

Università degli Studi di Padova

DIPARTIMENTO DI MATEMATICA “TULLIO LEVI-CIVITA”

CORSO DI LAUREA IN INFORMATICA



**Developer Portal Hub: una soluzione
centralizzata per la gestione sicura ed
intuitiva della documentazione API**

Tesi di laurea

Relatore

Prof.ssa Ombretta Gaggi

Laureando

Andrea Meneghello 2009084

ANNO ACCADEMICO 2022-2023

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

— Oscar Wilde

Dedicato a ...

Sommario

Il seguente documento ha lo scopo di descrivere in modo dettagliato il lavoro svolto durante il periodo di stage, dal laureando Andrea Meneghello, della durata di trecentododici ore, presso l'azienda THRON S.p.A. L'obiettivo principale del progetto di stage è stato realizzare un portale per favorire la consultazione delle API pubbliche e private di THRON, attraverso una soluzione centralizzata.

Il portale è stato sviluppato utilizzando il framework Vue.js accompagnato da vari strumenti del suo ecosistema, ed è stato protetto con autenticazione seguendo lo standard OAuth2, integrandosi con il provider aziendale Azure. Oltre alla consultazione della documentazione, il portale deve permettere all'utente di provare le API direttamente dall'interfaccia in modo intuitivo, permettendo inoltre il download delle stesse in formato yaml.

Infine tutte le componenti implementate sono state opportunamente documentate e il loro corretto funzionamento è stato verificato tramite test di unità e di accettazione.

“Life is really simple, but we insist on making it complicated”

— Confucius

Ringraziamenti

Innanzitutto, vorrei esprimere la mia gratitudine al Prof. Ombretta Gaggi, relatore della mia tesi, per l'aiuto e il sostegno fornitomi durante la stesura del lavoro.

Desidero ringraziare con affetto la mia famiglia per il sostegno, il grande aiuto e per essermi stati vicini in ogni momento durante gli anni di studio.

Ho desiderio di ringraziare poi i miei amici per tutti i bellissimi anni passati insieme e le mille avventure vissute.

Padova, Settembre 2023

Andrea Meneghello

Indice

1	Introduzione	1
1.1	L'azienda	1
1.2	Metodologie di sviluppo	2
1.3	Strumenti di sviluppo	2
1.4	Organizzazione del testo	2
2	Descrizione del progetto di stage	3
2.1	Introduzione al progetto	3
2.2	Obiettivi dello stage	3
2.3	Analisi preventiva dei rischi	3
3	Analisi dei requisiti	4
3.1	Descrizione dell'applicazione	4
3.2	Casi d'uso	4
3.3	Tracciamento dei requisiti	5
4	Progettazione e codifica	6
4.1	Struttura principale del sistema	6
4.2	Tecnologie utilizzate	6
4.3	Progettazione	6
4.4	Codifica	6
5	Attività di verifica e validazione	7
5.1	Verifica	7
5.2	Documentazione	7
5.3	Test di unità	7
5.4	Collaudo	7
5.5	Migliorie future	7
5.6	Presentazione finale	7
6	Conclusioni	8
6.1	Obiettivi raggiunti e consuntivo finale	8
6.2	Conoscenze acquisite	8
6.3	Scenari di applicabilità e sviluppi futuri	8
6.4	Valutazione personale	8
A	Appendice A	9
	Acronimi e abbreviazioni	10

INDICE

vi

Glossario

11

Bibliografia

12

Elenco delle figure

3.1 Use Case - UC0: Scenario principale	4
---	---

Elenco delle tabelle

Capitolo 1

Introduzione

Il seguente capitolo vuole introdurre brevemente l'azienda e il relativo contesto aziendale.

1.1 L'azienda

THRON S.p.A. è un'azienda italiana con sede a Piazzola sul Brenta specializzata nello sviluppo di SaaS (Software as a Service) e opera nel settore dei DAM (Digital Asset Management), offrendo servizi di marketing e business intelligence.

Il suo prodotto principale è la "Thron DAM Platform", una piattaforma per la gestione dei contenuti digitali, nata con l'obiettivo di valorizzare e gestire le informazioni sui prodotti in modo separato dalla piattaforma di distribuzione finale.

La Thron DAM Platform è una soluzione completa per la gestione dei contenuti aziendali, inclusi documenti, video, immagini, audio e molto altro. L'obiettivo è garantire una gestione centralizzata dei contenuti e semplificarne l'adattamento e la distribuzione su diversi canali in modo efficiente. La piattaforma è supportata da un motore semantico che consente l'arricchimento dei contenuti con tag e metadati, facilitando l'organizzazione e la categorizzazione dei materiali.

THRON S.p.A. ha strutturato l'area RD in due principali team: il team Contenuti, focalizzato sulle tematiche legate al DAM e alle sue funzionalità, e il team Prodotto, responsabile della gestione del PIM (Product Information Management) e delle funzionalità legate alle tag sui prodotti. Il metodo di lavoro adottato è il framework agile SCRUM, che coinvolge attivamente gli stakeholder nel processo decisionale, incoraggiando suggerimenti e miglioramenti.

