

---

# PROGETTO SIGEOL

*Norme di Progetto*

*v2.0.0*

Redazione: Scarpa Davide

9 giugno 2009



*quixoft.sol@gmail.com*

<b>Verifica:</b>	Grosselle Alessandro
<b>Approvazione:</b>	Scarpa Davide
<b>Stato:</b>	Formale
<b>Uso:</b>	Interno
<b>Distribuzione:</b>	QuiXoft

## Sommario

Norme interne per il progetto *SIGEOL*, necessarie per regolamentare lo svolgersi delle operazioni di sviluppo.

---



## Indice

<b>1</b>	<b>Introduzione</b>	<b>1</b>
1.1	Scopo del documento . . . . .	1
1.2	Definizioni . . . . .	1
<b>2</b>	<b>Comunicazioni</b>	<b>1</b>
2.1	Interne . . . . .	1
2.2	Esterne . . . . .	2
<b>3</b>	<b>Convocazioni ed incontri</b>	<b>2</b>
3.1	Interni . . . . .	2
3.1.1	Obbligatorie . . . . .	2
3.1.2	Facoltative . . . . .	2
3.2	Esterne . . . . .	3
3.2.1	Incontri con il committente . . . . .	3
3.2.2	Altri incontri esterni . . . . .	3
<b>4</b>	<b>Repository</b>	<b>3</b>
4.1	Versione . . . . .	3
<b>5</b>	<b>Documentazione</b>	<b>4</b>
5.1	Template documenti . . . . .	4
5.1.1	Documenti di resoconto incontri . . . . .	4
5.1.2	Documenti ufficiali interni ed esterni . . . . .	4
5.2	Norme tipografiche . . . . .	4
5.3	Nominazione dei file di documentazione . . . . .	5
5.3.1	Resoconto incontri interni ed esterni . . . . .	5
5.3.2	Documenti ufficiali interni ed esterni . . . . .	6
5.4	Approvazione e verifica documentazione . . . . .	6
<b>6</b>	<b>Presentazioni</b>	<b>6</b>
6.1	Template presentazioni . . . . .	7
<b>7</b>	<b>Codice</b>	<b>7</b>
7.1	Intestazione . . . . .	7
7.2	Codice Ruby . . . . .	8
7.2.1	Convenzioni sui nomi . . . . .	8
7.2.2	Altre convenzioni . . . . .	9
7.3	Codice Java . . . . .	10
7.3.1	Convenzioni sui nomi . . . . .	10
7.3.2	Altre convenzioni . . . . .	10



## INDICE

---

<b>8</b>	<b>Compiti</b>	<b>10</b>
8.1	Ticket . . . . .	10
8.2	Milestone . . . . .	11
<b>9</b>	<b>Strumenti</b>	<b>11</b>



# 1 Introduzione

## 1.1 Scopo del documento

Lo scopo del presente documento è quello di fissare linee guida e norme precise per quanto riguarda la comunicazione aziendale, la stesura della documentazione, la codifica su calcolatore e l'amministrazione del progetto denominato SIGEOL.

Ogni membro del team QuiXoft è tenuto a leggere con attenzione il presente documento e applicarne in modo più coerente possibile il contenuto, al fine di ottimizzare il lavoro in team e le risorse a disposizione.

Sarà compito del Responsabile verificare che tali norme vengano applicate ed, eventualmente, notificare all'Amministrazione di progetto qualsiasi insubordinazione che mini il corretto procedere dello stesso.

## 1.2 Definizioni

Le definizioni dei termini specialistici usati nella stesura di questo e di tutti gli altri documenti possono essere trovate nel documento GLOSSARIO al fine di eliminare ogni ambiguità e di facilitare la comprensione dei temi trattati. Ogni termine la cui definizione è disponibile all'interno del glossario verrà marcato con una sottolineatura.

# 2 Comunicazioni

## 2.1 Interne

Le comunicazioni interne ufficiali (quali incontri, scadenze, appuntamenti obbligatori o altre comunicazioni ritenute importanti) si appoggiano al portale <http://www.assembla.com>, che mette a disposizione del team una bacheca messaggi.

Ogni nuova comunicazione pubblicata nella bacheca verrà notificata automaticamente via e-mail a tutti membri del team QuiXoft, qualsiasi ruolo essi ricoprano; la pubblicazione della stessa, invece, spetta al Responsabile.

