

Premi



Dazzle Works

Norme di Progetto

Versione	1.0.0
Redattori	Bogdan Suierica Emanuele Crespan
Verificatori	Matteo Agostinetto
Responsabili	Valerio Burlin
Uso	Interno
Lista di distribuzione	<i>Dazzle Works</i> Prof. Vardanega Tullio, Dr. Cardin Riccardo

Documento contenente l'insieme di norme stabilite dal gruppo *Dazzle Works* per la realizzazione di Premi.



Indice

1	Introduzione	1
1.1	Scopo del Documento	1
1.2	Scopo del Progetto	1
1.3	Glossario	1
1.4	Riferimenti	1
1.4.1	Normativi	1
2	Collaborazione	2
2.1	Comunicazioni	2
2.1.1	Comunicazioni interne	2
2.1.2	Comunicazioni esterne	2
2.2	Composizione email	2
2.2.1	Mittente	2
2.2.2	Destinatario	2
2.2.3	Oggetto	3
2.2.4	Corpo	3
2.2.5	Allegati	3
2.3	Riunioni	3
2.3.1	Interne	3
2.3.2	Esterne	3
2.4	Verbale	4
2.5	Repository	4
2.5.1	Utilizzo del servizio	4
2.5.2	Struttura	4
3	Ruoli di Progetto	6
3.1	Responsabile di Progetto	6
3.2	Amministratore	6
3.3	Analista	7
3.4	Progettista	7
3.5	Verificatore	7
3.6	Programmatore	7
4	Documenti	8
4.1	Template	8
4.2	Versionamento	8
4.3	Struttura dei documenti	8
4.3.1	Prima pagina	8
4.3.2	Diario delle modifiche	9
4.3.3	Indice	9
4.3.4	Formattazione delle pagine	9
4.4	Classificazione documenti	9
4.4.1	Documenti informali	9
4.4.2	Documenti formali	9
4.5	Norme tipografiche	9
4.5.1	Punteggiatura	10
4.5.2	Stile del testo	10
4.5.3	Composizione del testo	10
4.5.4	Formati ricorrenti	10
4.5.5	Sigle	11
4.6	Componenti grafiche	11
4.6.1	Immagini	11
4.6.2	Tabelle	11



4.7	Procedure di avanzamento di un documento	12
5	Ambiente Lavoro	13
5.1	Sistema operativo	13
5.2	Coordinamento	13
5.2.1	Repository Git	13
5.2.2	Google Drive	13
5.2.3	Google Calendar	13
5.3	Ambiente documentale	14
5.3.1	Stesura documenti	14
5.3.2	Script	14
5.3.3	Pianificazione delle attività	14
5.4	Ambiente di sviluppo	14
5.5	Diagrammi UML	14
5.6	Verifica	14
5.6.1	Verifica dei documenti	14
5.7	Verifica dei diagrammi	15
6	Protocolli e Gestione di Progetto	16
6.1	Creazione e gestione dei ticket	16
6.2	Creazione delle milestone	16
6.3	Esecuzione dei compiti	17
6.4	Chiusura della milestone	17
7	Analisi Requisiti	18
7.1	Requisiti	18
7.2	Casi d'uso	18
7.3	Tracciamento	18
8	Progettazione	19
8.1	Diagrammi	19
8.2	Design Pattern	19
8.3	Classi di verifica	19
8.4	Stile di progettazione	19
9	Codifica e Convenzioni	20
9.1	Linguaggi di codifica	20
9.2	Convenzioni di codifica	20
9.3	Nomenclatura	20
9.4	Intestazione di un file Javascript	20
9.5	Commenti	21
A	Lista di controllo	22



Elenco delle tabelle



Elenco delle figure

1	Procedura avanzamento documento	12
2	Creazione ticket	16
3	Creazione milestone	17



1 Introduzione

1.1 Scopo del Documento

Questo documento ha lo scopo di definire le norme che tutti i membri del gruppo *DazzleWorks* dovranno seguire e rispettare durante lo svolgimento del progetto **Premi**. Tutti i membri sono tenuti a leggere il documento seguendo le norme per garantire uniformità al fine di ottenere una migliore efficacia ed efficienza delle attività. I membri del gruppo potranno contattare l'*Amministratore di Progetto* per eventuali suggerimenti, consigli, opinioni riguardanti le norme di progetto. L'*Amministratore di Progetto*, dopo essersi consultato con gli altri membri del gruppo, avrà la responsabilità di accettare o rifiutare i suggerimenti proposti. Il documento, in particolare, pone l'accento sui seguenti punti:

- Iterazione tra i membri del gruppo e la comunicazione verso l'esterno;
- Le modalità di accesso e le norme di utilizzo del repository;
- Stesura documenti e convenzioni di scrittura utilizzate;
- Modalità di lavoro durante lo sviluppo del progetto;
- Organizzazione dell'ambiente di lavoro.

1.2 Scopo del Progetto

Lo scopo del progetto è la realizzazione di un software di presentazione di "slide", sviluppato in tecnologia HTML5 che funzioni sia in ambiente desktop che mobile. Si richiede agli sviluppatori di realizzare effetti grafici a supporto dello "storytelling"¹ che siano di livello comparabile con *Prezi*². Il software dovrà coprire i due momenti fondamentali per questo tipo di attività: la creazione da parte dell'autore e la presentazione al pubblico, sia in presenza diretta che via web.

1.3 Glossario

Al fine di evitare ogni ambiguità e permettere al lettore una migliore comprensione dei termini tecnici, acronimi, e termini che necessitano di essere chiariti, essi sono riportati nel documento *Glossario v1.0.0*. Ogni occorrenza di un termine appartenente al *Glossario* è marcata da una "G" in pedice.

