

Università degli Studi di Padova

DIPARTIMENTO DI MATEMATICA "TULLIO LEVI-CIVITA"

CORSO DI LAUREA IN INFORMATICA



Titolo della tesi

Tesi di laurea

Relatore

Prof. Francesco Ranzato

Laureando

Luciano Wu

ANNO ACCADEMICO 2021-2022

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 trecentoventi ore, dal laureando Luciano Wu presso l'azienda Azienda SyncLab S.p.A. Gli obbiettivi da raggiungere erano molteplici.

In primo luogo era richiesto lo sviluppo di una web-app per la gestione delle ordinazione dei piatti di un ristorante sushi all-you-can-eat. Tale applicazione permette di registrare gli ordinazioni e tenere in traccia i piatti in arrivo. Le componenti fondamentali richiesti sono gestione del tavolo, gestione delle ordinazioni e infine gestione dell'accesso. Terzo ed ultimo obbiettivo era l'integrazione delle varie maschere. La valutazione dello stage è certamente positivo: tutte le interfacce sono state implementate con successo.

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

— Confucius

Ringraziamenti

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

Ringrazio Fabio Pallaro e tutta SyncLab, per avermi dato la possibilità di svolgere lo stage e il progetto per misurare le mie capacità e imparare delle nuove.

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, in modo particolare Diego e Samuele, per tutti i bellissimi anni passati insieme e le mille avventure vissute.

Padova, Luglio 2022

Luciano Wu

Indice

1	Introduzione	1
1.1	SyncLab	1
1.1.1	Prodotti	2
1.2	Organizzazione del testo	4
2	Il progetto di stage	5
2.1	L'idea	5
2.2	La soluzione individuata	5
2.3	obbiettivi richiesti	6
2.3.1	Notazione	6
2.3.2	obiettivi fissati	6
2.3.3	Pianificazione del lavoro	7
2.4	Modalità di lavoro	8
2.5	Strumenti di comunicazione	8
2.5.1	Google Meet:	10
3	Analisi dei requisiti	13
3.1	Descrizione generale	13
3.1.1	interfacce della web-app	13
3.1.2	Caratteristiche degli Utenti	13
3.1.3	Tecnologie utilizzate	14
3.2	Casi D'uso	15
3.2.1	Introduzione	15
3.2.2	Attori primari	15
3.2.3	Utente generico	15
3.2.4	UC1 - Visualizza menù	15
3.2.5	UC1.1 - Visualizza categorie	15
3.2.6	UC1.2 - Visualizza piatto	16
3.2.7	UC1.3 - Aumenta quantità	16
3.2.8	UC1.4 - Diminuisce quantità	16
3.2.9	UC1.5 - Visualizza dettaglio piatto	17
3.2.10	UC1.6 - Nascondi dettaglio piatto	17
3.2.11	UC2 - gestione tavolo	17
3.2.12	UC2.1 - Generazione sessione tavolo	18
3.2.13	UC2.2 - Unione sessione tavolo	18
3.2.14	UC2.3 - Uscita sessione tavolo	18
3.2.15	UC2.4 - Generazione QR-code sessione tavolo	19
3.2.16	UC3 - lista ordini	19