Per tutte le altre comunicazioni interne al team è inoltre a disposizione un forum, raggiungibile all'indirizzo <http://quixoft.forumup.it/>, la cui struttura è decisa preventivamente dall'Amministratore di progetto.

All'interno di tale forum ogni membro del team ha la possibilità di discutere, liberamente, argomenti annessi al progetto, o che possano in qualsiasi modo giovare al proseguimento dello stesso.

Sono sconsigliate, ma non vietate, le comunicazioni dirette tra singoli membri del team tramite qualsivoglia metodo: ogni comunicazione inerente al progetto potrebbe infatti tornare utile a qualche altro membro del gruppo, oppure tornare utile in futuro. E' consigliabile quindi che ogni comunicazione sia fatta usando i metodi sopra elencati. Nonostante ciò,



### 3 CONVOCAZIONI ED INCONTRI

---

sono a disposizione gli indirizzi e-mail e i recapiti telefonici di ogni singolo membro del team QuiXoft.

#### 2.2 Esterne

Ogni comunicazione con il committente o con altri soggetti esterni al team QuiXoft potrà essere proposta da qualsiasi membro del gruppo, e spetterà al Responsabile deciderne l'utilità e la fattibilità. In caso di responso positivo, al termine di ogni comunicazione esterna dovrà essere redatto un resoconto che ne elenchi sinteticamente i risultati. Per le modalità di creazione e diffusione di tale resoconto, si faccia riferimento a quanto spiegato in seguito per i resoconti di incontro con il committente.

Il committente o altri soggetti esterni potranno invece comunicare con il team QuiXoft tramite l'indirizzo e-mail *quixoft.sol@gmail.com*.

### 3 Convocazioni ed incontri

Ogni incontro, sia esso interno o esterno, dovrà essere preceduto da un annuncio in bacheca e dall'approvazione del Responsabile. Al termine di qualsiasi incontro dovrà essere redatto un resoconto che contenga sinteticamente i risultati o i problemi emersi durante la discussione. Per ulteriori informazioni sui documenti di resoconto di un incontro consultare il paragrafo 6.1.1.

#### 3.1 Interni

##### 3.1.1 Obbligatorî

La cadenza degli incontri interni obbligatori è fissata dal Responsabile e comunicata ai membri del team QuiXoft tramite segnalazione con messaggio in bacheca. Notifica via e-mail della convocazione verrà automaticamente mandata a tutti i membri del gruppo. Riunioni di questo genere implicano la presenza obbligatoria di tutti i componenti del gruppo.

Il Responsabile si occuperà di volta in volta di selezionare l'ordine del giorno appropriato e di controllare il corretto svolgimento dell'incontro. Gli argomenti da discutere e da inserire nell'ordine del giorno potranno essere proposti da qualsiasi membro del gruppo e saranno inseriti nell'annuncio di convocazione in bacheca, previa approvazione del Responsabile.

##### 3.1.2 Facoltativi

La cadenza degli incontri facoltativi non è fissata a priori ed è legata a particolari necessità di uno o più membri del team. Gli interessati dovranno richiedere la possibilità di svolgere tale incontro al Responsabile, il quale provvederà all'eventuale approvazione. Sarà poi compito del Responsabile



## 4 REPOSITORY

---

stesso comunicare, tramite pubblicazione sulla bacheca, gli estremi di svolgimento dell'incontro.

Riunioni di questo tipo non implicano la presenza obbligatoria di tutti i componenti, ma solamente dei ruoli interessati e specificati nel messaggio di convocazione in bacheca; non si esclude comunque la presenza degli altri elementi.

### 3.2 Esterni

#### 3.2.1 Incontri con il committente

Ogni membro del gruppo ha facoltà di richiedere al Responsabile un incontro con il Committente. Spetta al Responsabile valutare la richiesta e, se lo ritiene opportuno e utile al progetto, approvare l'incontro.

Per formalizzare l'incontro dovrà essere pubblicato in bacheca il messaggio di convocazione, la cui notifica arriverà a tutti i membri del team. All'incontro è obbligatoria solamente la presenza del richiedente, ma non si esclude la presenza di qualsiasi altro membro del gruppo che reputi di interesse tale incontro con il Committente.