1.4 Riferimenti

1.4.1 Normativi

- <http://www.w3.org/standards/webdesign/htmlcss>
- http://en.wikipedia.org/wiki/ISO_8601

¹Strumento riflessivo per la costruzione di significati interpretativi della realtà.

²È un servizio per la realizzazione di presentazioni.



2 Collaborazione

2.1 Comunicazioni

2.1.1 Comunicazioni interne

Le comunicazioni interne verranno eseguite attraverso la mailing list: dazzleworksgroup@gmail.com.

Quando un membro del gruppo vuole inviare una email a tutti i componenti, deve inviare il messaggio dalla sua casella di posta elettronica personale verso l'indirizzo dazzleworksgroup@gmail.com, dalla quale un inoltro automatico provvederà a mandare l'email agli altri indirizzi di posta elettronica personali dei componenti del gruppo, tenendo traccia di tutte le comunicazioni.

I membri sono tenuti a prestare attenzione al numero di messaggi diffusi. È sempre importante non arrivare a una sovraesposizione del pubblico interno, in quanto si creerebbe solo senso di smarrimento e confusione. Questo non esclude che uno stesso messaggio non sia proposto su più mezzi di informazione, azione spesso necessaria, ma questo presuppone un intervento ponderato e non casuale.

Con lo scopo di facilitare la comunicazione tra i membri del gruppo vengono utilizzati strumenti di messaggistica istantanea e videochiamata come Skype e Google Hangouts. È necessario redigere un verbale nel caso in cui siano state prese decisioni o siano emersi dettagli inerenti allo sviluppo del progetto.

2.1.2 Comunicazioni esterne

Per le comunicazioni esterne è stato creato un indirizzo di posta elettronica:

dazzleworksgroup@gmail.com

Tale indirizzo deve essere l'unico canale di comunicazione verso l'esterno. Sarà solo il *Responsabile di progetto* ad utilizzare l'indirizzo di posta per intrattenere le corrispondenze con i proponenti e i committenti. È compito del *Responsabile di progetto* informare i membri del gruppo delle discussioni avvenute ed eventualmente inoltrare i messaggi alla mailing list.

2.2 Composizione email

In questo paragrafo viene descritta la forma che deve avere una email sia per comunicazione interna che esterna.

2.2.1 Mittente

- **Interno:** indirizzo personale di chi scrive il messaggio;
- **Esterno:** l'unico indirizzo utilizzabile per comunicare verso l'esterno è dazzleworksgroup@gmail.com e deve essere usato esclusivamente dal *Responsabile di progetto*.

2.2.2 Destinatario

- **Esterno:** l'indirizzo del destinatario può avere variazioni a seconda che si voglia comunicare con il Prof. Tullio Vardanega, il Prof. Riccardo Cardin o con i committenti del progetto;
- **Interno:** l'unico indirizzo utilizzabile è dazzleworksgroup@gmail.com.

Le uniche eccezioni permesse sono:

- **Proposte all'Amministratore di progetto:** per eventuali proposte di cambiamento delle norme da parte di un membro del gruppo, quest'ultimo dovrà contattarlo al suo indirizzo di posta elettronica personale;



- **Comunicazione ristretta tra alcuni membri del gruppo:** in questi casi i membri del gruppo utilizzeranno i loro indirizzi personali.

2.2.3 Oggetto

L'oggetto deve sintetizzare il contenuto della email. Deve essere chiaro, breve e possibilmente univoco in modo da riconoscerlo dagli altri precedenti.

Nel caso si debba mandare un messaggio alla mailing list vi è l'obbligo di aggiungere "Group:" all'inizio dell'oggetto.

2.2.4 Corpo

Il corpo del messaggio deve essere chiaro, diretto e avere tutte le informazioni necessarie per permettere a tutti i destinatari di capire correttamente l'argomento trattato. Per riferirsi ad alcuni componenti del gruppo o ruoli di progetto si dovrà usare la seguente sintassi: "**Cognome Nome**" o "**Nome Ruolo**". Alla fine del corpo del messaggio, il mittente dovrà sempre firmarsi con il suo nome, cognome e ruolo all'interno del gruppo.

2.2.5 Allegati

Viene consentito l'uso di allegati che deve essere limitato solo al caso in cui essi siano realmente necessari. È buona norma presentare gli allegati scrivendo alcune righe di spiegazione. Prestare attenzione al formato (preferibilmente in formato PDF) del documento che si sta inviando.

2.3 Riunioni

Il *Responsabile di progetto* ha il compito di fissare le riunioni, sia interne che esterne. Per ogni riunione sarà necessario specificare data, ora, luogo, e l'ordine del giorno. Le informazioni sulla riunione dovranno essere rese disponibili con almeno quattro giorni di anticipo.

2.3.1 Interne

Sarà il *Responsabile di progetto* a convocare le riunioni generali, ovvero le riunioni a cui devono partecipare tutti i componenti del gruppo. Le riunioni avranno una frequenza almeno quindicinale. Ogni componente del gruppo è tenuto a leggere la posta elettronica e rispondere ad eventuali richieste di riunioni interne.

Il *Responsabile di progetto* può anticipare o posticipare una riunione in base alla disponibilità data dai componenti del gruppo. È possibile richiedere una riunione generale da parte di un qualsiasi componente del gruppo, che verrà inoltrata al *Responsabile di progetto* che può decidere se accoglierla o rifiutarla.

Inoltre, è possibile e auspicabile che siano necessarie riunioni tra specifici membri del gruppo, ad esempio: in fase di analisi può essere utile che solo gli *Analisti* si incontrino tra di loro, senza il resto del gruppo. I restanti componenti del gruppo saranno comunque informati sui contenuti e le decisioni prese tramite invio di una email alla mailing list o alla pubblicazione di un verbale sul repository nel caso siano state prese decisioni importanti.