3.2.17	UC3.1 - Visualizza lista ordini del tavolo	19
3.2.18	UC3.2 - Visualizza lista ordini personali	20
3.2.19	UC3.3 - Visualizza lista ordini in arrivo	20
3.2.20	UC3.4 - Visualizza in dettaglio lista ordini personali	20
3.2.21	UC3.5 - Ricezione piatto	21
3.2.22	UC3.6 - Sposta la lista in arrivo	21
3.2.23	UC3.7 - Mostra QR-code ordini	21
3.3	Utente Non Autenticato	22
3.3.1	UC4 - Area personale	22
3.3.2	UC4.1 - Registrazione	22
3.3.3	UC4.2 - Login	22
3.3.4	UC4.2 - Password dimenticata	23
3.4	Utente Autenticato	23
3.4.1	UC4.3 - Logout	23
3.5	Requisiti e obiettivi	24
3.6	Pianificazione	24
4	Analisi dei requisiti	25
4.1	Casi d'uso	25
4.2	Tracciamento dei requisiti	26
5	Progettazione e codifica	29
5.1	Tecnologie e strumenti	29
5.2	Ciclo di vita del software	29
5.3	Progettazione	29
5.4	Design Pattern utilizzati	29
5.5	Codifica	29
6	Verifica e validazione	31
7	Conclusioni	33
7.1	Consuntivo finale	33
7.2	Raggiungimento degli obiettivi	33
7.3	Conoscenze acquisite	33
7.4	Valutazione personale	33
A	Appendice A	35
	Bibliografia	39

Elenco delle figure

1.1	Punti di forza di SyncLab	2
1.2	I vari prodotti di SyncLab	3
2.1	Diagramma dei componenti per la web-app	6
2.2	Canale SyncLab su Discord	9
2.3	Borad di Trello per il progetto sushi-lab	9
2.4	Calendario personale per il progetto sushi-lab	10
2.5	Repo su GitHub per il progetto sushi-lab	11
4.1	Use Case - UC0: Scenario principale	25

Elenco delle tabelle

4.1	Tabella del tracciamento dei requisiti funzionali	27
4.2	Tabella del tracciamento dei requisiti qualitativi	27
4.3	Tabella del tracciamento dei requisiti di vincolo	27

Capitolo 1

Introduzione

Introduzione al contesto applicativo.

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

Esempio di citazione in linea
Manifesto Agile. URL: <http://agilemanifesto.org/iso/it/>.

Esempio di citazione nel pie' di pagina
citazione¹

1.1 SyncLab

SyncLab è una Innovative Company collocata in tutta italia, è nata nel 2002 con sede principale a Napoli ed è cresciuta velocemente. Attualmente, SyncLab ha 6 sedi in tutta italia, più di 300 dipendenti e più di 150 clienti diretti e finali.

SyncLab propone servizi Innovativi che aiutano i clienti nella realizzazione, progettazione e manutenzioni di soluzioni IT, sia dal punto di vista tecnologico, sia nel Governo del Cambiamento Organizzativo. L'azienda ha collobarobato con vari compagnie tra questi i più importanti sono: Tim, Trenitalia, HM, Grimaldi Lines, notartel, sky, eni, enel, vodafone, RayWay, Poste Italiane, Intesa Sanpaolo, Ministero dell'economia delle finanze, fastweb e UniCredit.

¹Daniel T. Jones James P. Womack. *Lean Thinking, Second Editon*. Simon & Schuster, Inc., 2010.



Figura 1.1: Punti di forza di SyncLab

1.1.1 Prodotti

Come accenato, SyncLab opera nel settore IT e i suoi prodotti nascono dalle competenze acquisite e maturate durante le loro 20 anni di collaborazioni. I prodotti coprono vari ambiti come quelli delle telecomunicazioni, utilities, finanza e salute.

