

NORRIS Framework



FLAMETECH Inc.

Norme di Progetto

Informazioni sul documento

Versione	1.4.0
Redazione	Faggin Andrea
Verifica	Merlo Gianluca
Responsabile	Sartor Michele
Uso	Interno
Lista di distribuzione	FlameTech Inc. Prof. Vardanega Tullio

Descrizione

Regole adottate dal gruppo **FlameTech Inc.** durante tutto lo svolgimento del progetto



Stato	Modifica	Autore	Ruolo	Data	Versione
Approvato	Documento approvato	Sartor Michele	Responsabile	2014/12/01	1.4.0
Verificato	Documento verificato	Merlo Gianluca	Verificatore	2014/12/01	1.3.0
In Lavorazione	Stesura sezione documenti	Faggin Andrea	Amminis.	2014/11/28	1.2.1
Verificato	Documento verificato	Merlo Gianluca	Verificatore	2014/11/28	1.2.0
In Lavorazione	Stesura Ambiente di lavoro, Processi Organizzativi e Riunioni	Faggin Andrea	Amminis.	2014/11/27	1.0.2
In Lavorazione	Stesura sezione comunicazioni	Faggin Andrea	Amminis.	2014/11/26	1.0.1
In Lavorazione	Creto scheletro del documento, iniziata stesura	Faggin Andrea	Amminis.	2014/11/26	1.0.0

Indice

1	Introduzione	1
1.1	Scopo del documento	1
1.2	Scopo del prodotto	1
1.3	Glossario	1
1.4	Riferimenti	1
1.4.1	Normativi	1
1.4.2	Informativi	1
2	Comunicazioni	2
2.1	Comunicazioni esterne	2
2.2	Comunicazioni interne	2
2.3	Composizione e-mail	2
2.3.1	Mittente	2
2.3.2	Destinatario	2
2.3.3	Oggetto	2
2.3.4	Corpo dell'e-mail	2
2.3.5	Allegati	3
3	Riunioni	4
3.1	Frequenza delle riunioni	4
3.2	Convocazione della riunione	4
3.3	Svolgimento della riunione	4
3.4	Verbale	4
3.4.1	Riunione Interna	4
3.4.2	Riunione Esterna	5
4	Documenti	6
4.1	<i>Template_G</i>	6
4.2	Struttura del <i>Template_G</i>	6
4.2.1	Prima Pagina	6
4.2.2	Diario delle modifiche	6
4.2.3	Indici	7
4.2.4	Struttura delle pagine	7
4.3	Norme Tipografiche	7
4.3.1	Stile del testo	7
4.3.2	Formati	8
4.4	Immagini	8
4.5	Tabelle	8
4.6	Elenchi	8
4.7	Ciclo di vita del documento	9
4.8	Versionamento	9
4.9	Norme sui nomi dei file	10
4.10	Classificazione dei documenti	10
4.10.1	Documento informale	10
4.10.2	Documento formale	10



5	Processi Organizzativi	11
5.1	Rotazione dei ruoli	11
5.1.1	Responsabile	11
5.1.2	Amministratore	11
5.1.3	Analista	12
5.1.4	Progettista	12
5.1.5	Programmatore	12
5.1.6	Verificatore	12
5.2	Strumenti e Ambiente di lavoro	12
5.2.1	Sistema operativo	13
5.2.2	Coordinamento e gestione del progetto	13
5.2.2.1	Creazione di un ticket	13
5.2.3	Software per il versionamento	13
5.2.4	Software per la pianificazione	14
5.2.5	Software per il tracciamento	14
5.2.6	Software creazione diagrammi UML	14
5.2.7	Software creazione delle presentazioni	14
5.2.8	Altri software utilizzati	14
5.3	Repository	15
5.3.1	Struttura del <i>repository_G</i>	15
5.3.2	Norme sui <i>commit_G</i>	15
5.4	Codifica e Convenzioni	15
5.4.1	Nomi	16
5.4.2	File contenente del codice	16
5.4.3	Ricorsione	16
6	Procedure di supporto	17
6.1	Gestione del progetto	17
6.1.1	Pianificazione attività	17
6.1.2	Pianificazione e controllo delle attività	17
6.1.3	Controllo e gestione delle risorse	17
6.1.4	Controllo e gestione dei rischi	17
6.2	Analisi dei rischi e Studio di Fattibilità	17
6.3	Analisi dei Requisiti	18
6.3.1	Classificazione dei requisiti	18
6.3.2	<i>Casi d'uso_G</i>	19
6.3.3	Tracciamento dei Requisiti e dei <i>Casi d'Uso_G</i>	19
6.4	Progettazione	19
6.4.1	Diagrammi <i>UML_G</i>	19
6.4.2	<i>Design Pattern_G</i>	19
6.5	Verifica	20
6.5.1	Strumenti per la verifica	20
6.5.2	Verifica dei processi	20
6.5.3	Verifica dei documenti	20
6.5.4	Verifica tracciamento dei requisiti	21
6.5.5	Verifica UML	21
7	Glossario	23
7.1	Inserimento del termine	23
7.2	Ordine di apparizione	23

Elenco delle figure

1	Procedura per l'analisi dei documenti	21
---	---	----

1 Introduzione

1.1 Scopo del documento

Lo scopo del documento è definire le norme che i componenti del gruppo **FlameTech Inc.** adotteranno durante la realizzazione del progetto Norris.

Tutti i componenti del gruppo sono tenuti a leggere il documento ed applicare le norme descritte allo scopo di migliorare l'uniformità del materiale prodotto, l'efficienza del lavoro e di ridurre al minimo il numero di errori. Le norme descritte in questo documento riguardano:

- convenzioni da usare all'interno dei documenti;
- interazioni tra i componenti del gruppo;
- interazioni con i proponenti;
- modalità di lavoro;
- l'ambiente su cui verrà sviluppato il prodotto;

1.2 Scopo del prodotto

Lo scopo del prodotto è la realizzazione di un *framework_G* per *Node.js_G*, compatibile con l'utilizzo standard dei *middleware_G* di *Express.js_G* in versione 4.x, per la realizzazione rapida di *client web* per la visualizzazione di grafici aggiornabili in tempo reale.

