



Università degli Studi di Parma
Facoltà di Scienze MM. FF. NN.
Corso di Laurea in Informatica
Ingegneria del Software

La fase di Implementazione

Giulio Destri



Ing. del Sw: Implementaz. - 1 Giulio Destri - © AreaSP for Univ. Parma, 2006

Scopo del modulo

Definire

**Proprietà e strutturazione
del processo di Implementazione
entro un progetto informatico**

Ing. del Sw: Implementaz. - 2 Giulio Destri - © AreaSP for Univ. Parma, 2006

Argomenti

- Definire il processo di Implementazione
- Caratteristiche importanti
- Linee guida
- Criteri di completezza

Ing. del Sw: Implementaz. - 3 [Giulio Destri - © AreaSP for Univ. Parma, 2006](#)

Definire il processo di Implementazione

- Partendo dai risultati della fase di progetto
- L'implementazione deve produrre il codice sorgente relativo e provvedere alla sua compilazione
- Seguendo opportuni standard
- Per avere un codice leggibile, efficiente e manutenibile

Ing. del Sw: Implementaz. - 4 [Giulio Destri - © AreaSP for Univ. Parma, 2006](#)

Glossario per l'Implementazione

- Definizione di tipo
- Metodi
- Attributi
- Istanza
- Collegamenti statici
- Collegamenti dinamici
- Polimorfismo
- Riuso

Ing. del Sw: Implementaz. - 5 [Giulio Destri - © AreaSP for Univ. Parma, 2006](#)

I vantaggi dell'OOP

- Uniformità
- Riuso
- Comprensibilità
- Flessibilità
- Stabilità

Ing. del Sw: Implementaz. - 6 [Giulio Destri - © AreaSP for Univ. Parma, 2006](#)

I fattori importanti nello sviluppo

- Linguaggio di programmazione usato
- Libreria di classi/componenti disponibile
- Strumenti software di sviluppo

Ing. del Sw: Implementaz. - 7 [Giulio Destri - © AreaSP for Univ. Parma, 2006](#)

I fattori importanti nello sviluppo - 2

- Qualità della progettazione di base
- Infrastruttura di base
- Rispetto dei principi generali
- Ambiente di lavoro
- Metodi dei singoli sviluppatori

Ing. del Sw: Implementaz. - 8 [Giulio Destri - © AreaSP for Univ. Parma, 2006](#)

Suddivisione dei linguaggi

- Non orientati agli oggetti (es. Basic, Prolog, Cobol, Fortran, LISP, C...)
- Basati sugli oggetti (es. Ada, VisualBasic...)
- Basati sulle classi
- Orientati agli oggetti (es. Delphi, Java, Objective C)
- Ibridi (es. C++, Ada95, VB.NET...)
- Pienamente orientati agli oggetti (es. Smalltalk, Eiffel...)

Ing. del Sw: Implementaz. - 9 [Giulio Destri - © AreaSP for Univ. Parma, 2006](#)

Librerie di classi/componenti/utility

- Facenti parte del linguaggio (es. Java Swing)
- Pubbliche (es. STL)
- Proprietarie (es. RogueWave, IBMTL)

Ing. del Sw: Implementaz. - 10 [Giulio Destri - © AreaSP for Univ. Parma, 2006](#)

Gli strumenti

- Debugger
- Build/mane
- Generatori dati campione
- Gestione dei sorgenti (es. versioni, configurazione...)
- Browser delle classi

Ing. del Sw: Implementaz. - 11 [Giulio Destri - © AreaSP for Univ. Parma, 2006](#)

Metodi di lavoro

- Convenzioni di denominazione degli identificatori
- Convenzioni di impostazione del codice
- Denominazione e Gestione delle versioni
- Pianificazione dell'integrazione

Ing. del Sw: Implementaz. - 12 [Giulio Destri - © AreaSP for Univ. Parma, 2006](#)

Regole per l'integrazione

- Velocità e tempo di risposta dell'applicazione risultante
- Quantità di dati e velocità del loro flusso
- Condizioni di run-time (es. concorrenza nell'uso delle risorse)