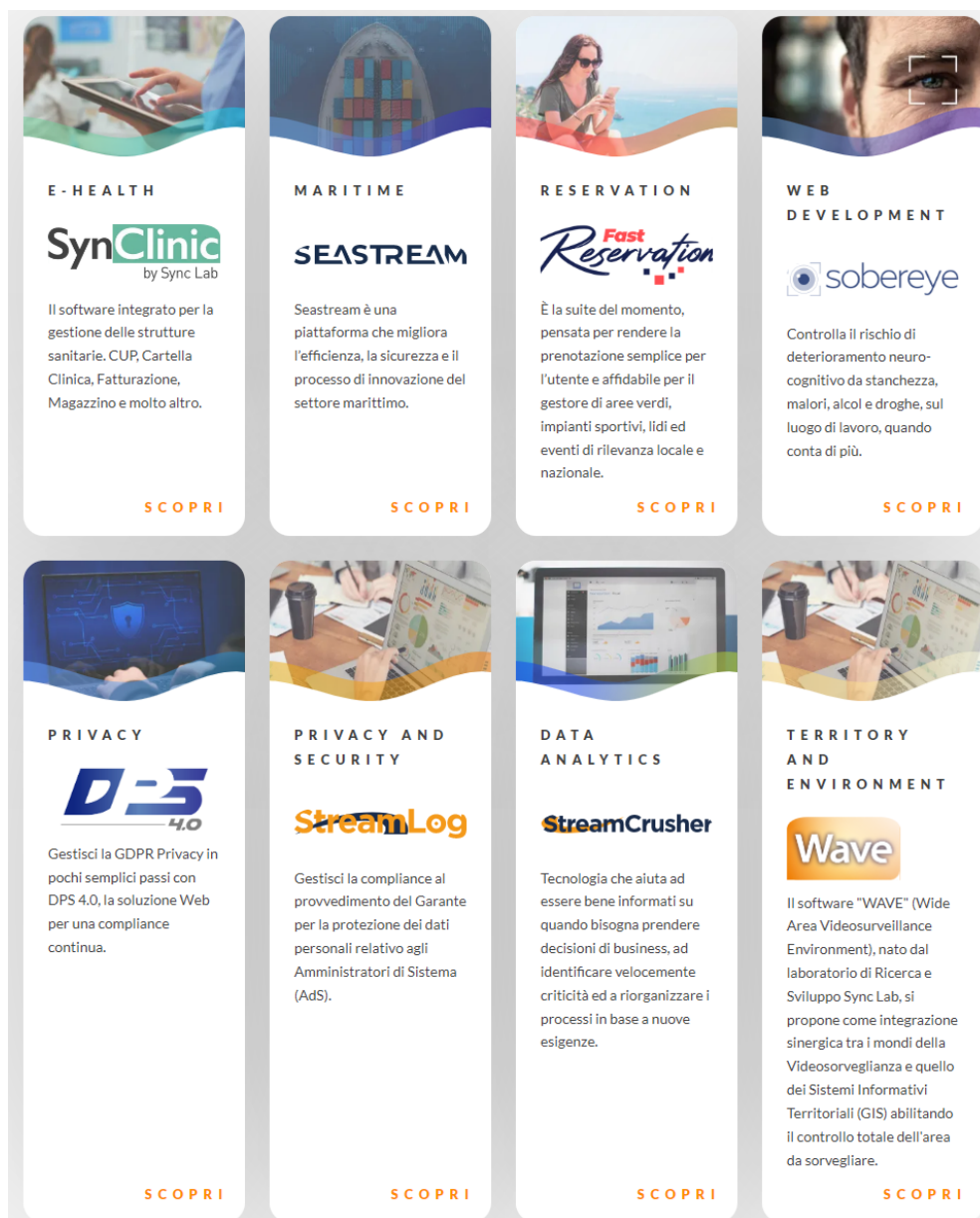


Figura 1.2: I vari prodotti di SyncLab

- * **SynClinic:** software integrato per la gestione delle strutture sanitarie, come il sistema di cartella clinica digitale, il servizio di fatturazione, la gestione informatizzata dei farmaci, i strumenti nativi di gestione amministrativa e tanto altro.
- * **SEASTREAM:** una piattaforma nata per migliorare e potenziare le attività di business nel settore armatoriali e di altri operatori del mercato marittimo
- * **FastReservation:** applicazione realizzato per rendere il sistema di prenotazione più facile e affidabile per gli utenti.