1.3 Glossario

Per evitare ogni possibile ambiguità che potrebbe sorgere verrà allegato il *Glossario_ver1.4.0* dove verranno inseriti termini tecnici, acronimi, termini di dominio ed eventuali parole che potrebbero comportare delle incomprensioni e delle ambiguità nella lettura dei documenti. Per rendere la lettura più facile i termini verranno riportati in corsivo ed in pedice verrà posta una "G" maiuscola. (Esempio: *Android_G*).

1.4 Riferimenti

1.4.1 Normativi

- **Analisi dei Requisiti:** *AnalisiRequisiti_ver1.4.0*;
- **Piano di Progetto:** *PianoDiProgetto_ver1.4.0*;
- **Piano di Qualifica:** *PianoDiQualifica_ver1.4.0*;
- **Studio di Fattibilità:** *StudioDiFattibilità_ver1.4.0*.

1.4.2 Informativi

- **Github_G:** <http://www.github.com>;
- **Hostedredmine:** <https://www.hostedredmine.com/>.

2 Comunicazioni

2.1 Comunicazioni esterne

Per le comunicazioni del gruppo con l'esterno è stato creato un apposito indirizzo di posta elettronica:

flametech.inc@gmail.com

L'e-mail sarà l'unico modo con cui il gruppo comunicherà con l'esterno. L'unica persona che avrà accesso all'indirizzo e-mail sarà il *Responsabile di Progetto*, quindi sarà lui l'unico a comunicare con il Committente. Starà poi sempre allo stesso *Responsabile* il compito di comunicare agli altri componenti del team il contenuto delle discussioni ed eventualmente inoltrarne le conversazioni.

2.2 Comunicazioni interne

Le comunicazioni interne saranno tenute tramite un gruppo privato nel *social network* *Facebook*_G. Tale gruppo sarà nominato [SWE] Norris, e sarà visibile ai soli membri del team. Per le comunicazioni veloci verrà utilizzata una chat di gruppo privata per i soli membri del team usando il software *Telegram*_G. Tale chat sarà nominata FlameTech Inc.. In caso di decisioni prese tramite questo metodo di comunicazione il *Responsabile* dovrà notificare in modo permanente le scelte effettuate tramite un post permanente nel gruppo [SWE] Norris. Nel caso sia richiesto l'utilizzo di una chat o di una videoconferenza, verrà utilizzata l'applicazione *Hangouts*_G.

2.3 Composizione e-mail

Per tutte le e-mail si è deciso di fare riferimento alla struttura riportata di seguito:

2.3.1 Mittente

Riguardo alle e-mail indirizzate all'esterno del gruppo, per esempio al Committente o Proponente, deve essere usata obbligatoriamente l'e-mail flametech.inc@gmail.com, che può essere inviata solamente dal *Responsabile*. Per le e-mail interne ogni componente userà il proprio indirizzo.

2.3.2 Destinatario

Una e-mail indirizzata all'esterno del gruppo può essere al Proponente, al Prof. Vardanega Tullio o al Prof. Cardin Riccardo.

2.3.3 Oggetto

L'oggetto di ogni e-mail deve essere breve, possibilmente non già usato in altre comunicazione ed il più esaustivo possibile. Per qualsiasi conversazione già avviata non è possibile modificarne l'oggetto.

2.3.4 Corpo dell'e-mail

Nel corpo dell'e-mail bisogna essere il più esaustivi possibile ed inserire tutte le informazioni che sono necessarie per fare in modo che il messaggio risulti chiaro e facilmente

comprensibile al destinatario. Nel caso si tratti di una risposta o di un inoltra è obbligatorio aggiungere il contenuto in testa alla e-mail ed è consigliabile non cancellare il resto del testo così da avere sempre a portata di mano tutta la conversazione avvenuta fino a quel momento.

2.3.5 Allegati

L'uso degli allegati è concesso ma deve essere usato il meno possibile e solamente nel caso ve ne sia necessità.

3 Riunioni

3.1 Frequenza delle riunioni

Il *Responsabile* ha stabilito che le riunioni interne saranno organizzate con una cadenza quindicinale. In casi di necessità o su richiesta da parte di alcuni membri del gruppo, previa conferma del *Responsabile*, potranno essere indette ulteriori riunioni. Diversamente le riunioni esterne verranno convocate dal *Responsabile* in accordo con il Committente, previa la disponibilità dello stesso, quando vi sarà la necessità di chiarimenti o di spiegazioni riguardo al progetto.

La data della riunione sarà comunicata dal *Responsabile* con almeno due giorni di anticipo in modo da permettere ai membri del gruppo di essere disponibili ed in caso di mancata disponibilità di raggiungere un accordo per un'occorrenza successiva.

3.2 Convocazione della riunione

La convocazione di una riunione deve essere comunicata in via ufficiale dal *Responsabile* tramite il gruppo su *Facebook_G* del team, per cui visibile a tutti i membri che dovranno confermare la presenza. La convocazione di una riunione potrà essere richiesta anche da uno dei componenti del team tramite l'invio di una comunicazione al *Responsabile*, il quale avrà poi il compito di decidere se accettare la proposta oppure di rigettarla, prendendo in considerazione il fatto segnalato per la successiva riunione periodica.

Le riunioni che verranno organizzate non comprenderanno necessariamente tutti i componenti del team ma potranno comprenderne solamente alcuni.

Ad ogni riunione è prevista la realizzazione di un verbale la cui condivisione fra i componenti del team permetterà di aggiornare anche gli assenti sulle decisioni prese e gli argomenti trattati. Vedi sez. 3.4.

3.3 Svolgimento della riunione

All'inizio di ogni riunione, sia questa interna oppure esterna, dovrà essere controllata la presenza di tutti i membri previsti e, fra di essi, dovrà essere scelto un segretario con il compito di annotare ogni argomento e domanda che viene posta in maniera tale da poter poi redigere in maniera consona il verbale della riunione.

Durante la riunione è chiesto a tutti i presenti di tenere un comportamento corretto ed adeguato al luogo in cui si trovano così da poter svolgere al meglio la riunione. Il segretario ha inoltre il compito di controllare che tutti i punti che erano stati comunicati all'ordine del giorno vengano discussi e non venga tralasciato nulla.

