

Università degli Studi di Padova

DIPARTIMENTO DI MATEMATICA "TULLIO LEVI-CIVITA"

CORSO DI LAUREA IN INFORMATICA



**Modifiche evolutive e ristrutturazione di una  
WebApp come studio sull' architettura a  
microservizi**

*Tesi di laurea*

*Relatore*

Prof. Francesco Ranzato

*Laureando*

Rolando Sartorello

---

ANNO ACCADEMICO 2021-2022



# Sommario

Il presente documento descrive il lavoro svolto durante il periodo di stage, della durata di circa trecentoventi ore, dal laureando presso l'azienda azienda Sync Lab.

Lo scopo di questo stage è stato quello di studiare ed implementare le componenti di back end per sviluppare alcune modifiche evolutive sulla web application prototipale *Challenginator*, strutturandola architetturalmente secondo il pattern dei microservizi ed integrando funzionalità ausiliarie, per poi integrare le novità anche da lato frontend, avendo sempre l'obiettivo primario di seguire le corrette pratiche architetturali.



# Ringraziamenti

*Ringrazio chiunque.  
Del mondo accademico o personale,  
ringrazio chiunque.*

*Padova, Giugno 2022*

Rolando Sartorello



# Indice

<b>1</b>	<b>Introduzione</b>	<b>1</b>
1.1	Scopo del documento . . . . .	1
1.2	L'azienda . . . . .	1
1.3	Struttura del testo . . . . .	1
1.4	Convenzioni tipografiche . . . . .	2
<b>2</b>	<b>Descrizione dello stage</b>	<b>3</b>
2.1	Introduzione al progetto . . . . .	3
2.2	Requisiti e obiettivi . . . . .	3
2.3	Pianificazione . . . . .	3
<b>3</b>	<b>Analisi del progetto</b>	<b>5</b>
3.1	Analisi preventiva dei rischi . . . . .	5
3.2	Casi d'uso . . . . .	5
3.3	Tracciamento dei requisiti . . . . .	6
<b>4</b>	<b>Progettazione e codifica</b>	<b>9</b>
4.1	Tecnologie e strumenti . . . . .	9
4.2	Ciclo di vita del software . . . . .	9
4.3	Progettazione . . . . .	9
4.4	Design Pattern utilizzati . . . . .	9
4.5	Codifica . . . . .	9
<b>5</b>	<b>Verifica e validazione</b>	<b>11</b>
<b>6</b>	<b>Conclusioni</b>	<b>13</b>
6.1	Consuntivo finale . . . . .	13
6.2	Raggiungimento degli obiettivi . . . . .	13
6.3	Conoscenze acquisite . . . . .	13
6.4	Valutazione personale . . . . .	13
<b>A</b>	<b>Appendice A</b>	<b>15</b>
	<b>Glossario</b>	<b>17</b>
	<b>Acronimi</b>	<b>19</b>
	<b>Bibliografia</b>	<b>21</b>

## Elenco delle figure

3.1 Use Case - UC0: Scenario principale . . . . .	6
---	---

## Elenco delle tabelle

3.1 Tabella del tracciamento dei requisiti funzionali . . . . .	7
3.2 Tabella del tracciamento dei requisiti qualitativi . . . . .	7
3.3 Tabella del tracciamento dei requisiti di vincolo . . . . .	7



# Capitolo 1

## Introduzione

### 1.1 Scopo del documento

Questo documento è la relazione conclusiva del tirocinio frequentato dallo studente in conclusione al piano di studio.

Si cerca qui di spiegare il lavoro svolto mostrandone i risultati, il percorso, e le relative scelte tecnologiche/architetturali, oltre che le competenze coinvolte per tale raggiungimento.

