

Simulatore di Gestione della Memoria di un Elaboratore SiGeM



<http://stylosoft.altervista.org>
stylosoft@gmail.com

Norme interne

08 Marzo 2008

Documento interno – v3.0

Norme_interne_3.0.pdf

Redazione:

Alberto Zatton

Revisione:

Daniele Bonaldo

Approvazione:

Luca Rubin

Lista di distribuzione:

Stylosoft

Registro delle modifiche:

Versione	Data	Descrizione delle modifiche
3.0	08/03/2008	Correzioni grammaticali
2.1	13/02/2008	Aggiunta norme di codifica
2.0	26/01/2008	Sistemazione hyperlink ed errori
1.3	23/01/2008	Inserimento stili ed indice automatizzato
1.2	18/01/2008	Aggiunte sezioni: segnalazioni interne e norme di codifica - progettazione
1.1	15/01/2008	Aggiunto repository SVN
1.0	06/12/2007	Correzione errori vari grammaticali, aggiustamento indice
0.2	04/12/2007	Sistemazione indice e ampliamento punto 5

Versione:
3.0

Creazione documento:
19/11/07

Ultima modifica:
08/03/08

Pagina 1 di 8

Sommario

Questo documento fissa le regole base che l'azienda intende seguire per comunicare con i soggetti interni ed esterni, regolando le modalità di stesura dei documenti, i programmi utilizzati, e le modalità di consegna.

Indice

1	Comunicazioni.....	3
1.1	Comunicazioni interne.....	3
1.2	Comunicazioni esterne.....	3
2	Convocazioni ed incontri.....	3
3	Gestione dei documenti ufficiali.....	3
3.1	Denominazione dei documenti.....	3
3.2	Gestione documenti interni.....	4
3.3	Gestione documenti esterni.....	4
3.4	Template dei documenti.....	4
3.5	Approvazione documenti.....	4
3.6	Distribuzione documenti.....	5
4	Versionamento.....	5
4.1	Connessione al repository.....	5
4.2	Utilizzo del repository.....	5
5	Strumenti e ambienti di sviluppo.....	5
6	Segnalazioni interne.....	6
7	Norme di progettazione e codifica.....	7
7.1	Introduzione.....	7
7.2	Nomenclatura e stile di codifica.....	7
7.3	Progettazione.....	7
7.4	Diagrammi delle classi.....	7
7.5	Codifica.....	7
7.5.1	Intestazione dei file.....	7
7.5.2	Documentazione del codice.....	8
7.5.3	Indentazione del codice.....	8

1 Comunicazioni

Modalità con cui vengono diffusi avvisi e comunicazioni all'interno e all'esterno del Team.

1.1 Comunicazioni interne

Ogni tipo di comunicazione e avviso interno all'azienda sarà notificato tramite posta elettronica a tutto il Team di sviluppo. Ciascuna di queste e-mail sarà mandata all'indirizzo pubblico dell'azienda e successivamente inoltrata in modo automatico a ogni componente del Team, all'indirizzo personale. L'oggetto dell'e-mail specificherà di volta in volta il tipo della comunicazione. Altri tipi di comunicazione potranno essere effettuate attraverso l'utilizzo delle "Issues" generabili nella sezione omonima all'indirizzo sigem.googlecode.com. Per maggiori dettagli sull'utilizzo delle Issues vedere il [punto 6](#) di questo documento.

1.2 Comunicazioni esterne

Le comunicazioni esterne dovranno essere effettuate dall'indirizzo mail dell'azienda: stylosoft@gmail.com. Ogni componente del Team dovrà necessariamente utilizzare questo indirizzo come mezzo di comunicazione con l'esterno, in modo tale che tutti possano accedere a tali informazioni. Con questo indirizzo di posta elettronica, il Committente potrà mettersi in contatto con ogni componente del gruppo in modo veloce e simultaneo.

2 Convocazioni ed incontri

Tutte le convocazioni e gli incontri saranno segnalati tramite comunicazioni interne, secondo quanto stabilito al [punto 1.1](#). Sono previsti due incontri settimanali per il controllo sullo stato di avanzamento dei lavori e per l'assegnazione dei compiti ai vari componenti del gruppo. Tali incontri avranno una frequenza di due giorni alla settimana. Le date e il luogo di tali incontri saranno decise via via in base alla disponibilità dei vari componenti.

3 Gestione dei documenti ufficiali

3.1 Denominazione dei documenti

Ogni documento dovrà riportare nel nome, il tipo del documento separando le parole che lo compongono con "_". Una volta approvati, i documenti dovranno essere

convertiti in formato pdf. Nel nome dovrà essere presente, inoltre la versione del documento.