3.4 Verbale

3.4.1 Riunione Interna

Il verbale di una riunione interna in cui prendono parte solo i componenti del gruppo sarà un documento informale che servirà a tenere traccia degli argomenti discussi durante la riunione. Il documento redatto verrà poi condiviso a tutti i componenti del gruppo **FlameTech Inc.** vedi sez. 4.10.1.

3.4.2 Riunione Esterna

Il verbale di una riunione esterna sarà invece un documento formale ed ufficiale che potrà avere un valore normativo. Tale documento sarà caricato nel *repository_G* all'interno della cartella denominata come **esterna**.

4 Documenti

In questo capitolo verranno descritte quali saranno le scelte tipografiche che il gruppo ha deciso di seguire nella realizzazione dei documenti.

4.1 *Template_G*

Per facilitare la verifica e uniformare la stesura dei documenti si è deciso di ricorrere alla realizzazione di un *template_G* L^AT_EX_G, il quale contiene tutte le impostazioni tipografiche che si è scelto di adottare. Questo si trova nel *repository_G* vedi sez. 5.3. La scelta di creare un *template_G* è stata fatta perché qualora si decidesse di apportare una modifica alla struttura dei documenti è sufficiente effettuarla su di un singolo *template_G* affinché questa poi venga acquisita da tutti i file che ne fanno uso.

4.2 Struttura del *Template_G*

Il *template_G* è stato realizzato con la seguente struttura:

4.2.1 Prima Pagina

In prima pagina sono contenute le seguenti informazioni:

- logo del gruppo;
- titolo del documento;
- nome del gruppo;
- versione del documento;
- lista dei redattori;
- lista dei verificatori;
- nome del Responsabile che ha approvato il documento;
- destinazione del documento (uso interno/esterno);
- lista di distribuzione;
- descrizione del contenuto all'interno del documento.

4.2.2 Diario delle modifiche

Nella seconda pagina del documento è contenuta una tabella contenente il diario delle modifiche che vengono apportate al documento. Le informazioni riportate per ogni modifica sono:

- stato del documento vedi sez. 4.7;
- descrizione della modifica;
- autore della modifica;
- il ruolo della persona che ha modificato il documento;

- data in cui è avvenuta la modifica;
- versione del documento.

Le modifiche devono essere riportate in tabella secondo l'ordine decrescente di versione del documento. Inoltre deve essere aggiornata la versione del documento secondo le norme riportate in sez. 4.8. Questo comporta anche che sia aggiornata la versione del documento nel frontespizio qualora c'è ne sia la necessità.

4.2.3 Indici

Nel *template_G*, dopo il diario delle modifiche c'è la possibilità di inserire o meno i seguenti indici :

- indice delle pagine;
- indice delle tabelle;
- indice delle figure.

L'inserimento dei seguenti indici è dato dalla presenza o meno di tabelle e/o figure all'interno del documento, qualora non fossero presenti tabelle o figure è sufficiente settare a false il flag di caricamento dell'indice corrispondente.

4.2.4 Struttura delle pagine

Ogni pagina interna del documento avrà la seguente struttura:

- **Intestazione:** nella parte sinistra il logo del gruppo ed il nome mentre nella parte destra conterrà il numero e il nome della sezione;
- **Piè di pagina:** nella parte sinistra il nome del documento e la sua versione mentre nella parte destra è presente il numero della pagina. In numero di pagina è espresso nel formato: [numeroPagina] di [numeroPagineTotali];

4.3 Norme Tipografiche

Questa sezione contiene le scelte in fatto di tipografia, ortografia e stile che dovranno essere attuate durante la stesura dei documenti.

4.3.1 Stile del testo

- **Grassetto:** deve essere usato nel seguente caso:
 - **elenchi:** per evidenziare la descrizione dell'elenco;
 - **header delle tabelle**
- **Corsivo:** solo nei seguenti casi:
 - **documenti:** per citare un documento (es. *NormeDiProgetto_ver1.4.0*);
 - **ruoli:** per nominare uno dei ruoli del gruppo (es. *Analista*);
 - **abbreviazioni:** quando c'è bisogno di inserire una abbreviazione di qualche termine;
 - **termini in glossario:** i termini presenti in glossario;

- ***URL_G***: Per l’inserimento di indirizzi web verrà utilizzato l’apposito tag \LaTeX_G .

- **Maiuscolo:**

- l’inserimento di un termine tutto in maiuscolo è possibile solamente nel caso si tratti di acronimi;
- per delle macro \LaTeX_G create appositamente per velocizzare la stesura dei documenti.

4.3.2 Formati

- **Data:** le date dovranno adottare il formato aaaa/mm/gg, dove aaaa indica l’anno per esteso, mm il mese, gg il giorno;
- **Ruoli di progetto:** in corsivo, con la prima lettera maiuscola e poi tutto minuscolo;
- **Nomi propri:** i nomi dei componenti del team dovranno essere nella forma “Cognome Nome”;
- **Nome del gruppo:** per inserire il nome del gruppo usare l’apposita macro $\backslash\text{GRUPPO}$.

4.4 Immagini

Tutte le immagini che vengono inserite all’interno del documento dovranno avere una didascalia significativa e singolare. Ogni immagine dovrà avere un numero progressivo che la identifichi così da poter essere più facilmente rintracciabile e dovrà essere inserita nell’indice delle immagini del documento. Tutte le immagini presenti nei file dovranno essere memorizzate in formato *png_G*; è stato scelto questo formato poiché è compatibile con tutte le applicazioni usate dal gruppo, crea file di dimensioni ridotte ed è un formato *open source_G* con licenza *GPL_G*.

4.5 Tabelle

Anche per le tabelle, al fine di renderle facilmente rintracciabili e comprensibili, dovranno essere identificate da un numero progressivo ed una didascalia singolare cosicché possano essere inserite nell’apposito indice. Per quanto riguarda la loro struttura grafica invece dovranno avere:

- **header colonne:** testo nero in grassetto;
- **celle generiche:** testo nero;
- **colore di sfondo delle celle:** bianco.

