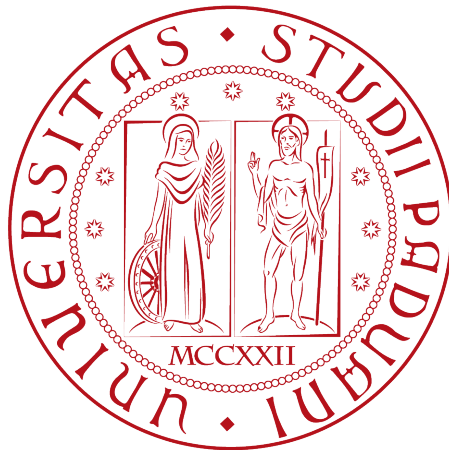


Università degli Studi di Padova

DIPARTIMENTO DI MATEMATICA "TULLIO LEVI-CIVITA "

CORSO DI LAUREA IN INFORMATICA



Titolo della tesi

Tesi di laurea

Relatore

Prof. Paolo Baldan

Laureando

Mattia Episcopo

ANNO ACCADEMICO 2022-2023

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

— Oscar Wilde

Dedicato a ...

Sommario

Il presente documento descrive il lavoro svolto durante il periodo di stage, della durata di circa trecento ore, dal laureando Pinco Pallino presso l'azienda Azienda S.p.A. Gli obbiettivi da raggiungere erano molteplici.

In primo luogo era richiesto lo sviluppo di ... In secondo luogo era richiesta l'implementazione di un ... Tale framework permette di registrare gli eventi di un controllore programmabile, quali segnali applicati Terzo ed ultimo obbiettivo era l'integrazione ...

"I'm not crazy, my mother had me tested"

— Dott. Sheldon Lee Cooper

Ringraziamenti

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

Desidero ringraziare con affetto i miei genitori 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, Febbraio 2023

Mattia Episcopo

Indice

1	Introduzione	1
1.1	L'azienda	1
1.2	L'idea	1
1.3	Organizzazione del testo	2
2	Processi e metodologie	3
2.1	Processo sviluppo prodotto	3
3	Descrizione dello stage	5
3.1	Introduzione al progetto	5
3.2	Analisi preventiva dei rischi	5
3.3	Requisiti e obiettivi	6
3.4	Pianificazione	6
4	Analisi dei requisiti	9
4.1	Casi d'uso	9
4.2	Tracciamento dei requisiti	11
5	Progettazione e codifica	15
5.1	Tecnologie e strumenti	15
5.2	Ciclo di vita del software	17
5.3	Progettazione	17
5.4	Design Pattern utilizzati	17
5.5	Codifica	17
6	Verifica e validazione	19
7	Conclusioni	21
7.1	Consuntivo finale	21
7.2	Raggiungimento degli obiettivi	21
7.3	Conoscenze acquisite	21
7.4	Valutazione personale	21
A	Appendice A	23
	Bibliografia	27

Elenco delle figure

4.1	Use Case - UC0: Scenario principale	9
-----	---	---

Elenco delle tabelle

3.1	Tabella dei Requisiti	6
4.1	Tabella del tracciamento dei requisiti funzionali	13
4.2	Tabella del tracciamento dei requisiti qualitativi	13
4.3	Tabella del tracciamento dei requisiti di vincolo	13

Capitolo 1

Introduzione

Introduzione al contesto applicativo.

Esempio di utilizzo di un termine nel glossario
[Application Program Interface \(API\)](#).

Esempio di citazione in linea
site:agile-manifesto.

Esempio di citazione nel pie' di pagina
citazione¹

1.1 L'azienda

Synthema Artificial Intelligence (S.AI) è una start-up innovativa che si occupa di ricerca, progettazione, sviluppo, commercializzazione e manutenzione di prodotti e servizi innovativi ad alto valore tecnologico, basati sull'Internet of Things, la blockchain e su tecniche di Intelligenza Artificiale (in particolare reti neurali profonde), per l'analisi integrata e la comprensione di dati multimodali da fonti eterogenee (linguaggio naturale sia scritto che parlato, audio, immagini, video, dati generati da sensori) e per la gestione dei workflow, sia nel settore pubblico che privato.

