

# Verbali Esterni

 $Gruppo\ SWEetBIT\ -\ Progetto\ SWEDesigner$ 

#### Informazioni sul documento

miormazioni sai accamento		
Versione	1.0.0	
Redazione	Sebastiano Bertolin	
Verifica	Da inserire	
Approvazione	Da inserire	
Uso	Esterno	
Distribuzione	Prof. Tullio Vardanega	
	Prof. Riccardo Cardin	
	Gruppo SWEetBIT	

#### Descrizione

Questo documento traccia i verbali di tutte le riuioni esterne, con il committente od il proponente, svolte dal gruppo SWEetBIT.

## INDICE

# Indice

1	$\operatorname{Riu}$	nione 1	2
	1.1	Informazioni sulla riunione	2
	1.2	Domande e Risposte	9

## 1 Riunione 1

#### 1.1 Informazioni sulla riunione

• **Data:** 23/02/2017

• Luogo: Zucchetti - sede di Padova , via Giovanni Cittadella 7

• Ora: 16:30

• Durata: 90 min

• Argomento: Chiarimenti dei requisiti sul capitolato<sub>G</sub>

• Partecipanti Interni: Davide Santimaria - Fabio Massignan - Gianmarco Salmistraro - Malick Bodian - Salvatore Pilò - Sebastiano Bertolin

• Partecipanti Esterni: Gregorio Piccoli

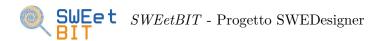
#### 1.2 Domande e Risposte

• Come deve essere la qualità del  $codice_G$  generato dai  $diagrammi_G UML_G$ ? Per rispondere al meglio alla domanda, ci si vuole da prima focalizzare su un altro punto, ovvero il perché sia stato suggerito il tema dei giochi da tavolo su questo  $capitolato_G$ .

Ebbene nel  $dominio_G$  dei giochi da tavolo, la ripetitività è molto elevata; Basti pensare al gioco della dama o degli scacchi, entrambi condividono la stessa scacchiera di gioco, anche il Monopoli ha una scacchiera, che si diversifica per colori e tipi di caselle , ma è pur sempre una scacchiera. Focalizzarsi su di un specifico ambito, in questo caso i giochi da tavolo, aiuta a migliorare la qualità del  $codice_G$  (vista anche la possibilità del suo riutilizzo) , in quanto se si decidesse di rappresentare ogni contesto, risulterebbe difficoltoso generare del  $codice_G$  che si adatti al meglio in ogni situazione.

Sfruttare il fattore del riutilizzo sicuramente aiuta, ma la scelta che incide, è il modo in cui si sceglie di disporre i  $diagrammi_G$ , ovvero se visualizzarli per  $layer_G$ , suddividendoli per categorie ad esempio, oppure disporre il tutto su un unico  $layer_G$  ed utilizzare un sistema di evidenziazione o  $zoom_G$  od altre tecniche. Tale scelta è importante perchè riflette quanto più complesso e specifico possa essere il progetto che l' $utente_G$  vuole realizzare, e da questo ne deriva anche la qualità del  $codice_G$ . La vera difficoltà sta nel generare  $codice_G$  dai  $diagrammi_G$  che rappresenteranno i  $metodi_G$ , per essi si suggerisce un approccio utilizzando l' $activity\ diagram_G$ .

Verbali Esterni Pagina 2 di 4



## • Nel nostro $designer_g$ dobbiamo includere dei $template_g$ ?

Si, a patto che venga scelto un  $dominio_G$  su cui il  $designer_G$  si basi. Mettere a disposizione ad esempio una scacchiera 8x8, evitando che l'utente finale debba crearsela da zero. Ovviamente i giochi da tavolo non son l'unico  $dominio_G$  su cui ci si può basare, la scelta potrebbe ricadere in altri settori, ma tale scelta deve esser fatta per poter avere dei  $diagrammi_G$  e quindi del  $codice_G$ . Il dominio fin ora citato è consigliato perchè offre molti approfondimenti sull'adattamento in  $diagrammi_G$  ed inoltre garantirà una fase di testing più piacevole.