4.6 Elenchi

Gli elenchi potranno essere sia di tipo numerico che di tipo puntato. I numerici verranno usati quando l’ordine degli elementi è rilevante, mentre gli altri verranno usati quando c’è il bisogno di scrivere una semplice lista di elementi in cui l’ordine di apparizione è superfluo. Gli elenchi numerici potranno iniziare sia da 1, formattazione standard del

LATEX_G , oppure da 0. La scelta sta al redattore del documento in base al contesto in cui si trova. Ogni punto di un elenco deve terminare con il punto e virgola, tranne l'ultimo che deve concludersi con il punto.

4.7 Ciclo di vita del documento

Durante la realizzazione ogni documento si troverà ad essere in una delle seguenti 3 fasi anche più di una volta, fino ad arrivare alla sua versione finale. Le 3 fasi in cui si potrà trovare sono:

- **in lavorazione:** un documento entra in questa fase nel momento della sua creazione e ne continua la sua permanenza fino alla conclusione della sua realizzazione, oppure vi rientra nel caso siano necessarie successive modifiche.

Una volta concluso il documento rimarrà in questa fase fino al momento in cui i *Verificatori* non confermeranno che il documento è privo di errori e non contiene delle imprecisioni, sia semantiche che sintattiche;

- **verificato:** il documento entra in questa fase una volta accertato dai *Verificatori* che non sono presenti errori o incongruenze. In questa fase ci rimane fino al momento in cui non riceve l'approvazione da parte del *Responsabile*.

Se invece vengono riscontrati degli errori allora il documento ritornerà nella fase "In lavorazione";

- **approvato:** se la fase di verifica ha dato esito positivo il documento dovrà essere approvato dal *Responsabile di Progetto*. L'approvazione ne decreta lo stato finale per la data versione.

Una volta approvato, se vi fosse la necessità di una revisione formale il documento rientrerà nella fase "In lavorazione" e dovrà affrontare nuovamente tutto l'iter che lo porterà ad una nuova approvazione e conseguentemente ad un avanzamento di versione. La fase di lavorazione del documento sarà annotata nel diario del documento.

4.8 Versionamento

Ogni documento nel momento in cui sarà distribuito dovrà essere corredato dal numero di versione. La codifica scelta per la versione è:

$$verA.B.C$$

con il significato:

- **A:** rappresenta il numero di uscite formali del documento;
- **B:** rappresenta il numero crescente di modifiche sostanziali;
- **C:** rappresenta il numero crescente di modifiche minori effettuate;

La versione di ogni documento è visibile nel frontespizio e dovrà corrispondere con quella riportata nel nome del documento, per quanto riguarda la sua evoluzione possiamo controllare tutti i passaggi nel "Diario delle modifiche", presente in ogni documento che verrà aggiornata ad ogni modifica.

4.9 Norme sui nomi dei file

Per rendere più agevole l'accesso ai file i membri sono tenuti ad utilizzare la seguente nomenclatura per tutti i file caricati nel *repository_G*:

NomeDelFile__verA.B.C.ext

Dove "NomeDelFile" indica il nome del file che deve iniziare con lettera maiuscola, "ext" indica l'estensione del file. Nel caso il nome sia composto da più parole verrà seguita la convenzione *CamelCase_G*, la prima parola può iniziare sia con maiuscola che con la minuscola mentre la prima lettera di ogni parola successiva dovrà essere maiuscola. Per quanto riguarda la versione del file si rimanda alla sezione 4.8. Non vi è la possibilità di introdurre separatori nei nomi dei file o di utilizzare nomenclature che non rispettino queste regole. Inoltre non è permesso l'utilizzo di nomi contenenti lettere accentate.

Per uniformare e velocizzare la stesura dei file sono state create delle apposite macro per la citazione dei vari documenti:

- \NdP: Norme di Progetto;
- \PdP: Piano di Progetto;
- \Pdq: Piano di Qualifica;
- \Glo: Glossario;
- \SdF: Studio di Fattibilità;
- \AR: Analisi Requisiti.

4.10 Classificazione dei documenti

Un documento realizzato sarà classificato come formale o informale, condizione dettata dallo stadio di avanzamento in cui si trova:

4.10.1 Documento informale

Un documento informale è da considerarsi esclusivamente come un documento interno al gruppo, il documento sarà così classificato da quando viene creato fino al momento in cui riceve l'approvazione dal *Responsabile*. Questo tipo di documento sarà archiviato all'interno di un *repository_G* a cui hanno accesso solamente i membri del team.

4.10.2 Documento formale

Una volta ricevuta l'approvazione dal *Responsabile*, il documento potrà essere classificato come formale e conseguentemente distribuito all'esterno del gruppo a chi ne facesse richiesta. Il passaggio da informale a formale coinciderà con le revisioni presso il Committente. Ogni documento per poter essere considerato come formale dovrà aver seguito tutto l'iter descritto in questo documento nella sezione 4.7.

5 Processi Organizzativi

5.1 Rotazione dei ruoli

Durante la realizzazione del progetto i membri del gruppo andranno a ricoprire i ruoli solitamente presenti all'interno di un'azienda. Ciascun componente del gruppo dovrà ricoprire almeno una volta ogni ruolo. Per garantire una coerente rotazione dei ruoli, con lo scopo di non creare conflitti è necessario che le attività di stesura e verifica vengano pianificate dettagliatamente e che i soggetti interessati rispettino i compiti a loro assegnati. Sarà poi compito del *Verificatore* controllare attentamente il diario delle modifiche di ogni documento per individuare eventuali incongruenze. Si descrivono ora i diversi ruoli di progetto, con le relative responsabilità e le modalità operative affinché essi possano svolgere i compiti assegnati con l'ausilio dei software scelti per il progetto.

5.1.1 Responsabile

È colui che rappresenta il progetto presso il Committente ed il Fornitore. Il *Responsabile* ha responsabilità su:

- pianificazione, coordinamento e controllo delle attività;
- gestione e controllo delle risorse;
- analisi e gestione dei rischi;
- approvazione dei documenti;
- approvazione dell'offerta economica.

È suo compito assicurarsi che le attività di verifica vengano svolte in maniera sistematica secondo quanto riportato nelle *NormeDiProgetto_ver1.4.0*, vengano rispettati i ruoli e le competenze assegnate nel *PianoDiProgetto_ver1.4.0* e non vi siano conflitti di interesse tra redattori e verificatori. Spetta al *Responsabile* decidere l'approvazione di un documento e a sancirne la distribuzione. Deve anche gestire la creazione e l'assegnazione dei ticket delle macro-fasi e di assegnare ad un membro del gruppo il ruolo di responsabile di quest'ultima. Redige il *PianoDiProgetto* e collabora alla stesura del *PianoDiQualifica*.