3.2 Gestione documenti interni

Internamente al Team, i documenti saranno disponibili presso il *repository* ufficiale all'indirizzo web: <http://sigem.googlecode.com/svn/trunk/doc>. E' disponibile una comoda interfaccia web per l'esplorazione dei file, mentre per la loro gestione sarà possibile utilizzare un qualsiasi client *SVN*. Per chiarimenti sull'utilizzo di *SubVersion* vedere il [punto 4](#) del presente documento relativo al versionamento dei file.

3.3 Gestione documenti esterni

Dall'esterno i documenti saranno resi noti con l'utilizzo del comando di consegna previsto dal corso. Tali documenti dovranno essere racchiusi in un file compresso. Al momento della decompressione i file dovranno risultare contenuti in un'unica cartella.

3.4 Template dei documenti

Ogni documento dovrà attenersi allo schema riportato nel file "modello.odt" disponibile nel *repository* ufficiale del Team. In particolare è stato adottato un nuovo stile dalla versione 2.0 di questo documento. Esso è composto da quattro modelli di stile: ["Corpo del Testo" (Arial; 12pt; Normale; Rientro e spaziatura: 0,3 - 0,3 - 1,0 - 0,2 - 0,2)]; ["Intestazione" usata per non numerati (Arial; 15pt; Normale; Rientro e spaziatura: 0 - 0 - 0 - 0,25 - 0,25)]; ["Intestazione 1" usata per paragrafi numerati (Arial; 14pt; Normale; Rientro e spaziatura: 0 - 0 - 0 - 0,25 - 0,25)]; ["Intestazione 2" usata per sottoparagrafi numerati (Arial; 13pt; Normale; Rientro e spaziatura: 0 - 0 - 0 - 0,25 - 0,25)]; ["Indice" (Arial; 12pt; Normale; Rientro e spaziatura: 0 - 0 - 0 - 0,25 - 0,25)]. Il corpo dei documenti dovrà essere giustificato. Dovranno necessariamente essere indicati: data di redazione, nome e versione del documento, nome e cognome del redattore, nome e cognome del responsabile, nome e cognome del verificatore, lista di distribuzione, registro delle modifiche, sommario e indice. Tutti i termini che saranno inseriti nel glossario dovranno essere scritti in *corsivo*. A partire dalla versione 2.0 di questo documento l'indice è generato automaticamente e ancorato.

3.5 Approvazione documenti

Ogni documento dovrà specificare la data e il nome del responsabile (per approvazione) e quello del verificatore (per revisione). Per ulteriori informazioni sulle modalità di verifica, consultare il documento "Piano_di_qualifica_3.0.pdf".

3.6 Distribuzione documenti

I documenti saranno disponibili ai destinatari specificati nella lista di distribuzione tramite le modalità elencate nei [punti 3.2](#) e [3.3](#) di questo documento. Per la modifica di un documento, il redattore, dovrà accedervi in modo esclusivo, per evitare situazioni di inconsistenza. L'accesso esclusivo ad un file si ottiene attraverso l'utilizzo di "lock", previsti dal server *SVN*.

4 Versionamento

Ogni documento sarà dotato di un "registro delle modifiche" in cui saranno riportati i cambiamenti apportati in ordine di data decrescente. Ogni modifica coinciderà con un diverso numero di versione. Il numero di versione consiste di 2 cifre. La prima aumenta al passaggio ad una revisione successiva, la seconda aumenta in seguito a una modifica. All'indirizzo web <http://sigem.googlecode.com> è presente un *repository SVN*, presso il quale saranno mantenuti i documenti generati dal Team. Ogni qualvolta uno di questi documenti sarà modificato, verrà assegnato un numero di revisione progressivo, e il numero di versione dovrà aumentare. Il numero di versione va riportato solo all'interno del documento e non nel nome del file. Grazie all'utilizzo di *SubVersion* saranno possibili operazioni di "Rollback" da una revisione ad una precedente.

4.1 Connessione al repository

Ogni componente del Team che voglia connettersi al *repository* ufficiale, dovrà dotarsi di un account Google. Per l'accesso al *repository* con un client *SVN*, ogni componente dovrà autenticarsi con il proprio username e password, in modo tale da poter risalire all'autore di ogni operazione effettuata nel *repository*.

4.2 Utilizzo del repository

Tramite le operazioni di "update" e "commit" sarà possibile sincronizzarsi al server dei file. Ad ogni operazione di "commit" si dovrà specificare (nel messaggio) un breve riassunto dell'operazione che si sta eseguendo, in generale specificando la modifica apportata al file.