2.3.2 Esterne

Sarà il *Responsabile di progetto* a fissare le riunioni esterne con i proponenti e/o i committenti utilizzando la casella di posta creata appositamente. Prima di contattare le parti esterne, il *Responsabile di progetto* dovrà verificare la presenza dei membri interni al gruppo.

In caso di risposta positiva si provvederà a contattare i proponenti e/o i committenti per fissare la data dell'incontro. È compito del *Responsabile di progetto* a redigere il verbale dell'incontro avvenuto.



2.4 Verbale

Per verbali si intende quanto si è discusso nel corso di una riunione e le decisioni che eventualmente sono state assunte.

Nella stesura di un verbale occorre essere sintetici, essenziali ed esaustivi, cercando di rappresentare gli interventi, le osservazioni o le decisioni assunte in maniera oggettiva, senza interpretazioni personali, ambigue o fuorvianti. Il tempo dei verbi che deve essere utilizzato è il presente.

I verbali dovranno essere nominati nel seguente criterio:

- Dove e quando si svolge la riunione, i partecipanti, gli assenti, il nome di chi presiede e quello del segretario verbalizzante;
- Tipo di incontro che potrà essere interno o esterno;
- Le questioni o i problemi ancora irrisolti;
- I (nuovi) argomenti posti in discussione all'ordine del giorno;
- Gli interventi più significativi rappresentati in ordine temporale;
- I suggerimenti, le proposte, le conclusioni e le decisioni assunte;
- I compiti assegnati ad ognuno con indicazione delle rispettive scadenze;
- Gli argomenti da trattare nella prossima riunione.

2.5 Repository

In questa sezione sono definiti gli strumenti utilizzati per la condivisione dei file.

Per la gestione della documentazione e della codifica è stato creato un repository. Il repository è privato e quindi accessibile solo dai membri del gruppo *DazzleWorks*.

Per il repository abbiamo scelto di usare il servizio GitHub, il quale utilizza il sistema di versionamento Git.

2.5.1 Utilizzo del servizio

È necessario eseguire le operazioni di sincronizzazione all'inizio e alla fine di ogni sessione di lavoro.

In particolare ad ogni sessione di lavoro andranno eseguite le seguenti operazioni:

- **Pull** - scaricare da la versione più aggiornata del progetto per poterci poi lavorare offline;
- **Commit** - serve a "scattare un'istantanea" della propria repository;
- **Push** - serve a pubblicare i risultati online, sulla piattaforma di sviluppo;
- **Merge** - permette di fondere un branch con il repository madre, così da implementare le modifiche apportate ai file e alle cartelle originarie.

2.5.2 Struttura

Tutta la documentazione si può trovare al seguente indirizzo: <https://github.com/FabioRos/Premi>. Tutti i file riguardanti la documentazione andranno messi in questo repository.

Il repository avrà questa struttura:

- **Norme di Progetto**
- **Analisi dei Requisiti**
- **Piano di progetto**
- **Piano di qualifica**



- Studio di fattibilità
- Glossario
- Lettera di presentazione
- Verbali
- Template



3 Ruoli di Progetto

Lo sviluppo prevede la collaborazione di individui a cui verranno assegnati ruoli diversi. Tali ruoli rappresentano figure aziendali specializzate, indispensabili per il buon esito del progetto. Ogni membro del gruppo dovrà ricoprire ogni ruolo almeno una volta. Si deve controllare attentamente che non vi siano conflitti di interesse specialmente nelle attività di approvazione e verifica. Per garantire che la rotazione dei ruoli non provochi conflitti è necessario che le attività vengano pianificate con attenzione e che i membri interessati rispettino ruoli e compiti loro assegnati. Spetterà al *Verificatore* controllare che tutte le condizioni sopra indicate vengano rispettate. Se il *Verificatore* troverà delle incongruenze con quanto menzionato sopra avrà il compito di avvisare il *Responsabile di Progetto* che dovrà risolvere la questione.

Di seguito i diversi ruoli di progetto con le relative responsabilità e modalità operative.

3.1 Responsabile di Progetto

Il *Responsabile di Progetto* rappresenta il progetto, in quanto accentra su di sé la responsabilità di scelta e approvazione, ed il gruppo, poiché presenta al committente i risultati del lavoro svolto. Detiene il potere decisionale, quindi la responsabilità in merito a:

- Pianificazione, coordinamento e controllo delle attività;
- Gestione e controllo delle risorse;
- Analisi e gestione dei rischi;
- Approvazione dei documenti;
- Approvazione dell'offerta economica.

Di conseguenza, ha il compito:

- Assicurarsi che le attività di verifica e validazione vengano svolte sistematicamente seguendo le *Norme di Progetto*;
- Garantire che vengano rispettati i ruoli e le competenze assegnate nel *Piano di Progetto*;
- Garantire che non vi siano conflitti tra *Verificatori* e *Redattori*;
- Gestire la creazione e l'assegnazione dei ticket ad ogni membro del gruppo.

3.2 Amministratore

L'*Amministratore di Progetto* è il responsabile dell'efficienza e del rendimento dell'ambiente di lavoro. Le mansioni che gli competono sono le seguenti:

- Agevolare le attività richieste dalle *Norme di Progetto v1.0.0*;
- Ricercare e rendere operativi tutti gli strumenti necessari all'automatizzazione del maggior numero di compiti;
- Fornire procedure e strumenti per il controllo e la segnalazione del controllo qualità;
- Gestire l'archiviazione e il versionamento della documentazione;
- Controllare le versioni dei prodotti e gestire le loro configurazioni.

Redige le *Norme di Progetto*, dove spiega e norma l'utilizzo degli strumenti, e redige la sezione del *Piano di Qualifica* dove vengono descritti strumenti e metodi di verifica.