1.2 L'idea

Si è resa necessaria per l'azienda, la costruzione di un'applicazione per il caricamento delle registrazioni audio, che vengono effettuate nelle aule di tribunale, verso un portale già esistente. Le registrazioni che vengono caricate, dispongono di vari metadati che possono essere usati per rendere il caricamento delle registrazioni stesse più veloce e semplice possibile.

¹womak:lean-thinking.

1.3 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

Processi e metodologie

Brevissima introduzione al capitolo

2.1 Processo sviluppo prodotto

Capitolo 3

Descrizione dello stage

Breve introduzione al capitolo

In questo capitolo viene descritto come si è pianificato e svolto lo stage presso SAI, introducendo il progetto, considerando gli obiettivi e i possibili rischi.

3.1 Introduzione al progetto

Lo stage svolto presso SAI aveva come scopo la creazione di un prototipo di piattaforma web/desktop per il caricamento delle registrazioni audio effettuate nelle aule di tribunale, con l'aiuto di alcuni metadata prodotti durante le medesime registrazioni. In particolare l'applicazione da sviluppare è divisa in due parti backend e frontend (spiegare meglio back e front) che comunicano tra loro mediante API rest (spiegare API). Lo stage si è focalizzato sullo sviluppo della parte frontend, avanzando di pari passo con lo sviluppo del backend realizzata da altri membri del team di sviluppo. Per la realizzazione del frontend si è scelto di utilizzare Javascript con i framework React e Redux.

3.2 Analisi preventiva dei rischi

Nella fase di analisi iniziali si sono individuati i principali rischi a cui si poteva andare incontro e si è proceduto a definire le possibili soluzioni per farne fronte.

1. Uso di nuove tecnologie

Descrizione: le tecnologie proposte per la gestione e lo sviluppo del progetto erano per lo più nuove o scarsamente conosciute..

Soluzione: è stato previsto un periodo iniziale di studio e formazione su queste tecnologie..

2. Modalità di lavoro smart working

Descrizione: lo stage è stato fatto completamente da remoto, e poteva portare ad una possibile mancanza di comunicazione e ad un'incertezza nelle attività da svolgere..

Soluzione: il tutor aziendale si è reso disponibile a vari meeting nelle prime fasi del progetto e nella formazione iniziale, e rimanendo a disposizione per altri meeting in caso di dubbi sulle attività da svolgere..

O01	autenticazione mediante server remoto
O02	lettura dati da CD
O03	precompilazione di form con i dati caricati da CD
O04	editing dei dati del form
D01	upload dei dati verso i sistemi esterni
D02	test di unità esaustivi
F01	possibilità di ascoltare le registrazioni
F02	possibilità di modificare i dati mediante interazioni evolute (per es. drag-n-drop)
F03	realizzazione di un'applicazione desktop con Electron
F04	compilazione multiplatforma dell'applicazione desktop

Tabella 3.1: Tabella dei Requisiti

3.3 Requisiti e obiettivi

Si farà riferimento ai requisiti secondo le seguenti notazioni:

- * O per i requisiti obbligatori, vincolanti in quanto obiettivo primario richiesto dal committente;
- * D per i requisiti desiderabili, non vincolanti o strettamente necessari, ma dal riconoscibile valore aggiunto;
- * F per i requisiti facoltativi, rappresentanti valore aggiunto non strettamente competitivo. Le sigle precedentemente indicate saranno seguite da una coppia sequenziale di numeri, identificativo del requisito.

3.4 Pianificazione

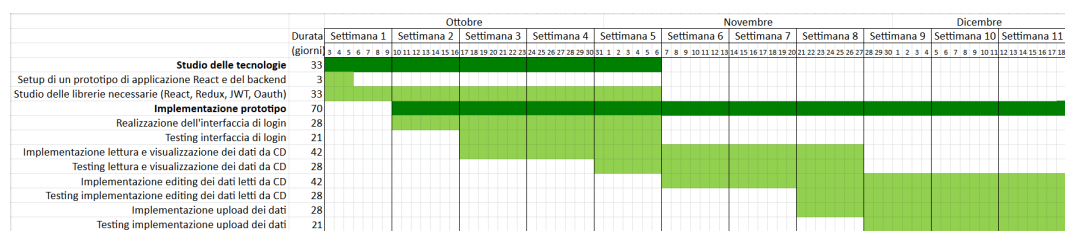
La pianificazione, in termini di quantità di ore, sarà distribuita in attività di studio e attività implementative.