#### 3.2.2 Altri incontri esterni

Oltre che con il committente, è data la possibilità di organizzare incontri con altri soggetti esterni al team QuiXoft, per qualsiasi argomento che possa aiutare il proseguimento del progetto. La procedura di richiesta di tale tipo di incontro è analoga a quella sopra citata per gli Incontri con il Committente.

## 4 Repository

Tutto il materiale prodotto dal team QuiXoft è disponibile online all'indirizzo <http://trac.assembla.com/sigeol/browser>.

Il software utilizzato per il versionamento dei file è Subversion. Ogni commit effettuata da un qualsiasi membro del team QuiXoft deve essere accompagnata da una chiara e sintetica descrizione. Tale descrizione deve contenere traccia della nuova versione dei file che vengono aggiornati o aggiunti.

### 4.1 Versione

Ogni file o documento deve essere accompagnato dalla sua relativa versione. Si useranno 3 cifre (es: 0.1.0): la prima cifra sta ad indicare il rilascio di una nuova versione del file, la seconda cifra rappresenta una modifica e la terza cifra indica una correzione. La versione di un documento, oltre che nel commento alla commit, deve assolutamente essere presente all'interno di



ogni file di testo creato dal team QuiXoft, sia esso un documento o un file di codice.

## 5 Documentazione

### 5.1 Template documenti

Ogni documento redatto dal team QuiXoft dovrà essere creato utilizzando il relativo template `LATEX` e successivamente esportato in Pdf per garantirne la giusta portabilità.

Se il template non dovesse più calzare con le esigenze di documentazione del team QuiXoft, qualsiasi componente potrà richiederne la modifica al Responsabile di Progetto.

#### 5.1.1 Documenti di resoconto incontri

I documenti di resoconto incontri dovranno essere creati usando il template `TEMPLATE_RESOCONTO.TEX` disponibile all'interno del repository nella cartella `/DOC/INCONTRI/TEMPLATE/`. Non saranno ammesse modifiche al layout e alla formattazione di tale template.

#### 5.1.2 Documenti ufficiali interni ed esterni

I documenti ufficiali interni ed esterni prodotti durante le varie attività aziendali dovranno essere redatti usando il template `TEMPLATE_UFFICIALE.TEX` disponibile nella cartella `/DOC/UFFICIALI/TEMPLATE/`. Come sopra, si raccomanda di aderire coerentemente al template, completando tutti i campi richiesti, senza apportare alcuna modifica.

### 5.2 Norme tipografiche

I documenti redatti, oltre che seguire la struttura dei template messi a disposizione, dovranno anche rispettare le seguenti convenzioni tipografiche:

- gli indirizzi internet e i percorsi di cartella dovranno essere scritti in corsivo utilizzando il comando `LATEX` `/textit` (esempio: `http://www.assembla.com`);
- i riferimenti a documenti esterni o a file (comprensivi di estensione) andranno scritti usando il comando `LATEX` `/textsc` (esempio: `TEMPLATE_RESOCONTO.TEX`);
- i termini definiti nel glossario verranno contrassegnati da una sottolineatura con il comando `LATEX` `/underline` (esempio: `template`);

- punto, virgola, punto e virgola, due punti, punto esclamativo e punto interrogativo sono attaccate alla parola che li precede, mentre sono separate con uno spazio dalla parola che li segue;
- parentesi (di ogni tipo), virgolette e trattini sono sempre attaccate al testo che delimitano, e separate con uno spazio dal resto del testo;
- i puntini di sospensione si scrivono attaccati alla parola che li precede;
- gli accenti in italiano sono sempre gravi, tranne nelle parole accentate che finiscono in -ché, in -né e in -sé.

### 5.3 Nominazione dei file di documentazione

#### 5.3.1 Resoconto incontri interni ed esterni

Ogni documento di resoconto di un incontro dovrà essere salvato sul repository del team QuiXoft in doppio formato:

- in formato Pdf nella cartella /DOC/INCONTRI/PDF, con estensione .pdf
- in formato sorgente L<sup>A</sup>T<sub>E</sub>X nella cartella /DOC/INCONTRI/TEX, con estensione .tex

Al fine di non creare incongruenze, file con estensioni diverse da quelle sopra indicate saranno eliminati dal repository.