Ing. del Sw: Implementaz. - 13 [Giulio Destri - © AreaSP for Univ. Parma, 2006](#)

Strategie per l'integrazione

- Integrazione big-bang
- Integrazione top-down
- Integrazione bottom-up
- Integrazione di build
- Integrazione a strati

Ing. del Sw: Implementaz. - 14 [Giulio Destri - © AreaSP for Univ. Parma, 2006](#)

L'implementazione delle classi

- Lo sviluppo basato sui componenti
- Implementazione delle interfacce utente
- Implementazione della logica business
- Implementazione della base di dati

Ing. del Sw: Implementaz. - 15 [Giulio Destri - © AreaSP for Univ. Parma, 2006](#)

I componenti: granularità

- Funzioni, procedure, metodi
- Classi, oggetti, moduli
- Librerie, Widget/controlli
- Componenti
- Sottosistemi, servizi
- Applicazioni



Complessità

Ing. del Sw: Implementaz. - 16 [Giulio Destri - © AreaSP for Univ. Parma, 2006](#)

Le interfacce utente

- Coerenza (es. nella gestione eventi)
- Regole di denominazione
- Messaggi chiari per gli utenti
- Uso chiarificante di font e colori
- Raggruppamenti chiarificanti di elementi
- Evitare potenziali azioni distruttive
- Gestire disattenzioni dell'utente

Ing. del Sw: Implementaz. - 17 [Giulio Destri - © AreaSP for Univ. Parma, 2006](#)

Check-list: sintassi e struttura

- Leggibilità del codice
- Usare le indentazioni
- Limitare lunghezza (es. 80 caratteri)
- Un'istruzione per riga
- Evitare abbreviazioni incomprensibili
- Definire un sistema di denominazione chiaro e coerente e rispettarlo

Ing. del Sw: Implementaz. - 18 [Giulio Destri - © AreaSP for Univ. Parma, 2006](#)

Check-list: commenti

- Non lesinare i commenti, soprattutto nei casi di ambiguità
- Descrivere prima di ogni metodo il suo funzionamento
- Non lesinare la documentazione di accompagnamento (es. file README)

Ing. del Sw: Implementaz. - 19 [Giulio Destri - © AreaSP for Univ. Parma, 2006](#)

Check-list: livello sistema

- Evitare l'uso di variabili globali
- Un metodo dovrebbe usare come dati solo parametri e variabili locali inizializzate
- Implementare una gestione efficace degli errori (es. uso corretto delle eccezioni)

Ing. del Sw: Implementaz. - 20 [Giulio Destri - © AreaSP for Univ. Parma, 2006](#)

Check-list: livello classi

- Usare ereditarietà quando le classi sono collegate dal punto di vista del contenuto
- Riunire in sottosistemi le classi che collaborano strettamente
- Evitare di definire classi troppo generali

Ing. del Sw: Implementaz. - 21 [Giulio Destri - © AreaSP for Univ. Parma, 2006](#)

Check-list: livello metodi

- Usare identificatori di metodi e variabili ben dichiarati (usare maiuscole e minuscole)
- Tenere separati metodi pubblici e privati
- Ogni metodo deve essere verificabile
- I metodi devono essere facilmente comprensibili: mantenerli corti e suddividerli
- Conoscere gli algoritmi standard

Ing. del Sw: Implementaz. - 22 [Giulio Destri - © AreaSP for Univ. Parma, 2006](#)

Check-list: livello design

- Inizializzare tutte le variabili
- Verificare lo stato delle variabili
- Verificare la validità dei parametri di input e prevedere azioni apposite
- Verificare possibili errori nei valori di ritorno (es. null o fuori intervallo)

Ing. del Sw: Implementaz. - 23 [Giulio Destri - © AreaSP for Univ. Parma, 2006](#)

Sommario

- Definire il processo di Implementazione
- Caratteristiche importanti
- Linee guida
- Criteri di completezza

Ing. del Sw: Implementaz. - 24 [Giulio Destri - © AreaSP for Univ. Parma, 2006](#)