Lo stage è stato strutturato in due fasi principali, la prima dedicata alle attività di studio e la seconda alle attività implementative. Di seguito vengono elencate le attività pianificate con la relativa quantità di ore stimata per ogni attività.

- * Studio delle tecnologie (80 ore)
 - Setup di un prototipo di applicazione React e del backend (16 ore);

- Studio delle librerie necessarie (React, Redux, JWT, OAuth) (64 ore);
- * Implementazione prototipo (240 ore)
 - Realizzazione dell'interfaccia di login (30 ore);
 - Implementazione lettura e visualizzazione dei dati da CD (70 ore);
 - Implementazione editing dei dati letti da CD (70 ore);
 - Implementazione upload dei dati (30 ore);
 - Testing dei prodotti realizzati (40 ore);

Per ogni attività riguardante la fase di implementazione del prototipo è stata prevista anche la relativa attività di testing. Il tutto viene rappresentato dal seguente diagramma di Gantt.



Capitolo 4

Analisi dei requisiti

Breve introduzione al capitolo

4.1 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.

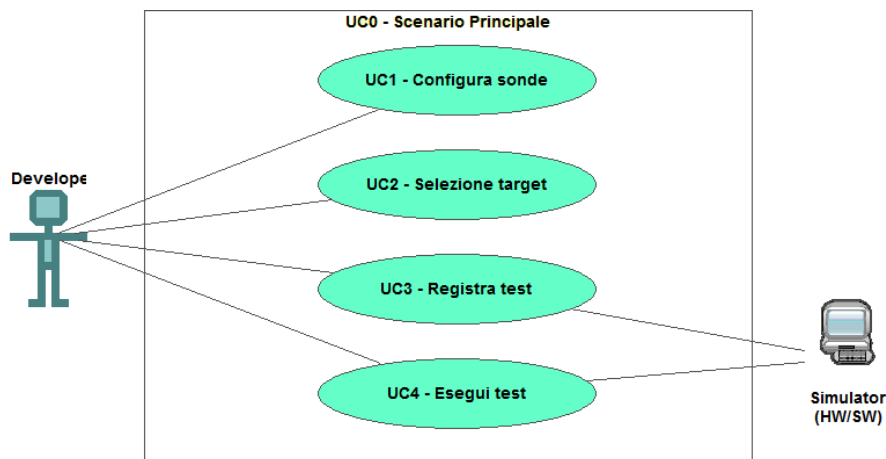


Figura 4.1: Use Case - UC0: Scenario principale

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.

UC1: Autenticazione

Attori Principali: Utente non autorizzato.

Precondizioni: L'utente non si è ancora autenticato.

Descrizione: Permette l'autenticazione di un utente.

Postcondizioni: L'utente è autenticato.

Scenario Principale: prova.

Scenario Alternativo: dswervr vvrdd.

UC1.1: Inserimento Username

Attori Principali: Utente non autorizzato.

Precondizioni: L'utente è sulla pagina di autenticazione.

Descrizione: Inserimento della username.

Postcondizioni: L'utente ha inserito lo username.

UC1.2: Inserimento Password

Attori Principali: Utente non autorizzato.

Precondizioni: L'utente è sulla pagina di autenticazione.

Descrizione: Inserimento della password.

Postcondizioni: L'utente ha inserito la password.

UC2: Lettura Dati da CD

Attori Principali: Utente autorizzato.

Precondizioni: L'utente è nella pagina per il caricamento dati.

Descrizione: Permette all'utente di caricare dei dati da CD selezionati.

Postcondizioni: L'utente visualizza i dati presenti nel CD.

UC3: Visualizzazione Dati ordinati per file

Attori Principali: Utente autorizzato.