5.1.2 Amministratore

Il compito dell'*Amministratore* è quello di equipaggiare, organizzare e gestire l'ambiente di lavoro e produzione. Per fare questo deve:

- ricercare gli strumenti per automatizzare i compiti dell'essere umano, ove possibile;
- effettuare il controllo delle versioni e configurazione dei prodotti;
- fornire strumenti e procedure per monitorare e segnalare il controllo di qualità;
- occuparsi del versionamento della documentazione realizzata.

Dovrà scrivere le *NormeDiProgetto*, gli strumenti e i metodi per la verifica nel *PianoDiQualifica*.

5.1.3 Analista

Si deve occupare delle attività di analisi. I compiti sono:

- conoscere il dominio e la complessità del problema;
- realizzare una specifica del progetto che sia motivata in ogni aspetto ma sia soprattutto di facile comprensione per il Proponente, il Committente ed il *Progettista*.

Si occuperà della realizzazione dello *StudioDiFattibilità*, dell'*AnalisiDeiRequisiti* e anche del *PianoDiQualifica*.

5.1.4 Progettista

Il *Progettista* è addetto alla fase di progettazione e ha come responsabilità quella di

- produrre una soluzione comprensibile ma soprattutto che sia attuabile;
- effettuare scelte, sia tecnologiche che progettuali, al fine di rendere il prodotto di facile manutenzione.

Dovrà anche redigere la *SpecificaTecnica*, *DefinizioneDiProdotto* e scegliere le metriche per la verifica della programmazione che sono riportate nel *PianoDiQualifica_ver1.4.0*.

5.1.5 Programmatore

Si deve occupare delle attività di codifica e delle componenti, *stub_G*, *driver_G*, *logger_G* per effettuare la verifica e la validazione. Per facilitare la fase di verifica deve:

- attenersi alle soluzioni decise e descritte dal *Progettista*;
- rendere il codice il più mantenibile possibile attraverso una buona documentazione e rispettando le scelte fatte per la codifica;
- implementare i test per la verifica e validazione.

Avrà anche il compito di realizzare il *ManualeUtente*.

5.1.6 Verificatore

Il compito del *Verificatore* è quello di:

- accertarsi che i documenti in uscita siano corretti e validi;
- assicurare che l'attuazione delle attività sia conforme a quanto stabilito;
- controllare la conformità di ogni stadio del *ciclo di vita_G*.

Per svolgere la verifica dei vari documenti deve utilizzare gli strumenti e i metodi descritti nei documenti di *NormeDiProgetto* e *PianoDiQualifica*. Terminata la verifica dovrà, nel documento *PianoDiQualifica*, descriverne l'esito e la completezza delle verifiche e prove fatte.

5.2 Strumenti e Ambiente di lavoro

Per coordinarsi nella realizzazione del prodotto e non incorrere in problematiche legate alla diversità dell'ambiente di lavoro il team ha scelto di adottare il medesimo ambiente. Di seguito sono elencati i software che verranno utilizzati, i software riguardanti la verifica sono stati inseriti nella sezione 6.5.

5.2.1 Sistema operativo

Tutti i membri del gruppo operano su sistema operativo Linux, specificamente nella distribuzione *Kubuntu_G* x64 con release 12.04 o successive.

5.2.2 Coordinamento e gestione del progetto

Per la gestione del progetto è stato scelto di creare un account sul sito [Hosted Redmine](#), creare il progetto ed aggiungere ogni componente del team come membro così da poter utilizzare il sistema di *ticketing_G*. La piattaforma fornisce:

- il diagramma di *Gantt_G* delle attività;
- l'assegnazione dei ticket;
- un calendario per la visualizzazione delle scadenze assegnate;
- un sistema per il rendiconto del tempo speso.

5.2.2.1 Creazione di un ticket

Per la creazione di un ticket il *Responsabile* deve seguire il seguente iter:

1. autenticarsi al sito [Hosted Redmine](#);
2. selezionare la scheda “*Nuova segnalazione*” nel menù in alto;
3. premere il bottone “*Privato*” in alto a destra;
4. compilare completamente l'apposita form, possono essere lasciati vuoti gli ultimi 4 campi: Checklist, Screenshot, File e Osservatore;
5. assicurarsi di aver salvato il ticket premendo il tasto “*Crea*”.

Il software provvederà automaticamente ad informare l'assegnatario del ticket, inserire il ticket nel calendario e una volta terminato a creare il *Gantt_G*. In questo modo il lavoro che dovrà esser svolto dal *Responsabile* risulterà alleggerito, ottimizzando anche le comunicazione tra i membri.

5.2.3 Software per il versionamento

Per il versionamento il team ha deciso di usare *Git_G*. La scelta è ricaduta su questo software perché era già conosciuto da tutti i componenti del team. In concomitanza alla scelta di utilizzare *Git_G* è stato deciso di appoggiarsi a *GitHub_G* per mantenere lo storico di tutte le modifiche apportate ai file e ai documenti relativi al progetto. La scelta è dovuta anche alla richiesta manifestata dal Proponente nel capitolato. *GitHub_G* sarà sfruttato dal team per tenere il *repository_G* del progetto.

Per maggiori informazione riguardo la gestione del *repository_G* vedere la sezione 5.3

5.2.4 Software per la pianificazione

Per organizzare la pianificazione del lavoro e delle risorse è stato scelto di utilizzare *ProjectLibre_G*. La scelta di utilizzare questo software è principalmente dovuta al sistema operativo su cui si è scelto di lavorare. Un'altra valida alternativa sarebbe stata *Microsoft Project_G*, che però è utilizzabile solamente su piattaforma Microsoft Windows. I diagrammi prodotti con *ProjectLibre* sono compatibili con le versioni 2010 e successive di *Microsoft Project_G*. Il software permette di:

- realizzare i diagrammi di *Gantt_G*;
- assegnare le risorse alle attività;
- creare automaticamente sia i diagrammi di *PERT_G* sia la *Work Breakdown Structure (WBS)*;
- calcolare la *Schedule Variance (SV)_G* e il *Budget Variance (BV)_G*.