### 1.2 L'azienda

Sync Lab è un'azienda che realizza prodotti e soluzioni per diversi ambiti come sanità, industria, energia, telecomunicazioni, finanza, trasporti e logistica. Si è specializzata in campi come GDPR, Big Data, Cloud Computing, IoT, Mobile e Cyber Security. Un'azienda con una storia di vent'anni e sedi a Napoli, Roma, Padova, Verona, Milano, Como; un organico di più di 300 persone, un fatturato di circa 13 milioni di euro e numerose certificazioni ISO in campi di qualità, gestione dati, sicurezza, gestione ambientale. Tra i suoi più di 150 clienti troviamo alcuni come TIM, Vodafone, Intesa San Paolo, Enel e Trentalia.

### 1.3 Struttura del testo

## TODO

[Il secondo capitolo](#) approfondisce ...

[Il terzo capitolo](#) approfondisce ...

[Il quarto capitolo](#) approfondisce ...

[Il quinto capitolo](#) approfondisce ...

[Nel sesto capitolo](#) descrive ...

## 1.4 Convenzioni tipografiche

Riguardo la stesura del testo, relativamente al documento sono state adottate le seguenti convenzioni tipografiche:

- \* Acronimi, le abbreviazioni e i termini ambigui o di uso non comune menzionati vengono definiti nel glossario, situato alla fine di questo documento;
- \* La prima occorrenza dei termini riportati nel glossario viene denotata da una 'g' in apice alla parola: [UML<sup>g</sup>](#);
- \* Termini in lingua straniera o facenti parti del gergo tecnico sono evidenziati con il carattere *corsivo*.
- \* Snippet di codice, nomi di file e simili verranno indicati con il **carattere monospace**;

## Capitolo 2

# Descrizione dello stage

*Breve introduzione al capitolo*

**2.1** Introduzione al progetto

**2.2** Requisiti e obiettivi

**2.3** Pianificazione



## Capitolo 3

# Analisi del progetto

*Breve introduzione al capitolo*

### 3.1 Analisi preventiva dei rischi

Durante la fase di analisi iniziale sono stati individuati alcuni possibili rischi a cui si potrà andare incontro. Si è quindi proceduto a elaborare delle possibili soluzioni per far fronte a tali rischi.

#### 1. Performance del simulatore hardware

**Descrizione:** le performance del simulatore hardware e la comunicazione con questo potrebbero risultare lenti o non abbastanza buoni da causare il fallimento dei test.

**Soluzione:** coinvolgimento del responsabile a capo del progetto relativo il simulatore hardware.

### 3.2 Casi d'uso

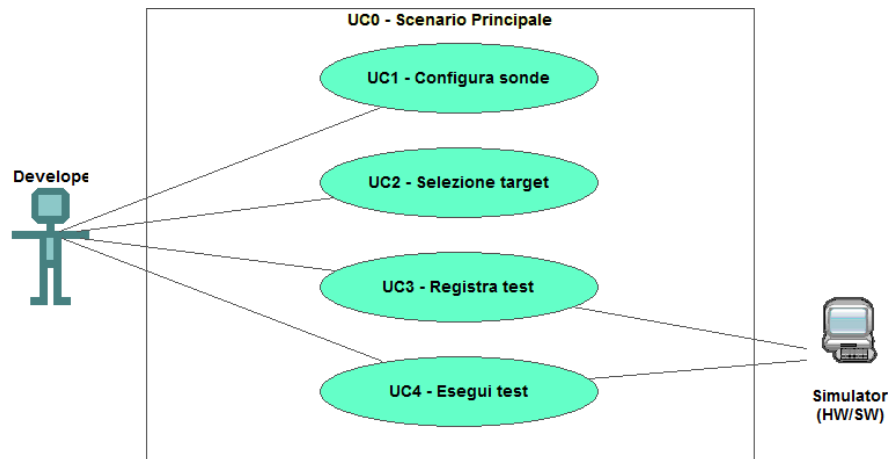
Per lo studio dei casi di utilizzo del prodotto sono stati creati dei diagrammi. I diagrammi dei casi d'uso (in inglese *Use Case Diagram*) sono diagrammi di tipo [Unified Modeling Language \(UML\)](#) dedicati alla descrizione delle funzioni o servizi offerti da un sistema, così come sono percepiti e utilizzati dagli attori che interagiscono col sistema stesso. Essendo il progetto finalizzato alla creazione di un tool per l'automazione di un processo, le interazioni da parte dell'utilizzatore devono essere ovviamente ridotte allo stretto necessario. Per questo motivo i diagrammi d'uso risultano semplici e in numero ridotto.