I file dovranno essere nominati secondo i seguenti criteri:

{data}-{sigla}

dove:

- {data} indica, nel formato AAAA-MM-GG, la data dell'incontro
- {sigla} indica il tipo di incontro

I tipi di resoconto degli incontri che è possibile organizzare sono caratterizzati dalle seguenti sigle:

- RIO: resoconto incontro interno obbligatorio
- RIF: resoconto incontro interno facoltativo
- RIC: resoconto incontro con il committente
- RIE: resoconto altri incontri esterni



### 5.3.2 Documenti ufficiali interni ed esterni

I documenti ufficiali prodotti dal team QuiXoft dovranno essere salvati sul repository in due modi:

- in formato Pdf nella cartella  
/DOC/UFFICIALI/{NOME\_DOCUMENTO}/PDF
- in sorgente  $\text{\LaTeX}$  nella cartella  
/DOC/UFFICIALI/{NOME\_DOCUMENTO}/TEX  
Insieme al file sorgente  $\text{\LaTeX}$  dovranno essere inseriti nella medesima cartella anche tutti gli altri file necessari alla compilazione (immagini che differiscano da quelle usate nel template, grafici, ecc...)

I file dei documenti ufficiali interni ed esterni dovranno essere nominati mettendo in sequenza le seguenti parti:

1. Nome del documento, tutto in minuscolo (gli eventuali spazi dovranno essere sostituiti con un underscore)
2. Versione del documento (tra il nome del documento e la versione dovrà essere presente un underscore, così come tra i vari numeri che compongono la versione)
3. Estensione del file (salvo diverse indicazioni, l'estensione sarà sempre .pdf per i documenti esportati in formato Pdf e .tex per i sorgenti  $\text{\LaTeX}$ )

Ad esempio, il presente documento alla versione 1.0.0 dovrà essere così nominato: norme\_di\_progetto\_1.0\_0.pdf

### 5.4 Approvazione e verifica documentazione

Una volta redatto, un qualsiasi documento deve essere marcato come "Preliminare" nel campo "Stato" del relativo template  $\text{\LaTeX}$ .

In seguito tale documento deve essere sottoposto ad una fase di controllo da parte del Verificatore, il cui compito è quello di notificare eventuali incongruenze al redattore del documento. Spetta a quest'ultimo apportare le dovute modifiche e sottoporre nuovamente al Verificatore il documento. Una volta che il Verificatore avrà segnalato come "verificato" il documento, il Responsabile provvederà all'approvazione ed al cambio di stato in "Formale".

## 6 Presentazioni

In occasione di ogni revisione, sia formale sia informale, deve essere organizzata una presentazione che illustri il proseguimento del progetto al committente o al docente di riferimento.

Ogni membro del team QuiXoft è tenuto a partecipare alle revisioni ed a esporre la sua parte dei contenuti richiesti. Tutti i componenti devono dare il loro contributo alla presentazione: il carico temporale durante la presentazione deve essere equamente distribuito tra tutti i presenti.

Assenze a tali revisioni dovranno essere giustamente motivate.

Il tempo massimo che una presentazione può occupare è fissato in 20 minuti: il superamento di tale limite avrà effetto sanzionatorio sul risultato del gruppo. Se la presentazione dovesse durare meno di 20 minuti, sarà dato spazio alle domande del committente o del docente di riferimento per colmare le eventuali lacune lasciate dall'esposizione.

### 6.1 Template presentazioni

Per ogni revisione a cui il team QuiXoft parteciperà dovrà essere creata una presentazione che, con una serie di slide, accompagni il proseguimento dell'esposizione dei contenuti.

Le presentazioni dovranno essere create usando il relativo template L<sup>A</sup>T<sub>E</sub>X messo a disposizione nel repository del team. Dovranno poi essere esportate in Pdf e mostrate agli interessati tramite proiettore. L'elaboratore usato per tale operazione sarà di volta in volta messo a disposizione da uno dei membri del team.

La scrittura dei contenuti dovrà seguire le norme tipografiche usate per la creazione della documentazione. Particolare attenzione dovrà essere posta nel limitare la quantità di testo presente in ogni slide: dovrà essere data priorità alla presenza di concetti e parole chiave, che poi saranno adeguatamente spiegati dal relatore.

