



Università degli Studi dell'Insubria
Dipartimento di Scienze Teoriche e Applicate

Laboratorio Interdisciplinare A

Documentazione di Progetto

Davide Tosi

Dipartimento di Scienze Teoriche e Applicate

davide.tosi@uninsubria.it



Documentazione di Progetto

- 1. Manuale Utente**
- 2. Manuale Tecnico**
- 3. JavaDoc**



Documentazione di Progetto

- **Manuale Utente**
 - Manuale di alto livello del funzionamento del programma
 - Come usare il programma (quale ambiente usare, come installare l'ambiente, come installare il programma, come eseguirlo, ...)
 - Aggiungere screenshot per semplificare la comprensione del funzionamento / uso del programma
 - Suddividere il manuale in due macro sezioni (una per l'area Centri Vaccinali e una per l'area Cittadini)
 - Utilizzare un livello di astrazione e un linguaggio adeguati ad un pubblico non esperto e non tecnico



Documentazione di Progetto

- Manuale Utente – Struttura
 - Frontespizio (Titolo, Autori, Data e Versione documento)
 - Indice
 - Installazione
 - Requisiti di sistema
 - Setup ambiente
 - Installazione programma
 - Esecuzione ed Uso
 - Setup e Lancio del programma
 - Uso delle Funzionalità
 - Data set di test
 - Limiti della soluzione sviluppata
 - Sitografia / Bibliografia
 - es. *QtSpim Simulator, Online: www.xyz.com*
 - es. *D. Tosi, "Come fare un progetto software", nome rivista, anno*



Documentazione di Progetto

- **Manuale Tecnico**
 - dettagliare dal punto di vista tecnico la struttura dell'applicazione
 - le scelte architetturali
 - le strutture dati utilizzate
 - le scelte algoritmiche
 - il formato dei file e la loro gestione (es. path)
 - l'uso eventuale di pattern, riportando parti di codice significativo
 - la JavaDoc è parte del manuale tecnico (il codice deve quindi essere commentato opportunamente)
 - Utilizzare un livello di astrazione e un linguaggio adeguati a esperti tecnici del dominio



Documentazione di Progetto

- Manuale Tecnico - Struttura
 - Si segua la struttura del Manuale Utente



Documentazione di Progetto

- JavaDoc ¹
 - è uno strumento che permette di documentare i file sorgenti di un programma usando le informazioni riportate al loro interno
 - il programmatore inserisce i commenti al codice in un formato predefinito
 - i commenti vengono riconosciuti dal programma javadoc che li converte in un formato di consultazione (html, pdf, ...)
 - un commento Javadoc è un testo HTML racchiuso tra i tag `/**` e `*/`
 - Javadoc è pensato solo per descrivere le funzionalità di Package, Classi, interfacce, Metodi (non può essere usato per commentare pezzi di codice)

¹ <https://www.oracle.com/technical-resources/articles/java/javadoc-tool.html>



Documentazione di Progetto

- JavaDoc (cont.)
 - la documentazione deve quindi comprendere la descrizione di ogni package, classe, interfaccia, metodo e attributo
 - Il commento è sempre posto subito prima della dichiarazione della classe, interfaccia, metodo, attributo
 - Può contenere tag HTML per aiutare la formattazione (es. tag di struttura come ``, ``, ecc)
- JavaDoc - Classi
 - deve descrivere lo scopo della classe
 - al termine della descrizione si mette il tag `@author Nome Cognome` per indicare l'autore del codice
 - Se ci sono più autori, si mettono più tag su righe separate, uno di seguito all'altro



Documentazione di Progetto

- JavaDoc – Attributi
 - deve descrivere a cosa serve l'attributo
 - solo il primo paragrafo verrà riportato nella descrizione generale. Gli altri andranno nella descrizione puntuale
 - un nuovo paragrafo inizia con il tag `<p>` a inizio linea
 - usare i tag `<code>` e `</code>` per racchiudere il nome degli attributi
 - vengono visualizzati nel font usato per il codice
- JavaDoc – Metodi
 - per ogni metodo usare sempre i tag a inizio linea:
 - `@param <nome parametro>` per fornire una breve descrizione del parametro (se più parametri, rispettare l'ordine con cui sono dichiarati)
 - `@return` per fornire una breve descrizione di ciò che il metodo ritorna
 - `@throws <nome eccezione>` per fornire una descrizione delle circostanze che determinano il sollevamento dell'eccezione



Documentazione di Progetto

- JavaDoc – Package
 - deve riportare la documentazione generale del package
 - usare i tag `@see` per riferirsi alle classi, metodi, attributi in esso contenuti
- Per generare la documentazione si usa il comando `javadoc` che fa parte della distribuzione di Java Development Kit (JDK)