5.2.5 Software per il tracciamento

Per realizzare il tracciamento dei requisiti, dei *casì d'uso_G* e delle fonti il team ha realizzato un proprio software, denominato *FIRE (Flametech Internal Requirements Engine)*. Il software appoggia su un *database_G MySQL_G* ed è reperibile all'indirizzo <http://flametech.web44.net/>. Per automatizzare la stesura del documento *Analisi dei Requisiti* è stata realizzata una apposita funzione che esporta il contenuto in codice $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}_{\text{G}}$.

5.2.6 Software creazione diagrammi UML

Il software utilizzato per la creazione dei diagrammi *UML_G* nella versione 2.0 è *Astah_G Community Edition*. Con l'utilizzo del software verranno realizzati i diagrammi dei *casì d'uso_G*, delle classi, dei *package_G*, di sequenza e di attività.

5.2.7 Software creazione delle presentazioni

Per la creazione delle presentazioni verrà usato *Prezi_G*. La scelta è ricaduta su questo software perché è gratuito, è online ed è possibile lavorarci in più persone contemporaneamente.

5.2.8 Altri software utilizzati

Il gruppo si servirà di altri software non strettamente collegati al progetto ma utili per varie comunicazioni interne tra i membri.

- **Facebook_G**: è stato creato un gruppo di lavoro privato interno al *social network_G* per gestire le comunicazioni interne;
- **Telegram_G**: è stato realizzato un gruppo che viene utilizzato dal team per le comunicazioni rapide;
- **Hangouts_G**: piattaforma utilizzata per effettuare chiamate o videochiamate nel caso non fosse possibile indire una riunione fisica tra i componenti.

5.3 Repository

5.3.1 Struttura del *repository_G*

I membri del gruppo sono indotti a rispettare l'organizzazione dei file all'interno di tale struttura per permettere una facile ricerca agli altri membri e per mantenere un'ottima organizzazione dei documenti. In particolare la *root_G* conterrà solamente:

- **Documentazione:** una cartella contenente la documentazione di progetto a sua volta suddivisa in:
 - **interna:** contenente la documentazione interna al gruppo e che non sarà distribuita esternamente;
 - **esterna:** contenente i documenti realizzati e che saranno poi distribuiti esternamente al gruppo;
 - **template:** contenente il *template_G* utilizzato per la realizzazione dei documenti.
- **Codice:** in questa sezione troviamo i file contenenti il software prodotto.

5.3.2 Norme sui *commit_G*

Ogni qual volta si termina una modifica ad un file, va eseguito l'update nel *repository_G* tramite il comando *commit_G*. Il commit verrà effettuato tramite il seguente comando:

```
git commit -m "motivo" nomefile
```

Il motivo del *commit_G* deve essere strutturato in due parti. La motivazione viene inserita fra parentesi quadre per facilitare la comprensione dell'ambito della modifica; a riguardo si propongono tre categorie:

- **[STE]** rappresenta la stesura del documento o di una sua parte;
- **[MOD]** per indicare un aggiornamento per ampliare il documento;
- **[REV]** sarà utilizzato per indicare che sono state effettuate modifiche in seguito a correzioni del documento.

In seguito si dovrà trovare una breve ma chiara descrizione delle operazioni effettuate. L'operazione di *commit_G* deve essere eseguita solo al termine delle operazioni di modifica o quando ci sono modifiche sostanziali ai file.

5.4 Codifica e Convenzioni

Tutti i file realizzati dovranno essere in codifica *UTF-8_G* senza l'utilizzo del *BOM_G*. Per andare a capo invece verrà usato *LF_G*(U+000A). Agli sviluppatori è richiesto di attenersi alle *Javascript Coding Conventions_G*.

Qualora qualche sviluppatore volesse modificare le convenzioni sopra descritte deve inviarne formale richiesta al *Responsabile*, il quale dovrà valutarne la proposta e in caso la richiesta fosse accettata deve comunicarla a tutti gli altri sviluppatori per uniformarsi.

5.4.1 Nomi

Per i nomi delle variabili, nomi delle classi e delle funzioni dovrà essere usata esclusivamente la lingua inglese e si dovrà utilizzare la notazione *CamelCaseG*. Le classi dovranno avere la prima lettera maiuscola, mentre le funzioni e le variabili dovranno averla minuscola.

5.4.2 File contenente del codice

Tutti i file che al loro interno conterranno del codice dovranno contenere la seguente intestazione:

```
/*
 * \date Data di creazione
 * \file Nome del file
 * \author Autore
 *
 *Descrizione del file
 */
```

Prima di ogni funzione/metodo dovrà essere inserito un commento come il seguente:

```
/*
 * \file Nome della funzione
 * \author Autore
 * \parameters param1, param2
 *
 * Descrizione dello scopo della funzione
 */
```

Prima di ogni classe dovrà essere inserito un commento come il seguente:

```
/*
 * \date Data di creazione
 * \classe Nome del classe
 * \author Autore
 *
 * Descrizione della classe
 */
```

5.4.3 Ricorsione

L'utilizzo della ricorsione in fase di programmazione deve essere, se possibile, totalmente evitata. Nel caso venisse utilizzata è obbligatorio fornire dei test che ne dimostrino l'effettiva terminazione e l'impatto che ha nell'occupazione della memoria. Al termine di questi test verrà valutato se può essere mantenuta o se deve essere trasformata nella corrispettiva versione iterativa.

6 Procedure di supporto

6.1 Gestione del progetto

6.1.1 Pianificazione attività

La gestione dell'intero progetto è compito del *Responsabile*, che dovrà garantire che le attività vengano svolte in maniera corretta. Per farlo può utilizzare strumenti che gli permettano di:

- gestire e controllare le risorse;
- analizzare e gestire i rischi;
- pianificare e controllare le attività.

6.1.2 Pianificazione e controllo delle attività

Per eseguire e garantire una buona pianificazione, il *Responsabile* dovrà realizzare, tramite il software *ProjectLibre_G*, il diagramma di *Gantt_G* per tutte le attività presenti nel

PdP. Fatta la pianificazione dovrà assegnare ogni attività ad un componente del gruppo e accertarsi che stiano avanzando in maniera corretta.