E' consigliato l'uso di figure e illustrazioni, che aiutino a rendere più agevole la comprensione dei concetti discussi.

Non è fissato a priori un numero massimo di slide per ogni presentazione. Si tenga comunque conto del limite temporale sopra illustrato. Caratteristica obbligatoria è la presenza del numero della slide corrente e del totale, per dare modo al docente o al committente di quantificare i tempi.

## 7 Codice

### 7.1 Intestazione

Ogni file di codice prodotto dal team QuiXoft durante lo sviluppo del progetto SIGEOL dovrà contenere un intestazione che ne certifichi l'origine e le modifiche per facilitarne la tracciabilità e le operazioni di manutenzione. Tale intestazione deve essere così formattata:

*QuiXoft - Progetto "SIGEOL"*  
*NOME FILE: {nome del file, completo di estensione}*



*AUTORE: {cognome e nome dell'autore del file}*

*DATA CREAZIONE: {data di creazione nel formato GG-MM-AAAA}*

*REGISTRO DELLE MODIFICHE:*

*{data e resoconto delle modifiche, in ordine cronologico decrescente}*

## 7.2 Codice Ruby

La scelta di Ruby on Rails come framework per lo sviluppo del progetto porta già con sé una gran quantità di convenzioni riguardanti la stesura del codice, senza che debbano essere scelte dal team di sviluppo. La filosofia che sta alla base di tutto ciò è denominata *Convention over Configuration*, che prevede configurazione minima (o addirittura assente) per il programmatore che utilizza tale framework, obbligandolo a configurare solo gli aspetti che si differenziano dalle implementazioni standard o che non rispettano particolari convenzioni di denominazione o simili.

Di seguito sono elencate le convenzioni utilizzate negli elementi sviluppati con il linguaggio Ruby.

### 7.2.1 Convenzioni sui nomi

- **Variabili locali:** prima lettera minuscola, seguita da altri caratteri minuscoli. Se la variabile comprende più parole, queste andranno separate con un `_` (underscore). Esempio: `variabile_locale`
- **Variabili d'istanza:** si utilizza la stessa convenzione adottata nelle variabili locali, con l'aggiunta di un `@` (at) prima del nome. Esempio: `@variabile_istanza`
- **Variabili di classe:** si utilizza la stessa convenzione adottata nelle variabili locali, con l'aggiunta di una doppia `@` (at) prima del nome. Esempio: `@@variabile_classe`
- **Variabili globali:** si utilizza la stessa convenzione adottata nella variabili locali, con l'aggiunta di un `$` (dollar) prima del nome. Esempio: `$variabile_globale`
- **Costanti:** prima lettera maiuscola, seguita da altri caratteri maiuscoli. Se la variabile comprende più parole, queste andranno separate con un `_` (underscore). Esempio: `UNA_COSTANTE`
- **Metodi d'istanza:** prima lettera minuscola, seguita da altri caratteri minuscoli. Se il nome comprende più parole, queste andranno separate con un `_` (underscore). Esempio: `metodo_istanza`
- **Classi e moduli:** prima lettera maiuscola, seguita da altri caratteri minuscoli. Se il nome comprende più parole, la prima lettera di ogni parola deve essere maiuscola. Esempio: `UnaClasse`



- **Model:** si utilizza la stessa convenzione per le classi ed inoltre il nome dovrà essere il singolare (in lingua inglese) del nome della tabella del database a cui si riferisce. Esempio: `Order`
- **Controller:** si utilizza la stessa convenzione per le classi ed inoltre il nome dovrà essere il plurale (in lingua inglese) del nome del Model a cui si riferisce, seguito dalla parola `Controller`. Esempio: `OrdersController`
- **Tabelle del database:** prima lettera minuscola, seguita da altri caratteri minuscoli. Se il nome comprende più parole, queste andranno separate con un `_` (underscore). Inoltre il nome deve essere il plurale del model (in lingua inglese) a cui la tabella si riferisce. Esempio: `orders`
- **Chiave primaria:** il nome della chiave primaria dovrà essere `id`
- **Chiavi esterne:** il nome della chiave esterna dovrà essere il singolare (in lingua inglese) della tabella di riferimento, seguito da un `_` (underscore) e dalla parola `id`, con ogni carattere minuscolo. Esempio: `order_id`
- **Tabelle per le relazioni molti a molti:** concatenazione tramite `_` (underscore) dei nomi al plurale (in lingua inglese) dei model coinvolti in ordine alfabetico, con ogni carattere minuscolo. Esempio: `items_orders`
- **File:** ogni nome di file è caratterizzato dalla presenza di soli caratteri minuscoli e la concatenazione di più parole viene effettuata tramite `_` (underscore).