5 Strumenti e ambienti di sviluppo

Per la redazione dei documenti si utilizzerà OpenOffice2 (www.openoffice.org) in ambiente Linux o Windows. L'ambiente di sviluppo utilizzato per la codifica e la progettazione sarà NetBeans 6 (www.netbeans.org). Per la generazione di diagrammi

di Gantt verrà utilizzato GanttProject v.2.0.6 (ganttproject.biz).

I diagrammi UML saranno generati con gli strumenti incorporati in NetBeans 6. Ogni programma utilizzato dal Team sarà dotato, generalmente, di licenza GPL. In ogni caso verranno usati software non a pagamento per non incidere ulteriormente sul costo del prodotto.

Per la verifica e la validazione del software, verranno utilizzati dei plugin per NetBeans6, che permettono una maggiore integrazione con l'ambiente di sviluppo utilizzato dal Team; questa maggiore integrazione permette di ottenere risultati più immediati con un margine d'errore minore. I plugin che saranno utilizzati sono:

- SQE (Software Quality Environment)
- JUnit
- Code Coverage Plugin

6 Segnalazioni interne

E' a disposizione del Team uno strumento di segnalazione, utile per comunicare errori, richieste di aiuto, richieste di attività e modifiche. Tale strumento è accessibile all'indirizzo web <https://sigem.googlecode.com> nella sezione Issues. La generazione di nuove Issues è possibile previa autenticazione con il proprio account Google. La visione di un Issue è invece effettuabile anche in modo non autenticato. Ogni segnalazione è identificata univocamente da un numero progressivo assegnato in modo automatico. Al momento della generazione di un nuovo Issue, si dovranno specificare alcuni parametri (tramite l'utilizzo della comoda interfaccia web) che permettano di classificare in modo dettagliato la natura della segnalazione. Tra i vari parametri dovranno essere specificati i file oggetto della comunicazione oltre ai destinatari. Così facendo ogni destinatario riceverà una mail di notifica. Ogni modifica apportata ad un Issue verrà comunicata ai destinatari. Un nuovo Issue rimane in stato "Open" fino a che esso non viene concluso con la relativa operazione di chiusura ("Close"). Esistono degli stati intermedi decidibili a seconda delle esigenze del segnalatore. Per evitare la generazione di Issue soggette a micromodifiche, è desiderabile che ogni componente provveda all'invio di una segnalazione unica a fine giornata.

7 Norme di progettazione e codifica

7.1 Introduzione

Java sarà il linguaggio di programmazione con cui verrà realizzato il progetto.

7.2 Nomenclatura e stile di codifica

Il nome delle variabili, delle classi e di ogni altra entità, dovrà essere espresso in italiano, secondo le convenzioni di Java.

Per fare in modo che il codice sia chiaro è facilmente mantenibile, ogni programmatore dovrà eseguire la codifica nel rispetto delle JCC (Java Code Conventions). Tali norme sono reperibili al sito <http://java.sun.com/docs/codeconv> in vari formati. Per completezza viene inoltre segnalato il link (<http://www.bigatti.it/download/JavaManualeDiStile.pdf>) che chiarifica e integra le norme JCC. Essendo entrambi i documenti basati sulle JCC non potranno sorgere conflitti; per questo motivo il programmatore potrà attingere ad entrambi i documenti.

7.3 Progettazione

La progettazione dovrà prevedere l'utilizzo di almeno tre *design pattern* i quali dovranno essere specificati nella Specifica Tecnica.

7.4 Diagrammi delle classi

La progettazione mediante diagrammi delle classi dovrà essere conforme al linguaggio UML2. L'ambiente di sviluppo da utilizzare sarà NetBeans6 che permette un'ottima integrazione tra Java e UML.

7.5 Codifica

7.5.1 Intestazione dei file

Ogni file dovrà essere dotato di un'intestazione seguendo l'esempio riportato in seguito:

```
/* Azienda: Stylosoft
* Nome file: NomeClasse.java
* Package: nomePackage
* Autore: Nome Cognome
```

- * Data: dd/mm/yyyy
- * Versione: x.y
- * Licenza: open-source
- * Registro delle modifiche:
 - * - v.x.y (dd/mm/yyyy): Descrizione modifica
- */

7.5.2 Documentazione del codice

Per una più semplice verifica e manutenzione del codice, si dovrà dotare ogni file della relativa documentazione JavaDoc. Tale documentazione dovrà essere in italiano. Per maggiori informazioni su JavaDoc consultare il sito <http://java.sun.com/j2se/javadoc/> .

7.5.3 Indentazione del codice

Sarà necessario verificare ed eventualmente correggere l'ampiezza dell'indentazione dell'editor utilizzato, settando a quattro l'ampiezza delle tabulazioni. Di default NetBeans utilizza una giusta misura di indentazione.