3.3 Analista

L'*Analista* è il responsabile delle attività di analisi. Le mansioni che gli competono sono:

- Comprendere appieno la natura e la complessità del problema;
- Produrre una specifica di progetto motivata in ogni suo punto e comprensibile a tutti gli interessati.

Redige lo *Studio di Fattibilità*, l'*Analisi dei Requisiti* e partecipa alla redazione del *Piano di Qualifica* in quanto conosce l'ambito di progetto.

3.4 Progettista

Il *Progettista* è il responsabile delle attività di progettazione. Le mansioni che gli competono sono:

- Produrre una soluzione comprensibile, attuabile e motivata;
- Effettuare scelte su aspetti progettuali che applichino al prodotto soluzioni note ed ottimizzate;
- Effettuare scelte su aspetti progettuali e tecnologici che rendano il prodotto facilmente manutenibile.

Redige la Specifica Tecnica, la Definizione di Prodotto e le sezioni relative alle metriche di verifica della programmazione del Piano di Qualifica.

3.5 Verificatore

Il *Verificatore* è il responsabile delle attività di verifica. Le mansioni che gli competono sono:

- Garantire che l'attuazione delle attività sia conforme alle norme stabilite;
- Verificare che non siano stati introdotti errori lungo il percorso;
- Controllare la conformità di ogni stadio del ciclo di vita del prodotto.

Redige la sezione del Piano di Qualifica che illustra l'esito e la completezza delle verifiche e delle prove effettuate.

3.6 Programmatore

Il *Programmatore* è responsabile delle attività di codifica e delle componenti di ausilio necessarie per l'esecuzione delle prove di verifica e validazione. Le mansioni che gli competono sono:

- Implementare in maniera rigorosa le soluzioni descritte dal *Progettista*;
- Scrivere codice che sia documentato, versionato, manutenibile e che rispetti le metriche stabilite per la scrittura del codice;
- Implementare i test sul codice prodotto, necessari per le prove di verifica e validazione.

Redige il *Manuale Utente*.



4 Documenti

In questo capitolo si descrivono i vari standard adottati da *DazzleWorks* nella stesura, verifica e approvazione della documentazione da produrre.

4.1 Template

Per semplificare la redazione dei documenti è stato creato un template \LaTeX contenente tutte le impostazioni per l'aspetto grafico.

Ogni documento dovrà essere realizzato con il template \LaTeX presente nel repository, il quale utilizzo è regolamentato da un'apposita guida.

4.2 Versionamento

Per ogni documento è obbligatorio specificare la versione. Un numero di versione deve essere nella seguente forma:

$$X.Y.Z$$

dove X, Y, Z sono interi non negativi e non devono contenere zeri iniziali. X è la versione *Major*, Y è la versione *Minor*, e Z è la versione *Patch*.

- X: la versione Major ($X.y.z \mid X > 0$) identifica la versione di rilascio. Deve essere incrementata se è stata introdotta qualsiasi modifica non retrocompatibile. Le versioni Patch e Minor devono essere reimpostate a 0 quando la versione Major è incrementata.
- Y: la versione Minor ($x.Y.z \mid x > 0$) deve essere incrementata se è stata introdotta una nuova funzionalità. La versione Patch deve essere reimpostata a 0 quando la versione Minor è incrementata.
- Z: la versione Patch ($x.y.Z \mid x > 0$) deve essere incrementata solo se sono state introdotte correzioni retrocompatibili di bug. Una correzione di un bug è definita come una modifica interna che corregge un comportamento errato.

Una volta che un pacchetto versionato è stato rilasciato, i contenuti di quella versione **non devono** essere modificati. Qualsiasi modifica **deve** essere rilasciata come una nuova versione.

4.3 Struttura dei documenti

4.3.1 Prima pagina

Ogni documento deve avere in prima pagina le seguenti informazioni:

- Nome del progetto;
- Logo del gruppo;
- Nome del gruppo;
- Nome del documento;
- La versione del documento;
- Cognome e nome dei redattori del documento;
- Cognome e nome dei verificatori del documento;
- Cognome e nome di chi ha approvato il documento;
- Tipo d'uso del documento;
- Lista di distribuzione del documento;
- Breve descrizione del documento.



4.3.2 Diario delle modifiche

Nella seconda pagina di ogni documento deve essere presente il diario delle modifiche. Ogni riga del diario delle modifiche contiene:

- Una breve descrizione sulle modifiche effettuate;
- Cognome e nome di chi ha effettuato la modifica;
- Data della modifica;
- Versione del documento dopo la modifica.

4.3.3 Indice

In ogni documento deve essere presente un indice delle sezioni.

Nel caso in cui il documento contenga immagini e/o tabelle devono essere presenti anche i relativi indici.

4.3.4 Formattazione delle pagine

L'intestazione di ogni pagina contiene:

- Logo del gruppo;
- La sezione corrente all'interno del documento.

A piè di ogni pagina è presente:

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

4.4 Classificazione documenti

4.4.1 Documenti informali

Si definiscono documenti informali tutti i documenti in fase di sviluppo e devono ancora essere approvati dal *Responsabile di Progetto*. I documenti sono da considerarsi esclusivamente per uso interno e non potranno essere divulgati prima di essere stati verificati ed approvati.

4.4.2 Documenti formali

Si definiscono documenti formali tutti i documenti che sono stati approvati dal *Responsabile di Progetto* e quindi sono pronti ad essere diffusi a terze parti.

I documenti pronti per il rilascio dovranno essere rinominati osservando le seguenti regole:

- La prima lettera di ogni parola, che non sia una preposizione, deve essere maiuscola;
- Gli spazi devono essere sostituiti con il carattere "_" (underscore);
- Tutte le parole devono essere prive di accenti;
- La versione del documento verrà aggiunta dopo il nome, e sarà preceduta dal carattere "-"(trattino) e da una "v".