#### UC0: Scenario principale

**Attori Principali:** Sviluppatore applicativi.

**Precondizioni:** Lo sviluppatore è entrato nel plug-in di simulazione all'interno dell'IDE.

**Descrizione:** La finestra di simulazione mette a disposizione i comandi per configurare, registrare o eseguire un test.



**Figura 3.1:** Use Case - UC0: Scenario principale

**Postcondizioni:** Il sistema è pronto per permettere una nuova interazione.

### 3.3 Tracciamento dei requisiti

Da un'attenta analisi dei requisiti e degli use case effettuata sul progetto è stata stilata la tabella che traccia i requisiti in rapporto agli use case.

Sono stati individuati diversi tipi di requisiti e si è quindi fatto utilizzo di un codice identificativo per distinguerli.

Il codice dei requisiti è così strutturato R(F/Q/V)(N/D/O) dove:

R = requisito

F = funzionale

Q = qualitativo

V = di vincolo

N = obbligatorio (necessario)

D = desiderabile

Z = opzionale

Nelle tabelle 3.1, 3.2 e 3.3 sono riassunti i requisiti e il loro tracciamento con gli use case delineati in fase di analisi.

**Tabella 3.1:** Tabella del tracciamento dei requisiti funzionali

Requisito	Descrizione	Use Case
RFN-1	L'interfaccia permette di configurare il tipo di sonde del test	UC1

**Tabella 3.2:** Tabella del tracciamento dei requisiti qualitativi

Requisito	Descrizione	Use Case
RQD-1	Le prestazioni del simulatore hardware deve garantire la giusta esecuzione dei test e non la generazione di falsi negativi	-

**Tabella 3.3:** Tabella del tracciamento dei requisiti di vincolo

Requisito	Descrizione	Use Case
RVO-1	La libreria per l'esecuzione dei test automatici deve essere riutilizzabile	-





# Capitolo 4

## Progettazione e codifica

*Breve introduzione al capitolo*

### 4.1 Tecnologie e strumenti

Di seguito viene data una panoramica delle tecnologie e strumenti utilizzati.

#### **Tecnologia 1**

Descrizione Tecnologia 1.

#### **Tecnologia 2**

Descrizione Tecnologia 2

### 4.2 Ciclo di vita del software

### 4.3 Progettazione

#### **Namespace 1**

Descrizione namespace 1.

**Classe 1:** Descrizione classe 1

**Classe 2:** Descrizione classe 2

### 4.4 Design Pattern utilizzati

### 4.5 Codifica



## Capitolo 5

# Verifica e validazione



## Capitolo 6

# Conclusioni

6.1 Consuntivo finale

6.2 Raggiungimento degli obiettivi

6.3 Conoscenze acquisite

6.4 Valutazione personale



Appendice A

Appendice A

Citazione

---

Autore della citazione





# Glossario

**UML** in ingegneria del software *UML, Unified Modeling Language* (ing. linguaggio di modellazione unificato) è un linguaggio di modellazione e specifica basato sul paradigma object-oriented. L'*UML* svolge un'importantissima funzione di “lingua franca” nella comunità della progettazione e programmazione a oggetti. Gran parte della letteratura di settore usa tale linguaggio per descrivere soluzioni analitiche e progettuali in modo sintetico e comprensibile a un vasto pubblico. [2](#), [19](#)



# Acronimi

**UML** [Unified Modeling Language](#). 5, 17



# Bibliografia

## Riferimenti bibliografici

James P. Womack, Daniel T. Jones. *Lean Thinking, Second Editon*. Simon & Schuster, Inc., 2010.

## Siti web consultati

*Manifesto Agile*. URL: <http://agilemanifesto.org/iso/it/>.