6.1.3 Controllo e gestione delle risorse

Oltre al controllo delle attività il *Responsabile* dovrà monitorare le varie risorse e verificare l'avanzamento di ogni processo, come descritto nel *PianoDiProgetto_ver1.4.0*.

6.1.4 Controllo e gestione dei rischi

Per tutta la durata del progetto il *Responsabile* dovrà monitorare i possibili rischi, sia quelli già descritti nel *PianoDiProgetto_ver1.4.0* sia quelli nuovi che potrebbero emergere. Per mitigare gli effetti collaterali dovrà attuare gli accorgimenti previsti nel *PianoDiProgetto_ver1.4.0*.

6.2 Analisi dei rischi e Studio di Fattibilità

In seguito alla presentazione dei capitolati, il *Responsabile di Progetto* deve convocare delle riunioni che serviranno al confronto tra i membri del team per confrontarsi su quale capitolato svolgere e sulla possibilità di realizzazione entro i termini previsti. Da queste riunioni gli *Analisti* usciranno con la conoscenza di quale sia la preferenza e la preparazioni, in merito ad ogni capitolato, dei rispettivi membri del gruppo. Da questo momento è compito degli *Analisti* realizzare lo *StudioDiFattibilità* tenendo conto dei vari aspetti:

- **Rapporto Costi/Benefici:** gli *Analisti* dovranno conoscere quali sono i competitor ed i prodotti simili, il costo di realizzazione del progetto e quali devono essere i requisiti obbligatori che dovranno essere soddisfatti;
- **Dominio Applicativo e Tecnologico:** conoscenza di quali saranno le tecnologie da applicare, problematiche emerse dal capitolato e conoscenza del dominio applicativo;

- **Rischi:** conoscenza di quali potrebbero essere i punti critici durante la realizzazione, difficoltà nell'analisi, individuazione dei requisiti e come questi dovranno poi essere verificati. Gli *Analisti* dovranno inoltre conoscere le lacune di ogni membro del gruppo per quanto riguarda l'ambito tecnico e applicativo.

Al termine della realizzazione dello *StudioDiFattibilità_ver1.4.0* sarà indetta una riunione per presentare la decisione finale sul capitolato.

6.3 Analisi dei Requisiti

Oltre alla stesura dello *StudioDiFattibilità*, agli *Analisti* spetta il compito di realizzare l'*AnalisiRequisiti*. La stesura del documento dovrà rispettare le fasi che verranno ora descritte.

6.3.1 Classificazione dei requisiti

Ogni requisito, individuati tramite un incontro con il Proponente o l'analisi del capitolato, dovrà essere identificato rispettando la seguente codifica:

R[importanza][tipo][codice identificativo]

- **Importanza:** può assumere solo uno dei seguenti valori:
 - 0. requisito obbligatorio;
 - 1. requisito desiderabile;
 - 2. requisito opzionale.
- **Tipo:**
 - F: Requisito Funzionale;
 - Q: Requisito Qualitativo;
 - P: Requisito Prestazionale;
 - V: Requisito di Vincolo.
- **Codice identificativo:**
 - è il codice univoco di ogni requisito e sarà espresso da un numero seguendo l'ordine gerarchico (esempio: 1.1.5).

Ogni requisito, oltre ad essere codificato, avrà una breve descrizione e la fonte da cui è stato estrapolato. Le uniche **fonti** possibili sono:

- **capitolato:** il requisito è estrapolato direttamente dal capitolato;
- **caso d'uso_G:** questo requisito non è emerso direttamente dal capitolato ma durante la realizzazione dei *casi d'uso_G*;
- **padre:** nasce dalla necessità di suddividere un requisito di livello superiore;
- **interno:** gli *Analisti* hanno ritenuto importante l'aggiunta di questo requisito;
- **verbale:** il requisito è nato tramite l'incontro con il Committente o durante una riunione del gruppo.

6.3.2 *Casi d'uso_G*

Concluso lo *Studio di Fattibilità* spetta sempre agli *Analisti* l'analisi dei *casi d'uso_G*. Anche i *casi d'uso_G* dovranno essere classificati secondo una codifica precisa. La codifica scelta è

UC[codice identificativo]

- **codice identificativo:** codice identificativo gerarchico di ogni caso d'uso. Il codice univoco sarà così strutturato :

[codice univoco del padre.codice univoco di livello].

Nel caso di derivazioni superiori a due livelli il codice corrisponderà alla sequenza dei codici di derivazione di tutti i livelli.

6.3.3 Tracciamento dei Requisiti e dei *Casi d'Uso_G*

Per automatizzare il tracciamento dei requisiti e dei *casi d'uso_G* il team ha deciso di realizzare un applicativo, per uso interno, che si appoggia su un db. In questa maniera ad ogni modifica dei requisiti e/o dei *casi d'uso_G* si avrà un aggiornamento automatico della corrispondenza con le fonti. L'applicativo, oltre al tracciamento dei requisiti e dei *casi d'uso_G*, è stato dotato di una apposita funzione che esporta direttamente in codice \LaTeX_G il contenuto del *database_G*, in maniera da velocizzare la realizzazione del documento dell'*AnalisiRequisiti_ver1.4.0*. Nonostante questo sia automatizzato spetta agli *Analisti* controllare la correttezza e la corrispondenza biunivoca di un requisito con una o più fonti. Nel caso dovessero essere riscontrate delle anomalie nel funzionamento bisogna informare il *Responsabile*, che dovrà prendere le decisioni su come procedere per la correzione del problema.

6.4 Progettazione

6.4.1 Diagrammi *UML_G*

Per la realizzazione dei diagrammi *UML_G* è stato scelto di utilizzare il software *Astah_G*, nella versione Community, e di adottare la versione *UML_G 2.0*. Nella fase di progettazione il gruppo farà uso dei seguenti diagrammi:

- Diagrammi dei *Casi d'Uso_G*;
- Diagrammi delle classi;
- Diagrammi di sequenza;
- Diagrammi delle classi;
- Diagrammi dei package_G.

6.4.2 *Design Pattern_G*

La scelta dei *Design Pattern_G* da utilizzare per l'architettura del prodotto spetta ai *Progettisti*. Per ogni *design pattern_G* adottato dovranno fornire una breve descrizione ed un diagramma della struttura.