4.5 Norme tipografiche

Questa sezione racchiude tutte le informazioni riguardanti l'ortografia, la tipografia e l'assunzione di uno stile uniforme in tutti i documenti allo scopo di evitare incoerenze tra le diverse parti.



4.5.1 Punteggiatura

- **Punteggiatura:** qualsiasi carattere di punteggiatura non può seguire un carattere di spazio;
- **Lettere maiuscole:** l'uso delle maiuscole è obbligatorio nelle seguenti situazioni:
 - All'inizio di testo o di una sua parte (capitolo, paragrafo, ecc.);
 - Dopo il punto, il punto esclamativo, il punto interrogativo;
 - All'inizio di ogni elemento di un elenco puntato;
 - Per i ruoli, di progetto, i nomi dei documenti, le fasi di progetto, revisioni di progetto oltre che dove previsto dalla lingua italiana.
- **Parentesi:** il testo racchiuso dentro le parentesi non deve mai iniziare con un carattere di spazio e non deve mai terminare con un carattere di spazio o punteggiatura.

4.5.2 Stile del testo

- **Maiuscolo:** l'uso delle maiuscole è limitato alla trascrizione degli acronimi;
- **Grassetto:** l'uso del grassetto deve essere utilizzato nei seguenti casi:
 - **Elenchi puntati:** per evidenziare l'oggetto trattato;
 - **Parole chiavi;**
- **Corsivo:** l'uso del corsivo deve essere utilizzato nei seguenti casi:
 - **Citazioni:** ogni citazione deve essere scritta in corsivo;
 - **Documenti:** ogni riferimento ad un documento deve essere scritto in corsivo (esempio: *Glossario*);
 - **Ruoli:** ogni riferimento a figure particolari deve essere scritto in corsivo (esempio: *Verificatore*);
 - **Altri casi:** è possibile utilizzare il corsivo per evidenziare parole particolarmente significative.
- **L^AT_EX:** ogni riferimento a L^AT_EX va scritto tramite il comando `\LaTeX`.

4.5.3 Composizione del testo

- **Elenchi puntati:** per redigere un elenco si qualificano gli elementi e si introducono mediante i due punti. Dopo i due punti si deve andare a capo. Ogni elemento dell'elenco termina con un punto e virgola, eccetto l'ultimo che termina con un punto semplice;
- **Pedice "G":** il pedice "G" viene utilizzato in presenza di termini presenti nel *Glossario*;
- **Note a piè di pagina:** ogni nota deve iniziare con la lettera della prima parola maiuscola e non deve essere preceduta da alcun carattere di spazio. Ogni nota deve terminare con un punto.

4.5.4 Formati ricorrenti

- **Nomi propri:** l'utilizzo dei nomi propri deve seguire la seguente forma "Cognome Nome";
- **Percorsi:** per gli indirizzi email e gli indirizzi web deve essere usato esclusivamente il comando `LATEX\url`;
- **Date:** ogni data deve seguire lo standard internazionale per date ed orari ISO 8601:

AAAA.MM.GG



dove:

- AAAA: rappresenta il formato dell'anno scritto con quattro cifre;
- MM: rappresenta il formato del mese scritto con due cifre;
- GG: rappresenta il formato del giorno scritto con due cifre.
- Riferimenti ai documenti: ci si riferirà ai vari documenti scrivendo in corsivo il nome del documento e mettendo una lettera maiuscolo per ogni parola che non sia un articolo. Nel caso in cui ci si deve riferire ad una versione specifica del documento, essa andrà indicata alla fine del nome del documento (esempio: *Norme di Progetto v1.0.0*);
- Nome del gruppo: ci si riferirà al gruppo solo come "*Dazzle Works*" con il nome del gruppo in corsivo. Per una scrittura corretta del nome è stato creato il comando `LATEX \GRUPPO`;
- Nome del progetto: ci si riferirà al nome del progetto solo come "**PREMI**". Per una scrittura corretta del nome è stato creato il comando `LATEX \PROGETTO`.

4.5.5 Sigle

Le Sigle potranno comparire esclusivamente all'interno delle tabelle e diagrammi. Nello specifico quelle consentite sono:

- Documenti:
 - NdP = *Norme di Progetto*;
 - AdR = *Analisi dei Requisiti*;
 - PdP = *Piano di Progetto*;
 - PdQ = *Piano di Qualifica*;
 - SdF = *Studio di Fattibilità*.
- Revisioni:
 - RR = *Revisione dei Requisiti*;
 - RP = *Revisione di Progettazione*;
 - RQ = *Revisione di Qualifica*;
 - RA = *Revisione di Accettazione*.

4.6 Componenti grafiche

4.6.1 Immagini

Tutte le immagini dovranno avere il formato PDF oppure PNG. In modo da garantire una maggiore qualità dell'immagine in caso di ridimensionamento è consigliato usare il formato PDF vettoriale. La conversione di un'immagine in formato PDF è possibile attraverso il software GIMP³, il quale è stato già usato da tutti i componenti del gruppo.

Le immagini devono essere accompagnate da una didascalia che inizia con la parola "Figura", seguita dal numero della figura, dal carattere "due punti" e da una breve descrizione.

4.6.2 Tabelle

Ogni tabella deve essere accompagnata da una didascalia che inizia con la parola "Tabella", seguita dal numero della tabella, dal carattere "due punti" e da una breve descrizione non banale.

³GNU Image Manipulation Program è un programma per la creazione e modifica di immagini digitali.