### 7.2.2 Altre convenzioni

- **Lunghezza delle righe:** la lunghezza massima delle righe è fissata in 80 caratteri, per agevolare la visualizzazione e l'impaginazione del codice, anche in caso di eventuali stampe
- **Indentazioni:** la convenzione prevede di non utilizzare Tab ma Soft Tab (ovvero tab convertite in automatico in spazi) o spazi. La dimensione di ogni Soft Tab è fissata a 2 caratteri
- **Commenti:** i commenti sul codice seguono le convenzioni di Rdoc. Per maggiori informazioni consultare la relativa documentazione.

Tutto quanto non esplicitamente indicato è da intendersi regolamentato dalle norme fissate dal framework Ruby on Rails, dal linguaggio di programmazione Ruby e dai vari plugin che verranno utilizzati durante lo sviluppo del progetto. Per la lista completa delle regole da rispettare è sufficiente consultare la relativa documentazione.

### 7.3 Codice Java

Data la scelta di utilizzare un algoritmo per il calcolo dell'orario già scritto, in linguaggio Java, e la necessità di aggiungere ad esso piccole funzionalità e moduli, il team ha scritto il codice Java attenendosi alle seguenti convenzioni.

#### 7.3.1 Convenzioni sui nomi

- **Variabili:** prima lettera minuscola, seguita da altri caratteri minuscoli. Se la variabile comprende più parole, la lettera iniziale di ogni parola (tranne la prima) dovrà essere maiuscola. Esempio: `variabileLocale`
- **Metodi:** valgono le stesse convenzioni adottate per le variabili
- **Classi:** valgono le stesse convenzioni adottate per le variabili, escluso il fatto che la prima lettera deve essere maiuscola. Esempio: `ClassePubblica`
- **File:** il nome di ogni file deve essere esattamente uguale al nome dell'unica classe pubblica contenuta in esso, seguito dall'estensione ".java". Esempio: `ClassePubblica.java`

#### 7.3.2 Altre convenzioni

- **Lunghezza delle righe:** segue la convenzione omonima per il codice Ruby
- **Indentazioni:** similmente alla omonima convenzione per Ruby è previsto l'utilizzo dei Soft Tab oppure degli spazi. La dimensione dei Soft Tab, per coerenza con il codice non scritto dal team, è fissata in 4 spazi.

## 8 Compiti

### 8.1 Ticket

Appoggiandosi al portale <http://www.assembla.com> il Responsabile dovrà creare un ticket per ogni singolo compito da assegnare ad uno o più membri del team QuiXoft, comprese le operazioni di verifica. Sarà poi cura di quest'ultimo accettare o rifiutare il ticket.

Al momento della creazione di ciascun ticket è indispensabile stimare le ore necessarie per portarlo a termine e completare correttamente tutti i campi dati richiesti.

Una volta completato, il ticket assegnato dovrà essere marcato come "Close as fixed". Dovranno altresì essere indicate con precisione le ore che sono state necessarie al suo completamento.



Caso particolare di tutto ciò appena elencato sono i ticket relativi al Responsabile: quest'ultimo dovrà crearli, accettarli e gestirli autonomamente, e segnalare le ore impiegate per portarli a compimento come qualsiasi altro membro del team QuiXoft.

E' importante e necessario che tutti i ticket siano tracciati con le ore impiegate a completarli per tenere sotto controllo le risorse e i relativi costi e per non rischiare di sfiorare le stime illustrate nel PIANO DI PROGETTO.

### 8.2 Milestone

I ticket assegnati ai vari membri del team QuiXoft sono raggruppati in milestone e gestiti sempre attraverso il portale <http://www.assembla.com>.

Il raggiungimento di un milestone rappresenta il completamento di uno degli obiettivi che il Responsabile ha stabilito in fase di definizione del progetto stesso.