Precondizioni: L'utente si trova all'interno dell'applicazione.

Descrizione: Permette la visualizzazione dei dati caricati da CD.

Postcondizioni: L'utente può visualizzare i dati caricati da CD ordinati per file.

UC4: Visualizzazione Dati ordinati per Procedimenti

Attori Principali: Utente autorizzato.

Precondizioni: L'utente si trova all'interno dell'applicazione.

Descrizione: Permette la visualizzazione dei dati caricati da CD.

Postcondizioni: L'utente può visualizzare i dati caricati da CD ordinati per procedimenti.

UC5: Upload Dati

Attori Principali: Utente autorizzato.

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.

UC6: Visualizzazione Jobs

Attori Principali: Utente autorizzato.

Precondizioni: L'utente si trova all'interno dell'applicazione.

Descrizione: Permette la visualizzazione dei jobs all'interno dell'applicazione.

Postcondizioni: L'utente può visualizzare i jobs.

UC7: Visualizzazione singolo Job

Attori Principali: Utente autorizzato.

Precondizioni: L'utente si trova sulla pagina di visualizzazione dei jobs.

Descrizione: Permette di visualizzare i dettagli del singolo job.

Postcondizioni: L'utente può visualizzare la pagina del singolo job.

4.2 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 [4.1](#), [4.2](#) e [4.3](#) sono riassunti i requisiti e il loro tracciamento con gli use case delineati in fase di analisi.

Tabella 4.1: 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 4.2: 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 4.3: 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 5

Progettazione e codifica

Breve introduzione al capitolo

5.1 Tecnologie e strumenti

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

Javascript

JavaScript è un linguaggio di programmazione orientato agli oggetti e agli eventi, comunemente utilizzato nella programmazione Web lato client per la creazione applicazioni web. L'intera applicazione è stata scritta con questo linguaggio.

React

React è una libreria Javascript utilizzata per implementare interfacce utente (UI) lato frontend. React si basa sul concetto di component, idealmente è una libreria che permette di costruire i propri component come fossero degli elementi HTML del DOM per poi poterli riutilizzare nell'intera applicazione.

Redux & Redux RTK

Redux è un contenitore dello stato per le applicazioni Javascript. Viene usato per la gestione centralizzata dello stato delle applicazioni sviluppate in React Javascript. In particolare con la sua libreria Redux-Toolkit, permette una gestione dello stato semplice ed efficiente.

API rest

Descrizione Tecnologia 2

JSON

Markup usato per la pubblicazione dei dati dell'API.

MUI

Material UI è una libreria React open-source che permette di implementare i Google's Material Design. Essa comprende una collezione di componenti React precostruiti che possono essere facilmente adattati e messi in uso nella UI dell'applicazione.

React Router

React Router è la libreria standard per il routing in React. Questa libreria permette la navigazione tra le varie viste dell'applicazione, permette di gestire le URL, e mantenere la sincronizzazione tra URL e viste.

slack

È uno strumento di collaborazione aziendale, che facilita la gestione delle comunicazioni all'interno del gruppo di lavoro.

Git

Descrizione Tecnologia 2

Gitlab

Descrizione Tecnologia 2

Jira

Jira è uno strumento di gestione del lavoro

Docker

Docker è un sistema che permette di facilitare il deployment delle applicazioni.

VS Code

IDE usato per lo sviluppo del progetto.

Jitsi

Descrizione Tecnologia 2

Scrum

(???) Descrizione Tecnologia 2

airbnb js guidelines

(???) Descrizione Tecnologia 2

eslint

(???)

5.2 Ciclo di vita del software

5.3 Progettazione

Namespace 1

Descrizione namespace 1.

Classe 1: Descrizione classe 1

Classe 2: Descrizione classe 2

5.4 Design Pattern utilizzati

5.5 Codifica

Capitolo 6

Verifica e validazione

Capitolo 7

Conclusioni

7.1 Consuntivo finale

7.2 Raggiungimento degli obiettivi

7.3 Conoscenze acquisite

7.4 Valutazione personale

Appendice A

Appendice A

Citazione

Autore della citazione

Bibliografia