6.5 Verifica

La fase di verifica deve essere attuata fin dal principio e non relegata solamente alla fine, così facendo è possibile ottenere un risultato migliore. In questa sezione verranno descritte le metodologie che i *Verificatori* dovranno seguire per effettuare le analisi statiche e dinamiche dei processi, dei documenti e del codice.

6.5.1 Strumenti per la verifica

Durante il processo di verifica verranno usati i seguenti strumenti software:

- **Correttore TeXstudio:** l'ambiente per la stesura dei documenti in $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}_G$, consigliato nelle *NormeDiProgetto_ver1.4.0*, permette l'integrazione dei dizionari di *Apache OpenOffice_G* che segnalano possibili errori ortografici;
- **FIRE_G:** è un software di tracciamento di requisiti scritto appositamente dal gruppo **FlameTech Inc.** ;
- **Aspell_G:** è un software libero per il controllo ortografico;
- **JSHint_G:** è uno strumento per il rilevamento di errori e potenziali problemi nel codice *JavaScript_G*, e per imporre convenzioni di codifica;
- **Jasmine_G:** è un *framework_G* di testing *open source_G* per *JavaScript_G*. Non dipende da altri *framework_G* *JavaScript_G* e non richiede un *DOM_G*.

6.5.2 Verifica dei processi

Il protocollo per l'analisi dei processi dovrà eseguire tutti i passi di seguito descritti nell'ordine di apparizione:

1. Controllo valore indici

Terminata ogni macro attività, come definita nel *Piano di Progetto*, dovranno essere verificati i valori degli indici BV_G ed SV_G . Per una buona misurazione sarà necessario valutarne la media;

2. Analisi grafico $PDCA_G$

Si dovrà anche andare a valutare il grafico del $PDCA_G$ e da qui si potranno trarre delle conclusioni sullo svolgimento dei processi. Dall'analisi di questo grafico il *Responsabile* potrà capire se sono stati fatti errori durante la fase di pianificazione del lavoro e la velocità con cui procede il gruppo nel portare avanti i processi.

6.5.3 Verifica dei documenti

Per un controllo efficiente ed accurato dei documenti dovranno essere eseguiti tutti i passi nel seguente ordine:

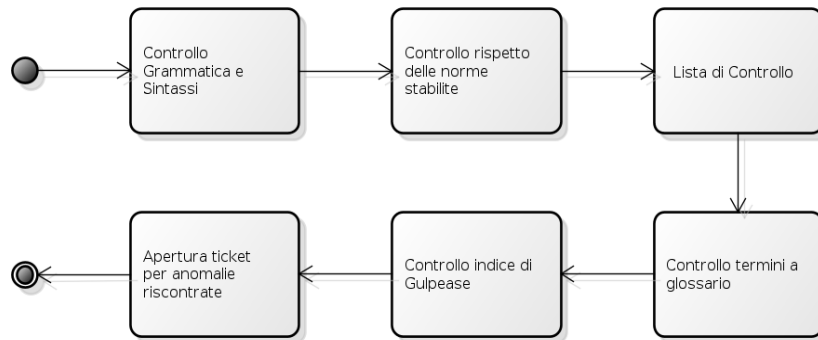


Figura 1: Procedura per l'analisi dei documenti

1. Controllo grammatica e sintassi

Questo controllo viene eseguito in parte in maniera automatica, con l'ausilio di software, ed in parte deve essere fatto manualmente dal verificatore. Per la correzione degli errori grammaticali e di sintassi è stato usato il correttore di *TeXstudio*, dopo avergli importato il dizionario italiano. Per quanto riguarda invece la punteggiatura e la comprensione delle frasi complesse deve intervenire manualmente un *Verificatore*;

2. Norme di Progetto

La verifica delle norme dovrà essere svolta interamente manualmente da parte del *Verificatore*. Egli dovrà controllare che quanto scritto nei vari documenti corrisponda a ciò che è stato stabilito nel documento *NormeDiProgetto_ver1.4.0*;

3. Lista controllo

Il *Verificatore* dovrà utilizzare la lista di controllo per i documenti al fine di verificare che gli errori tipici non siano presenti;

4. Termini a glossario

Il controllo della corrispondenza tra i termini segnalati come presenti a glossario e la presenza effettiva, almeno nella fase iniziale, dovrà essere svolta manualmente;

5. Indice di *Gulpease_G*

Per ogni documento dovrà essere verificato che l'indice di leggibilità del documento rientri nel range di accettazione, in caso negativo si dovrà provvedere alla modifica ed ad una successiva verifica;

6. Segnalazione anomalie

Il *Verificatore* terminata l'analisi del documento dovrà segnalare le anomalie riscontrate attraverso l'attivazione dei ticket, come descritto nella sezione 5.2.2.1 *Creazione di un ticket*.

6.5.4 Verifica tracciamento dei requisiti

Il *Verificatore* deve accertarsi della corretta realizzazione dei *casi d'uso_G* prodotti ed il corretto tracciamento con i requisiti.

6.5.5 Verifica UML

Il verificatore deve accertarsi che i diagrammi prodotti siano conformi alle specifiche *UML_G 2.0*.

- **Diagrammi dei casi d'uso_G:** questo tipo di controllo deve essere effettuato in maniera manuale dal *Verificatore*. Un punto importante su cui prestare attenzione è l'uso delle relazioni di inclusione e estensione. Inoltre deve accertarsi che quanto descritto nel caso d'uso sia effettivamente rappresentato nel diagramma.

7 Glossario

Per non creare ambiguità a chi leggerà i documenti è stato redatto un glossario nel quale andranno inserite tutte le definizioni che potrebbero essere fraintese dal lettore.

7.1 Inserimento del termine

L'inserimento del termine andrà fatto parallelamente alla stesura del documento. Qualora non fosse possibile per qualche motivo inserire immediatamente la definizione, il redattore dovrà inserire il termine lasciando la descrizione vuota, così facendo si ridurranno i possibili errori umani, ad esempio la possibilità che un termine segnato, in qualche documento, come presente a glossario sia poi tralasciato.

7.2 Ordine di apparizione

I termini contenuti nel glossario seguiranno l'ordine alfabetico, sta quindi a chi andrà ad inserirlo accertarsi di rispettare l'ordine.