## 9 Strumenti

Per facilitare il lavoro di tutti i membri del team QuiXoft e per rendere interoperabili i loro risultati saranno ora elencati gli strumenti da usare per svolgere i vari compiti assegnati.

Salvo diverse indicazioni, è fatto obbligo lo sviluppo di tutto il progetto con i medesimi software e lo stesso sistema operativo, per non incappare nei vari problemi che sorgerebbero nell'usare una gran varietà di ambienti diversi.

Per ogni membro del team QuiXoft, tutti i software dovranno essere aggiornati alla versione riportata in seguito: non è consigliato l'utilizzo di software in diverse versioni, così come non sono consigliati aggiornamenti in corso d'opera, per evitare eventuali incompatibilità e malfunzionamenti.

Eccezione a quanto appena detto, durante la fase di collaudo del sistema SIGEOL sarà necessario testare il prodotto sotto vari sistemi operativi e con diversi browser, per verificarne l'effettiva portabilità.

- tutto il progetto dovrà necessariamente essere sviluppato all'interno di un sistema operativo **GNU/Linux**. La scelta di distribuzione e versione è lasciata ai singoli membri del team QuiXoft, ma è comunque consigliabile l'utilizzo di un sistema operativo basato su Debian (come, per esempio, Ubuntu) per la facilità di gestione e per la qualità dell'architettura e dell'infrastruttura.
- per la scrittura e la compilazione di tutta la documentazione e delle presentazioni verrà usato il programma **Kile** (KDE Integrated LaTeX Environment) alla versione 2.0.1. Per la compilazione dei sorgenti  $\text{\LaTeX}$  verrà utilizzato PDFLaTeX, incluso nell'installazione di Kile.

- l'intero progetto verrà sviluppato usando **NetBeans** alla versione 6.5. La scelta del linguaggio di programmazione e del relativo framework è caduta su **Ruby on Rails** versione 2.2.2, eccellente per le esigenze del progetto SIGEOL e perfettamente integrato all'interno di NetBeans. Quest'ultimo mette a disposizione notevoli caratteristiche che facilitano lo sviluppo con Ruby on Rails. NetBeans verrà anche utilizzato come editor per la creazione di tutti gli schemi UML necessari.
- il DataBase Management System (DBMS) scelto per lo sviluppo del progetto è MySQL 5.0.67. Viene consigliato anche l'uso di MySQL Query Browser versione 1.2.12, ma non sono tuttavia vietati altri tool di interrogazione e di consultazione del database. Ci si potrà facilmente interfacciare al DBMS scelto fruttando le funzionalità offerte da Active Record, comoda libreria messa a disposizione per Ruby on Rails.

Per gli specifici strumenti da adottare per lo svolgimento delle attività di verifica, validazione e controllo qualità si prega di consultare il PIANO DI QUALIFICA alla sua ultima versione.

La lista appena conclusa è da intendersi provvisoria ed incompleta: col procedere dello sviluppo del progetto potranno sorgere ulteriori esigenze da parte del team QuiXoft, che dovranno essere soddisfatte con un aggiornamento o una estensione degli strumenti da utilizzare.



### Diario delle modifiche

DATA	VERSIONE	MODIFICA
09-06-2009	2.0.0	Approvazione del responsabile e passaggio in stato di Formale
07-06-2009	1.6.0	Correzioni ortografiche varie
06-06-2009	1.4.0	Aggiunta sezione sul codice Java e correzione sezione codice Ruby
07-03-2009	1.3.2	Correzioni varie
09-02-2009	1.3.1	Correzione dei problemi riscontrati nella RPP
06-02-2009	1.3.0	Aggiornate le convenzioni sul codice
21-01-2009	1.2.1	Correzioni varie
20-01-2009	1.2.0	Aggiunta la sezione "Strumenti"
18-01-2009	1.1.0	Aggiunta la sezione "Presentazioni", correzioni varie
09-12-2008	1.0.0	Approvazione del Responsabile e passaggio di stato in "Formale"
07-12-2008	0.3.1	Correzioni varie
05-12-2008	0.3.0	Aggiunta delle norme tipografiche
04-12-2008	0.2.0	Aggiunta del sommario
28-11-2008	0.1.1	Correzione di alcuni errori grammaticali riportati dal verificatore
27-11-2008	0.1.0	Prima stesura del documento