4.7 Procedure di avanzamento di un documento

Nel momento in cui il redattore ritenga di aver concluso la stesura di un documento, si dovranno seguire i seguenti passi nell'ordine esatto di comparizione:

1. Il redattore ha il compito di contattare il *Responsabile di Progetto* per informarlo della terminazione della stesura del documento;
2. Il *Responsabile di Progetto* provvederà a contattare ed assegnare la verifica del documento ad un verificatore disponibile, che non sia in conflitto di interessi;
3. Il *Verificatore* provvederà alla verifica del documento;
4. Nel caso in cui vengano trovati errori nel documento, il *Verificatore* provvederà alla creazione di un ticket di correzione e lo assegnerà al redattore, il quale correggerà gli errori trovati e infine si tornerà di nuovo al passo uno;
5. Se non sono stati trovati errori, il documento verrà approvato dal *Responsabile di Progetto*.

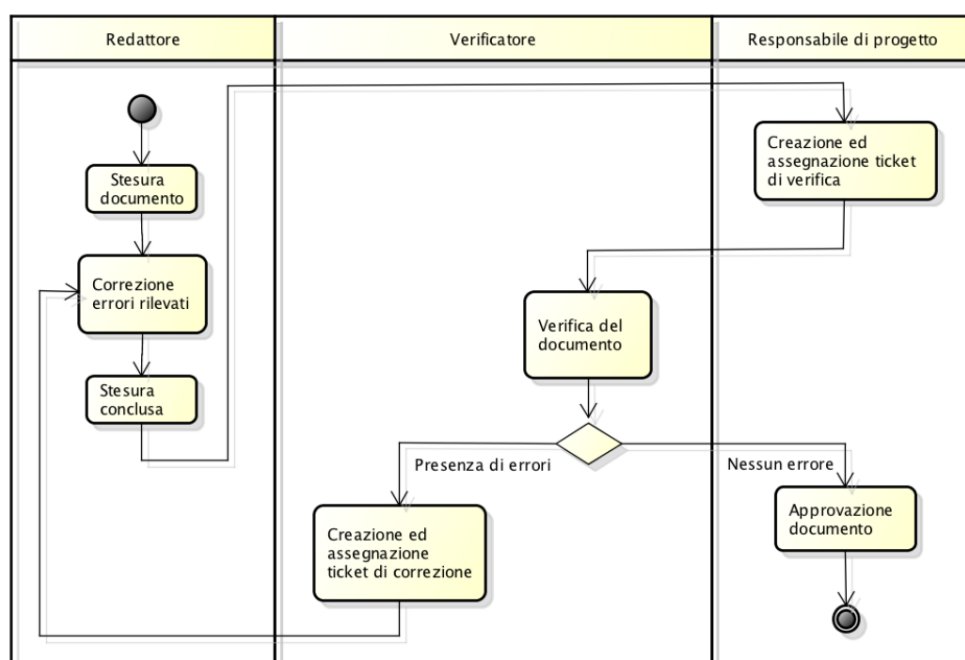


Figura 1: Procedura avanzamento documento



5 Ambiente Lavoro

In questa sezione verrà illustrato l'ambiente di lavoro che sarà utilizzato durante lo sviluppo del progetto **Premi**.

5.1 Sistema operativo

Il sistema operativo utilizzato è lasciato a discrezione di ogni membro del gruppo. Questa scelta è dovuta principalmente al fatto che il progetto dovrà supportare più piattaforme. I membri del gruppo utilizzeranno i seguenti sistemi operativi:

- Windows 7 64 bit;
- Ubuntu 14;
- Mac OS X.

5.2 Coordinamento

Il coordinamento del gruppo avviene tramite:

- Repository Git;
- Google Drive;
- Google Calendar;

5.2.1 Repository Git

Nonostante siano disponibili molti repository (Git, Mercurial, SVN) è stato scelto di utilizzare Git in quanto il servizio soddisfa pienamente le necessità di hosting e versionamento necessarie per lo sviluppo di questo progetto, inoltre diversi membri del gruppo avevano già usato tale repository. Esso ci permette di lavorare senza una connessione attiva a internet e ci dà la possibilità di ignorare certe estensioni attraverso un `.gitignore` file⁴.

Per i membri del gruppo c'è la possibilità di utilizzare i client. Si consigliano i seguenti:

- **SourceTree**;
- **GitHub**;

Per maggiori dettagli si rimanda alla sezione 2.5.

5.2.2 Google Drive

Abbiamo scelto di utilizzare questo servizio cloud per condividere tutti i documenti che non necessitano un controllo di versione e vengono utilizzati frequentemente da parte dei membri del gruppo. È un servizio molto semplice e accessibile direttamente da browser che permette di lavorare su documenti creati con Google Docs.

5.2.3 Google Calendar

Google Calendar viene utilizzato per la gestione delle risorse umane e per tenere traccia degli eventi importanti; infatti è stato creato un calendario condiviso con tutti i membri del gruppo in modo da conoscere le date in cui una persona è assente o non reperibile e le date importanti relativi al gruppo.

Inoltre è possibile modificare la gestione delle notifiche per un evento o più eventi in modo che venga ricevuta una email da parte di tutti i membri del gruppo il giorno prima.

⁴File globale di Git che contiene una lista di regole per ignorare i file.



5.3 Ambiente documentale

5.3.1 Stesura documenti

Per la stesura dei documenti verrà utilizzato il linguaggio di markup $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$. Come editor è consigliato TeXstudio⁵, il quale è disponibile per tutti i principali sistemi operativi.

5.3.2 Script

Per facilitare la stesura dei documenti sono stati creati alcuni script:

- Generazione di tutti i documenti PDF: con il comando `make all` verranno generati tutti i PDF dei documenti contenuti nella directory corrente;
- Evidenziare glossario: con il comando `java glossary` verrà eseguito uno script che selezionerà nei documenti, della directory corrente, le parole contenute nella versione più recente del glossario e le evidenzierà con il simbolo "*g*".