L'azienda mira a una vasta ed eterogenea clientela, per cui offre un prodotto flessibile e adattabile alle specifiche esigenze dei clienti. Inizialmente, la piattaforma viene venduta con un modulo base che offre le funzionalità standard, ma attraverso un marketplace, è possibile acquistare o aggiungere moduli gratuiti che ampliano e migliorano la piattaforma.

La cultura di prossimità con gli stakeholder e il contatto costante con diverse esigenze dei clienti portano THRON a un continuo processo di innovazione per soddisfare le richieste e rimanere all'avanguardia nel settore.

Attualmente, THRON conta circa cinquanta dipendenti, suddivisi in team in base alle rispettive competenze. La collaborazione tra i dipartimenti e la capacità di adattarsi alle esigenze del mercato e dei clienti sono elementi fondamentali nel successo dell'azienda. Un'ulteriore peculiarità dell'azienda è la possibilità di offrire soluzioni

personalizzate per specifici casi d'uso dei clienti, garantendo un servizio su misura per le loro esigenze.

La continua evoluzione e innovazione della Thron DAM Platform, insieme all'impegno costante nel fornire soluzioni di alto livello, consolidano la posizione di THRON S.p.A. nel mercato del Digital Asset Management, portando il "made in Italy" in un ambito altamente competitivo e in continua crescita.

1.2 Metodologie di sviluppo

Introduzione all'idea dello stage.

1.3 Strumenti di sviluppo

1.4 Organizzazione del testo

Il secondo capitolo descrive

Il terzo capitolo approfondisce ...

Il quarto capitolo approfondisce ...

Il quinto capitolo approfondisce ...

Il sesto capitolo approfondisce ...

Nel settimo capitolo descrive ...

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

- gli acronimi, le abbreviazioni e i termini ambigui o di uso non comune menzionati vengono definiti nel glossario, situato alla fine del presente documento;
- per la prima occorrenza dei termini riportati nel glossario viene utilizzata la seguente nomenclatura: *parola*^[g];
- i termini in lingua straniera o facenti parti del gergo tecnico sono evidenziati con il carattere *corsivo*.

Capitolo 2

Descrizione del progetto di stage

Il seguente capitolo vuole introdurre brevemente il progetto affrontato durante lo stage

2.1 Introduzione al progetto

2.2 Obiettivi dello stage

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.

2.3 Analisi preventiva dei rischi

Capitolo 3

Analisi dei requisiti

Breve introduzione al capitolo

3.1 Descrizione dell'applicazione

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.

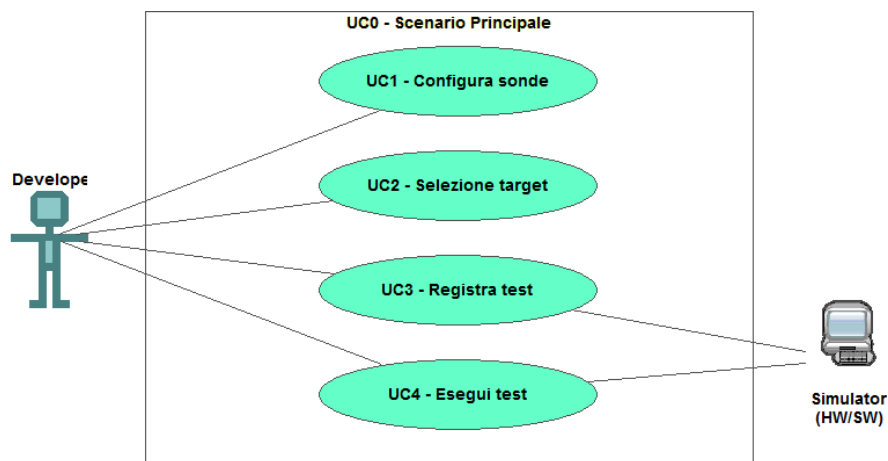


Figura 3.1: Use Case - UC0: Scenario principale

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.

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.

Capitolo 4

Progettazione e codifica

Breve introduzione al capitolo

4.1 Struttura principale del sistema

4.2 Tecnologie utilizzate

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

Tecnologia 1

Descrizione Tecnologia 1.

Tecnologia 2

Descrizione Tecnologia 2

4.3 Progettazione

Namespace 1

Descrizione namespace 1.

Classe 1: Descrizione classe 1

Classe 2: Descrizione classe 2

4.4 Codifica

Codifica back-end

Codifica front-end

Capitolo 5

Attività di verifica e validazione

5.1 Verifica

5.2 Documentazione

5.3 Test di unità

5.4 Collaudo

5.5 Migliorie future

5.6 Presentazione finale

Capitolo 6

Conclusioni

- 6.1 Obiettivi raggiunti e consuntivo finale
- 6.2 Conoscenze acquisite
- 6.3 Scenari di applicabilità e sviluppi futuri
- 6.4 Valutazione personale

Appendice A

Appendice A

Citazione

Autore della citazione

Acronimi e abbreviazioni

UML [Unified Modeling Language](#). 4, 11

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. [10](#)

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/>.