



Università degli Studi di Padova

Laurea: Informatica

Corso: Ingegneria del Software

Anno Accademico: 2021/2022



Gruppo: MERL

Email: merlunipd@gmail.com

Manuale Sviluppatore

Informazioni sul documento

Versione	V0.0.2
Uso	Esterno
Data approvazione	xx/xx/2022
Distribuzione	Prof. <i>Vardanega Tullio</i> Prof. <i>Cardin Riccardo</i> <i>Zucchetti s.p.a.</i> Gruppo <i>MERL</i>

Registro delle Modifiche

Versione	Data	Autore	Verificatore	Modifica
v0.0.3	04/04/2022	Emanuele Pase	Marco Mamprin	Aggiunto il capitolo "Configurazione"
v0.0.2	03/04/2022	Riccardo Contin	Marco Mazzucato	Aggiunto il capitolo "Introduzione"
v0.0.1	02/04/2022	Marko Vukovic	Emanuele Pase	Aggiunto il capitolo "Tecnologie"
v0.0.0	02/04/2022	Marko Vukovic	Emanuele Pase	Creata prima struttura del documento

Indice

Elenco delle figure

Elenco delle tabelle

1. Introduzione

1.1 Scopo del documento

Lo scopo del *Manuale Sviluppatore* è quello di presentare l'architettura del prodotto, l'organizzazione del codice sorgente e in particolare fornire informazioni utili al mantenimento e all'estensione del progetto. Questo documento ha inoltre il fine di illustrare le procedure di installazione e di sviluppo in locale, citare i framework e le librerie di terze parti coinvolte e di presentare la struttura del progetto a livelli progressivi di dettaglio, grazie all'utilizzo di diagrammi di classe e di sequenza.

1.2 Scopo del prodotto

Il capitolato proposto dall'azienda *Zucchetti S.p.A* ha come obiettivo quello di creare un'applicazione di visualizzazione di dati di login con numerose dimensioni che permettano di rintracciare eventuali anomalie a colpo d'occhio. Lo scopo del prodotto è quindi quello di fornire all'utente diversi tipi di visualizzazione di dati in modo da rendere più veloce ed efficace l'individuazione di anomalie.

1.3 Glossario

Per evitare ambiguità relative alle terminologie utilizzate, queste verranno evidenziati da una 'G' a pedice e riportate nel glossario presente nell'appendice §A.

1.4 Riferimenti

1.4.1 Riferimenti normativi

- *Norme di Progetto V1.0.0*
- Capitolato d'appalto C5 - Zucchetti S.p.A.: Login Warrior
<https://www.math.unipd.it/~tullio/IS-1/2021/Progetto/C5.pdf>

1.4.2 Riferimenti informativi

- Slide T9 del corso di Ingegneria del Software - Progettazione
<https://www.math.unipd.it/~tullio/IS-1/2021/Dispense/T09.pdf>

- Slide P2 del corso di Ingegneria del Software - Diagramma delle classi
https://www.math.unipd.it/~rcardin/swea/2021/Diagrammi%20delle%20Classi_4x4.pdf
- Slide P3 del corso di Ingegneria del Software - Gestione delle dipendenze
<https://www.math.unipd.it/~rcardin/swea/2022/Dependency%20Management%20in%20Object-Oriented%20Programming.pdf>
- Slide P5 del corso di Ingegneria del Software - Diagramma di sequenza
<https://www.math.unipd.it/~rcardin/swea/2022/Diagrammi%20di%20Sequenza.pdf>
- Slide L02 del corso di Ingegneria del Software - Pattern MVC e derivati
<https://www.math.unipd.it/~rcardin/sweb/2022/L02.pdf>
- Slide L03 del corso di Ingegneria del Software - SOLID programming
<https://www.math.unipd.it/~rcardin/sweb/2022/L03.pdf>

2. Tecnologie

Tecnologia	Versione	Descrizione
Linguaggi		
HTML	5	Linguaggio di markup utilizzato per definire gli elementi dell'interfaccia.
CSS	3	Linguaggio utilizzato per la gestione dello stile degli elementi HTML.
Javascript	ES6	Linguaggio di programmazione ad alto livello, interpretato, multi-paradigma, con tipizzazione debole. Viene utilizzato dal motore del browser per eseguire codice da lato client. Utilizzati i <i>Moduli ES6</i> per gestire i file contenenti il codice Javascript.
Librerie		
D3	7.4.0	Libreria Javascript utilizzata per manipolare elementi del DOM in base a dati. Permette di creare visualizzazioni e grafici.
Strumenti		
NodeJS	17.2.0	Runtime costruito sul motore V8 di Google per l'esecuzione di codice JavaScript. Utilizzato per accedere a strumenti di supporto allo sviluppo (e.g. JestJS, ESLint) e per la definizione di piccoli script.
NPM	8.1.4	Package manager per la gestione di dipendenze di progetti NodeJS.
JestJS	27.5	Strumento per effettuare analisi dinamica di codice Javascript e per generare il code coverage.
ESLint	8.12	Strumento di analisi statica del codice. Viene utilizzato con le best practices configurate dallo standard <i>AirBnB</i> .

JSDocs	3.5.5	Linguaggio di markup che permette di annotare il codice sorgente Javascript e generare documentazione.
IndexedDB	3.0	API Javascript fornite dai browser per permettere il caching di dati da lato client.
Git	2.34.1	Strumento di controllo della versione distribuito. Utilizzato per gestire la repository remota su GitHub.

3. Configurazione

In questa sezione verranno affrontati i requisiti richiesti per l'utilizzo dell'applicazione.

3.1 Requisiti di sistema

3.1.1 Requisiti minimi

Per far funzionare l'applicazione non ci sono particolari richieste, trattandosi di una Single-page Application.

3.1.2 Requisiti consigliati

Per avere un'esperienza completa nell'uso dell'applicazione si consiglia d'installare nella propria macchina i seguenti software:

Software	Versione	Riferimenti per il download
Node.js	16.14.2	https://nodejs.org/en/
Npm	8.x	Integrato nel download di Node.js

Tabella 3.1: Tabella dei requisiti consigliati

3.2 Requisiti hardware

Al fine di garantire prestazioni accettabili si consiglia di soddisfare i seguenti requisiti hardware:

Componente	Versione
Processore	Quad-Core
RAM	8GB

Tabella 3.2: Tabella dei requisiti hardware

3.3 Browser

I browser testati e resi compatibili con l'applicazione sono:

Browser	Versione
Chrome	99
Edge	99
Firefox	98
Opera	83
Safari	15.2

Tabella 3.3: Tabella dei browser testati e supportati

4. Architettura

4.1 Introduzione

Marco

4.2 Diagrammi delle classi

4.2.1 Model

4.2.2 View

Marco

4.2.3 Controller

4.3 Diagrammi di sequenza

4.3.1 Caricamento dataset

4.3.2 Nuovo campionamento

Marco

4.4 Design pattern utilizzati

4.4.1 Strategy

4.4.2 Template method

Marco

5. **Estensione**

A. Glossario