5.3.3 Pianificazione delle attività

Per pianificare le attività e la gestione del progetto e delle risorse umane si è scelto di usare GanttProject⁶.

5.4 Ambiente di sviluppo

"Descrizione IDE e Framework vari."

5.5 Diagrammi UML

Per la realizzazione dei diagrammi UML è stato scelto Astah Professional Edition⁷. E' possibile ricevere gratuitamente la licenza della versione professional inviando una richiesta al sito <http://astah.net/student-license-request>.

5.6 Verifica

5.6.1 Verifica dei documenti

La verifica dei documenti verrà eseguita ogni volta che è stata effettuata una modifica ad un documento e debba essere approvato. Per una corretta verifica di un documento vanno seguite le seguenti pratiche:

- **Controllo tipografico:** tramite l'utilizzo di TeXstudio verranno individuati errori tipografici presenti nel documento;
- **Controllo lessicale:** il *Verificatore* dovrà controllare che il documento non presenti errori lessicali attraverso un'attenta analisi del testo utilizzando la tecnica inspection o walkthrough;
- **Controllo glossario:** il *Verificatore* dovrà controllare che ogni parola, nel testo, presente nel glossario sia correttamente evidenziata;
- **Controllo contenuto:** il *Verificatore* dovrà controllare che il documento contenga tutto il necessario e che sia impaginato adeguatamente;
- **Rispetto delle norme del progetto:** il *Verificatore* dovrà controllare che il documento segua le norme di progetto stabilite;

⁵<http://texstudio.sourceforge.net>

⁶<http://www.ganttproject.biz>

⁷<http://astah.net/download>



- **Lista di controllo:** si dovrà stilare una lista di errori più frequenti, per semplificare le successive verifiche dei documenti;
- **Rispetto indice Gulpease:** il *Verificatore* dovrà calcolare e controllare, per ogni documento, che gli indici di Gulpease risiedano nel range di valori specificato nel *Piano di Qualifica*, altrimenti si dovrà effettuare una ricerca walkthrough alla ricerca delle frasi troppo lunghe e complesse;
- **Segnalazione errori:** una volta completata la verifica di un documento, se sono stati riscontrati errori, il *Verificatore* dovrà aprire dei task per segnalarli.

5.7 Verifica dei diagrammi

I diagrammi devono essere verificati manualmente dal *Verificatore* e deve controllare che aderiscano correttamente allo standard UML 2.0. In particolare deve controllare che i diagrammi di flusso siano rappresentati in maniera corretta e che i casi d'uso utilizzino correttamente le inclusioni e le estensioni.



6 Protocolli e Gestione di Progetto

Il servizio GitHub fornisce un sistema per la gestione dei task e creazione di milestone. Si è quindi deciso di utilizzare questo servizio.

6.1 Creazione e gestione dei ticket

I ticket vengono creati e gestiti quasi tutti dal responsabile di progetto. Qualora il *verificatore* trovasse imprecisioni o errori durante la verifica, avrà la possibilità di creare dei ticket per segnalare suddetti errori. Ogni ticket può essere assegnato ad uno o più membri del gruppo a seconda della complessità del lavoro e della disponibilità dei membri del gruppo. Per creare un nuovo ticket bisogna:

- Andare su Issue;
- Premere "New issue";
- Compilare i campi richiesti:
 - **Titolo e commenti:** titolo descrittivo con una descrizione del nuovo ticket;
 - **Labels:** dovranno avere due caratteristiche fondamentali e cioè la tipologia di ticket assegnato(modifica, verifica, richiesta approvazione) e la priorità assegnata(alta, media, bassa);
 - **Milestone:** la milestone a cui è associato il ticket;
 - **Assignee:** a chi viene assegnato il ticket.

Figura 2: Creazione ticket

6.2 Creazione delle milestone

Il responsabile di progetto ha il compito della creazione di una milestone in occasione di ogni revisione al quale il gruppo *DazzleWorks* ha intenzione di partecipare, più altre milestone qualora il responsabile di progetto lo ritenga necessario.

Per creare una nuova milestone bisogna:

- Andare su milestone;
- Premere "New milesone";
- Compilare i campi richiesti:
 - **Titolo;**
 - **Descrizione;**
 - **Data.**



New milestone
Create a new milestone to help organize your issues and pull requests. Learn more about [milestones and issues](#).

Title

Description

Due Date (optional) [clear](#)

◀ April ▶ ◀ 2015 ▶

Mon	Tue	Wed	Thu	Fri	Sat	Sun
30	31	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	1	2	3

Create milestone

Figura 3: Creazione milestone

6.3 Esecuzione dei compiti

Ogni membro del gruppo è tenuto a visionare regolarmente la presenza di ticket a lui assegnati e segnalarne la presa in consegna. Una volta che un membro porta a termine un ticket deve modificarne lo stato per segnalare il termine del lavoro. Se un ticket non ha avuto i risultati attesi il *Responsabile di Progetto* può riaprirlo ed eventualmente assegnare altri membri al lavoro.

6.4 Chiusura della milestone

Una volta raggiunta la scadenza, il *Responsabile di Progetto*, deve chiudere la milestone ed eventualmente aprirne un'altra per poi ricominciare tutto il protocollo da capo.



7 Analisi Requisiti

E' compito degli *Analisti* stilare il documento *Analisi dei requisiti*. In questo documento dovranno essere presenti tutti i requisiti ed i casi d'uso emersi dall'analisi del capitolato e dalle riunioni con il Proponente.

7.1 Requisiti

I requisiti dovranno essere classificati per tipo e priorità, utilizzando la seguente notazione:

R{importanza}{tipo}{codice}

- Importanza può assumere i seguenti valori:
 - OBB : Requisito obbligatorio;
 - DES : Requisito desiderabile;
 - OPZ : Requisito opzionale.
- Tipo può assumere i seguenti valori:
 - F : Requisito funzionale;
 - Q : Requisito di qualità
 - P : Requisito di prestazione;
 - V : Requisito di vincolo.
- Codice rappresenta il codice univoco di ogni requisito in forma gerarchica.