### Se viene modificato il codice<sub>G</sub> generato, il diagramma<sub>G</sub> deve aggiornarsi anch'esso?

È un aspetto sicuramente interessante se si riuscisse ad implementare. Solitamente dopo la creazione dei  $diagrammi_G$  ed il rispettivo  $codice_G$ , le successive modifiche vengono apportate solo al  $codice_G$ ; in quanto il  $diagramma_G$  ha lo scopo principale da fungere da linea guida, rappresentando solo le classi principali, tralasciando nella visualizzazione quelle di supporto. Rappresentare tutte le classi che costituiscono il progetto,potrebbe ridurre la leggibilità del  $diagramma_G$ ; una soluzione potrebbe essere quella di celare o inserire in un layer diverso le classi di supporto. Visualizzare solo lo scheletro della classe senza l'implementazione dei  $metodi_G$  può esser una soluzione alla domanda posta.

# • L'applicazione deve essere solo $desktop_{\scriptscriptstyle G}$ o deve essere anche una $Web-App_{\scriptscriptstyle G}$ ?

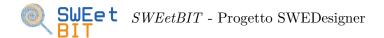
È preferita la web-application, ma la scelta non è vincolante. Lo scopo principale è entrare nell'ottica di fare un progetto usando molto i  $diagrammi_G$ .

## • Il $codice_{G}$ prodotto deve esser in formato $Java_{G}$ o $JavaScript_{G}$ ?

Nella fase di disegno dei  $diagrammi_G$ , in particolare a quelli dei  $metodi_G$ , si deve procedere in modo astratto , ovvero tracciando solo l'algoritmo necessario al  $metodo_G$  in esame. Procedendo con quest'ottica si può generare  $codice_G$  in entrambi i linguaggi. Proseguire in quest'ottica di pensare per specifiche risulta interessante ed una buona sfida , ma se si decidesse di pensare per programma, iI consiglio è quello di procedere inizialmente con Java, il quale è un linguaggio più controllato e verificato.

# L'applicazione dovrà esser disponibile su di un server oppure è sufficiente in locale?

Non è necessario l'acquisto di uno spazio su cui ospitare l'applicazione, eventualmente si possono utilizzare dei server gratuiti a tempo limitato come ad esempio Amazon, Heroku.



- Considerando che il numero di gruppi che aderiscono a tale  $capitolato_G$  è aumentato, alcuni dei requisiti opzionali son diventati obbligatori? I requisiti opzionali rimangono invariati, un aspetto su cui focalizzarsi è lo studio dell' $UML_G$ . Il progetto consiste nel creare un disegnatore che abbia il  $diagramma_G$  della classe con i rispettivi  $diagrammi_G$  dei  $metodi_G$ ; la soluzione di come rappresentare il collegamento tra questi  $diagrammi_G$ , è sicuramente il punto su cui ci si deve concentrare, proponendo anche modelli ibridi dei  $diagrammi_G$ .
- Ci devono esser dei controlli nella costruzione del  $diagramma_{c}$ ? e come devo restituire il  $codice_{c}$  generato?

Ciò che viene restituito può essere un pacchetto contenente i  $\mathit{file}_G$  oppure una visualizzazione testuale del  $\mathit{codice}_G$ ; l'importante è che venga visualizzato il risultato che l' $\mathit{utente}_G$  si aspetta. Per quanto riguarda i controlli, son un lavoro di cortesia, evitare che l'utente durante la fase di "disegno" faccia degli errori come ad esempio dimenticarsi il nome della classe è sicuramente apprezzato. Ovviamente se si desse la possibilità di aggiungere nel  $\mathit{diagramma}_G$  un rettangolo dove si possa aggiungere del  $\mathit{codice}_G$  manualmente, in quella situazione risulteerà impegnativo gestire automaticamente l'errore.

Verbali Esterni Pagina 4 di 4