

Project Reporting



TravelDream



Cirigliano Andrea

Di Sabatino Di Diodoro Serena

Sommario

1. Introduction

1.1 Purpose	3
-------------------	---

2. Size Estimation

2.1 Function Point Analysis (FPA)	4
---	---

3. Cost Estimation

3.1 COConstructive COst Model (COCOMO)	7
3.2 Conclusions	8

1. Introduction

1.1 Purpose

Lo scopo di questo documento è quello di stimare nel modo più oggettivo possibile l'effort e i costi di sviluppo del sistema TravelDream.

Non potendo chiaramente avvalerci di un approccio di tipo experience-based, utilizzeremo dei modelli predefiniti: la tecnica dei Function Points (IBM) per una stima delle dimensioni dell'applicazione e il modello COCOMO (Barry Boehm) per il calcolo dei tempi di consegna e i mesi-uomo necessari.

Tali valutazioni a posteriori ci permettono di mettere a confronto le stime ottenute con i dati reali.

2. Size Estimation

2.1 Function Point Analysis (FPA)

In questa sezione verrà stimato l'effort richiesto per progettare e completare l'applicazione in base alle funzionalità che il sistema fornisce. Le funzionalità del sistema verranno prima suddivise per tipo (function types), e per ciascuna di esse verrà data una dimensione in termini di complessità, in modo da poter calcolare il numero di Function Points come somma pesata dei punti. I pesi utilizzati sono quelli riportati nella tabella seguente:

Function Types		Weights	
	Simple	Medium	Complex
Input	3	4	6
Output	4	5	7
Inquiry	3	4	6
Internal Files	7	10	15
External Files	5	7	10

Internal Logic Files:

l'applicazione include un certo numero di ILFs che vengono utilizzati per gestire le informazioni di:

- Utenti: struttura semplice e con pochi campi (simple: x7)
- Componenti: tipo e numero di campi variabile a seconda della tipologia (medium: x10)
- Pacchetto viaggio: struttura semplice che funge solamente da contenitore di componenti esistenti (simple: x7)
- Acquisti: struttura semplice che lega un utente ad un pacchetto (simple: x7)

numero di FPs: $3 \times 7 + 1 \times 10 = 31$ FPs

External Logic Files:

L'applicazione è in grado di funzionare autonomamente senza interfacciarsi con applicazioni esterne, perciò il contributo degli ELF è di 0 FPs.

External Inputs:

Il sistema interagisce con il cliente registrato per permettere:

- Login/logout: operazioni semplici a cui si può assegnare il peso minimo (simple: x3)
- Salvataggio pacchetto personalizzato: operazione semplice (simple: x3)
- Acquisto pacchetto: (simple: x3)
- Eliminazione pacchetto personalizzato: operazione semplice (simple: x3)

Input relativi agli amministratori:

- Inserimento di un componente base e delle sue informazioni (simple: x3)
- Eliminazione componente (simple: x3)
- Inserimento nuovo pacchetto standard: richiede il recupero di tutti i componenti ed il rispetto dei vincoli (medium: x4)
- Eliminazione pacchetto standard: (simple: x3)

Input relativi a utenti non registrati:

- Registrazione: (simple: x3)

numero di FPs: $8 \times 3 + 1 \times 4 = 28$ FPs

External inquiries:

Operazioni elementari del sistema che richiedono input e output:

- Personalizzazione del pacchetto da parte di un utente: richiede il recupero di tutti i componenti ed il rispetto dei vincoli (medium: x4)
- Modifica di un pacchetto da parte di un amministratore: richiede il recupero di tutti i componenti ed il rispetto dei vincoli (medium: x4)
- Ricerca di un pacchetto tramite keyword (simple: x5)

numero di FPs: $2 \times 4 + 1 \times 5 = 13$ FPs

External Outputs:

- Visualizzazione di tutti i pacchetti: (simple: x4)
- Visualizzazione dettagli di un pacchetto: (medium: x5)
- Visualizzazione di tutti i componenti: (simple: x4)
- Visualizzazione dei pacchetti salvati: (medium: x5)
- Visualizzazione di un pacchetto cercato tramite id: (medium: x5)

numero di FPs: $2 \times 4 + 3 \times 5 = 23$ FPs

In totale il numero di Function Points è di 95 FPs.

Dato un indice indicativo di 6-7 ore per function point relative al tipo di progetto¹, la stima risultante è di 546 ore.

¹ Productivity Factor: <http://www.servicetechmag.com/i68/1112-4>

3. Cost Estimation

3.1 COnstructive COst MOdel (COCOMO)

Avendo sviluppato il progetto seguendo rigidamente lo schema waterfall, un utile strumento adatto alla stima del costo è il modello COCOMO.

L'analisi delle linee di codice effettuata sul file della prima implementazione conclusa mostra i seguenti risultati²:

Language	files	blank	comment	code
Java	34	633	124	1940
JavaServer Faces	24	120	0	1449
SQL	2	50	72	214
XML	12	3	0	159
Other	4	0	0	40
Total	76	806	196	3802

Considerata la natura semplificata del progetto si è deciso di applicare cocomo al livello basic, per applicazioni semplici (organic mode).

Size = 3802

M [mesi-uomo] = $2,4 \times 3,802^{1,05} = 9,75$

T [mesi] = $2,5 \times 3,802^{0,38} = 4,15$

La stima si riduce nel caso in cui dalle linee di codice si escludono quelle meno significative (blank e commenti):

Size = 3802 - blank - comments = 2800

M [mesi-uomo] = $2,4 \times 2,8^{1,05} = 7,075$

T [mesi] = $2,5 \times 2,8^{0,38} = 3,7$

² Effettuata con cloc v1.60 (cloc.sourceforge.net)

Il progetto in esame ha un livello di semplificazione non trascurabile rispetto ad un reale processo di sviluppo. E' stato perciò deciso di introdurre alcuni coefficienti correttivi nel calcolo cocomo:

Product Corrective coefficient	Level	Value
RELY – Required reliability	Low	0,88
DATA – Database size	Low	0,94
CPLX – product Complexity	Low	0,85

Per tutti gli altri coeficienti esistenti non specificati, si presuppone un valore nominale (x1).

Il valore di T non cambia, mentre la stima di M viene ridotta di un fattore complessivo di 0,703:

$$\mathbf{M \text{ [mesi-uomo]} = 7,075 \times 0,703 = 4,973}$$

3.2 Conclusions

Le stime T ed M di Cocomo riguardano le fasi di design, di sviluppo, e di testing. Secondo quanto specificato nel documento "Working Hours" queste tre fasi hanno richiesto un totale di circa 360 ore. I paramentri T ed M di Cocomo mostrano una sovrastima rispetto all'effort realmente impiegato; ciò è dovuto alla natura semplificata del progetto analizzato rispetto a progetti software reali, e alla tendenza di tutte le tecniche di calcolo di sovrastimare il costo per evitare la scarsità di risorse in fase di sviluppo.