Ogni requisito dovrà essere inserito (nel documento Analisi dei Requisiti) in una tabella contenente l'identificativo, una breve descrizione e la fonte.

7.2 Casi d'uso

Dopo la stesura dei requisiti è sempre compito degli analisti analizzare i casi d'uso (abbreviati con UC, use case). Per ogni caso d'uso è richiesto:

- Codice identificativo: nel formato UCxx, dove xx indica un numero identificativo del caso d'uso;
- Titolo: titolo sintetico del caso d'uso;
- Descrizione: una breve descrizione chiara e dettagliata del caso d'uso;
- Diagramma: dovrà essere usato UML 2.4 per la creazione dei diagrammi dei casi d'uso;
- Attori: gli attori coinvolti nel caso d'uso (principali e secondari);
- Precondizione: la precondizione del requisito;
- Postcondizione: la postcondizione del requisito.

7.3 Tracciamento

Per il tracciamento dei requisiti è stato creato un database, accessibile via web, che associa ad ogni caso d'uso i corrispettivi requisiti e viceversa.



8 Progettazione

Dopo la fase di **Analisi** si passerà alla fase di **Progettazione** dove i *Progettisti* dovranno seguire le seguenti regole (organizzate per macro-aree).

8.1 Diagrammi

Dovrà essere utilizzato il linguaggio UML per la realizzazione dei seguenti diagrammi:

- Diagrammi delle classi;
- Diagrammi dei package;
- Diagrammi di attività;
- Diagrammi di sequenza.

8.2 Design Pattern

I *Progettisti* dovranno utilizzare il design pattern che ritengono più adatto al contesto per rendere l'applicazione più efficiente possibile.

8.3 Classi di verifica

Andranno create delle classi di verifica per testare che i componenti abbiano un comportamento corretto.

8.4 Stile di progettazione

Durante la fase di **Progettazione** bisognerà fare attenzione a:

- **Ricorsione:** non dovrà essere utilizzata la ricorsione a meno che non sia necessaria, in caso dovrà essere fornita una dimostrazione induttiva sulla correttezza del metodo in questione;
- **Annidamento di cicli:** all'interno di un metodo non dovranno esserci cicli annidati con una profondità maggiore a cinque.



9 Codifica e Convenzioni

9.1 Linguaggi di codifica

Dopo un'analisi del capitolato d'appalto e dei requisiti si è deciso che per lo sviluppo del software richiesto si utilizzeranno i linguaggi HTML e Javascript.

9.2 Convenzioni di codifica

Di seguito è riportato l'insieme di norme e convenzioni che il gruppo dovrà seguire nella scrittura e documentazione del codice. L'unica lingua ammessa per i nomi di variabili, classi, metodi e commenti è l'inglese.

9.3 Nomenclatura

Per l'assegnazione di nomi a variabili, metodi e costanti andranno seguite le seguenti regole:

- Funzioni: va utilizzata la notazione mixed case, con la prima lettera minuscola;
- Variabili: va utilizzata la notazione mixed case, con la prima lettera minuscola;
- Costanti: va scritto il nome interamente in maiuscolo, separando le varie parole con il carattere "_" (underscore).

9.4 Intestazione di un file Javascript

/*

File
Autore
Data
Descrizione

Modifiche:

Versione	-	Data	-	Programmatore	-	Modifica	-	Descrizione
x.y.z	-	aaaa-mm-gg	-	Nome Cognome	-	Funzione	-	Descrizione modifica

*/

- File : nome del file;
- Autore : nome e cognome del creatore del file;
- Data : data di creazione del file nel formato aaaa-mm-gg;
- Descrizione : poche righe di descrizione delle funzionalità contenute nel file;
- Cambiamenti : tabella dello stato di avanzamento del file, contenente tutte le modifiche effettuate :
 - Versione : versione una volta effettuata la modifica;
 - Data : data della modifica;
 - Programmatore : nome e cognome del programmatore che ha effettuato la modifica;
 - Modifica : segnatura della funzione a cui è stata apportata una modifica;
 - Descrizione : breve descrizione della modifica effettuata.



9.5 Commenti

Prima di ogni funzione dovrà essere presente un commento con la seguente forma:

```
/*
```

Descrizione della funzione

Descrizione dei parametri

Descrizione del tipo di ritorno

```
*/
```

Ogni variabile di particolare importanza dovrà essere fornita di commento che ne spieghi il suo scopo e funzionamento.



A Lista di controllo

Di seguito verrà presentata la lista degli errori più comuni trovati durante le verifiche dei documenti:

- **Norme stilistiche:** la conoscenza **non** approfondita delle norme per la stesura dei documenti potrebbe portare a errori:
 - Nel caso degli elenchi potrebbe non terminare con il "." nel caso dell'ultimo elemento oppure potrebbe mancare il ";" per uno o più elementi tranne l'ultimo;
 - Potrebbe mancare la marcatura "G" in pedice per i termini presenti nel *Glossario*;
 - Il non utilizzo del grassetto per le fasi principali del progetto, il corsivo per i documenti e i ruoli di progetto;
 - Le note a piè di pagina che non iniziano con una maiuscola e non finiscono con il ".".
- **Lingua italiana:**
 - Più tempi verbali all'interno della stessa frase;
 - Utilizzo di termini con significato ambiguo.
- **L^AT_EX:** non viene considerato il carattere di spaziatura dopo l'inserimento dei comandi L^AT_EX;
- **Altro:** errori dovuti a distrazioni e/o errori di battitura.