "Ciclo di vita di un Ticket".