- * **sobereye:** soluzione per la sicurezza proattiva per la prevenzione degli incidenti nei settori di trasporti, estrazione, costruzioni e industriale.
- * **DPS 4.0:** applicativo che permette di gestire la GDPR Privacy Policy. L'applicativo, tramite una guida semplice sviluppata dagli ingegneri esperti nell'ambito dei user experience, offre la possibilità di modificare e aggiornare i documenti sulla privacy con il minimo sforzo.
- * **StreamLog:** soluzione per la protezione dei dati personali relativo agli Amministratori di Sistema, dà possibilità di effettuare il controllo degli accessi degli utenti ai sistemi in modo semplice ed efficace.
- * **StreamCrusher:** tecnologia che serve per aiutare a prendere decisioni di business, indentificando velocemente i punti critici e riformando i processi in base alle nuove esigenze.
- * **Wave:** software nato per i sistemi di vidersorveglianza, con l'obiettivo di avere una maggiore copertura territoriale col minor numero di telecamere installate e possibilmente utilizzare il minor numero possibile di risorse.

1.2 Organizzazione del testo

Il secondo capitolo describe ...

Il terzo capitolo approfondisce ...

Il quarto capitolo approfondisce ...

Il quinto capitolo approfondisce ...

Il sesto capitolo approfondisce ...

Nel settimo capitolo describe ...

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*^[§];
- * i termini in lingua straniera o facenti parti del gergo tecnico sono evidenziati con il carattere *corsivo*.

Capitolo 2

Il progetto di stage

Brevissima introduzione al capitolo

2.1 L'idea

Nei ristoranti sushi moderni quasi tutti applicano la formula "all-you-can-eat", con tale formula i ristoranti offrono la possibilità di ordinare senza limiti ad un prezzo fisso. I consumi dei clienti in questi locali è maggiore rispetto ai ristoranti à la carte, perché con la formula all-you-can-eat spesso i clienti mangiano oltre il loro livello di sazietà, quindi comporterà un elevato numero di ordinazioni da gestire creando così il problema di capire poi chi ha ordinato i piatti singoli nel momento della consegna. L'idea dunque è stata quella di creare una web-app per la gestione delle ordinazione dei piatti.

2.2 La soluzione individuata

SyncLab ha deciso di risolvere questo problema tramite una web-app, essa è composta da due componenti la parte front-end e la parte back-end. La parte di back-end dovrà essere implementata tramite spring, il back-end ha il ruolo principale di un web server, il quale ha il compito di comunicare con il data-base, dove verranno salvati tutte le informazioni dei piatti e degli eventuali ordini dei ristoranti. Il web server deve offrire delle API, con le quali la parte front-end, che utilizza Angular, comunicherà con il server.

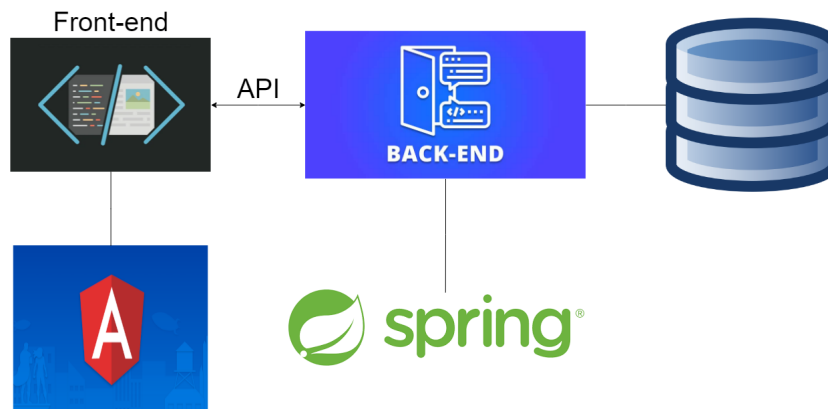


Figura 2.1: Diagramma dei componenti per la web-app

Angular è un framework open source gratuita per lo sviluppo di applicazione web. Il linguaggio di programmazione utilizzato è TypeScript, TypeScript estende la sintassi di JavaScript, quindi qualsiasi codice scritto in JavaScript è eseguibile anche tramite TypeScript senza nessuna modifica o aggiunta di codice. Grazie a TypeScript il codice generato da Angular gira su tutti i principali web browser comuni come Google Chrome, Firefox, Safari, Opera, Microsoft Edge e tanti altri. La web-app, per ciascuna parte dell'applicazione è stata affidata a più persone. Discutendo con il tutor aziendale, Fabio Pallaro, abbiamo individuato i principali obiettivi per realizzare la web-app ed a me è stato assegnato il compito di sviluppare la visuale dell'applicazione.

2.3 obiettivi richiesti

2.3.1 Notazione

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.

2.3.2 obiettivi fissati

Si prevede lo svolgimento dei seguenti obiettivi:

- * Obbligatori:
 - O01: Acquisizione competenze sulle tematiche sopra descritte;
 - O02: Capacità di raggiungere gli obiettivi richiesti in autonomia seguendo il cronoprogramma;

- Portare a termine le implementazioni previste con una percentuale di superamento pari al 80%
- * Desiderabili:
 - D01: Portare a termine le implementazioni previste con una percentuale di superamento pari al 100%
- * Facoltativi:
 - F01: Apportare un valore aggiunto al gruppo di lavoro durante le fasi di progettazione delle interfacce.

2.3.3 Pianificazione del lavoro

Pianificazione settimanale

- * Prima Settimana (40 ore)
 - Incontro con persone coinvolte nel progetto per discutere i requisiti e le richieste relativamente al sistema da sviluppare;
 - Presentazione strumenti di lavoro per la condivisione del materiale di studio e per la gestione dell'avanzamento;
 - Condivisione scaletta di argomenti;
 - Ripasso del linguaggio Java SE;
 - Ripasso concetti Web (Servlet, servizi Rest, Json ecc.).
- * Seconda Settimana (40 ore)
 - Studio principi generali di Spring Core (IOC, Dependency Injection);
 - Studio SpringBoot;
 - Studio Spring Data/DataRest.
- * Terza Settimana (40 ore)
 - Ripasso linguaggio Javascript;
 - Studio del linguaggio TypeScript.
- * Quarta Settimana (40 ore)
 - Studio piattaforma NodeJS e AngularCLI;
 - Studio framework Angular.
- * Quinta Settimana (40 ore)
 - Analisi e studio del progetto SushiLab;
 - Progettazione ed implementazione della nuova maschera di accesso.
- * Sesta Settimana (40 ore)
 - Progettazione ed implementazione nuova maschera "Inserimento Ordine e Gestione Tavolo".

* Settima Settimana (40 ore)

- Progettazione ed implementazione nuova maschera "Merge Ordini e Visualizzazione Ordine unico".

* Ottava Settimana - Conclusione (40 ore)

- Verifica del funzionamento della web-app;
- Validazione della web-app;
- Termine integrazioni e collaudo finale.

2.4 Modalità di lavoro

L'azienda utilizza lo sviluppo agile del software. La modalità agile aiuta a ridurre il rischio di fallimento sviluppando il software in finestre di tempo limitate chiamate iterazioni che, in genere, durano qualche settimana. Ogni iterazione è un piccolo progetto a sé stante e deve contenere tutto ciò che è necessario per rilasciare un piccolo incremento nelle funzionalità del software: pianificazione, analisi dei requisiti, progettazione, implementazione, test e documentazione. Ogni settimana viene effettuato un incontro con il tutor aziendale per discutere dei problemi trovati durante lo sviluppo e il punto di situazione del progetto.

2.5 Strumenti di comunicazione

Per avere una buona comunicazione con l'azienda si è deciso di utilizzare:

Discord:

una piattaforma di VoIP^G, messaggistica istantanea e distribuzione digitale. Su discord gli utenti comunicano con chiamate vocali, video chiamate ed è anche possibile condividere lo schermo.

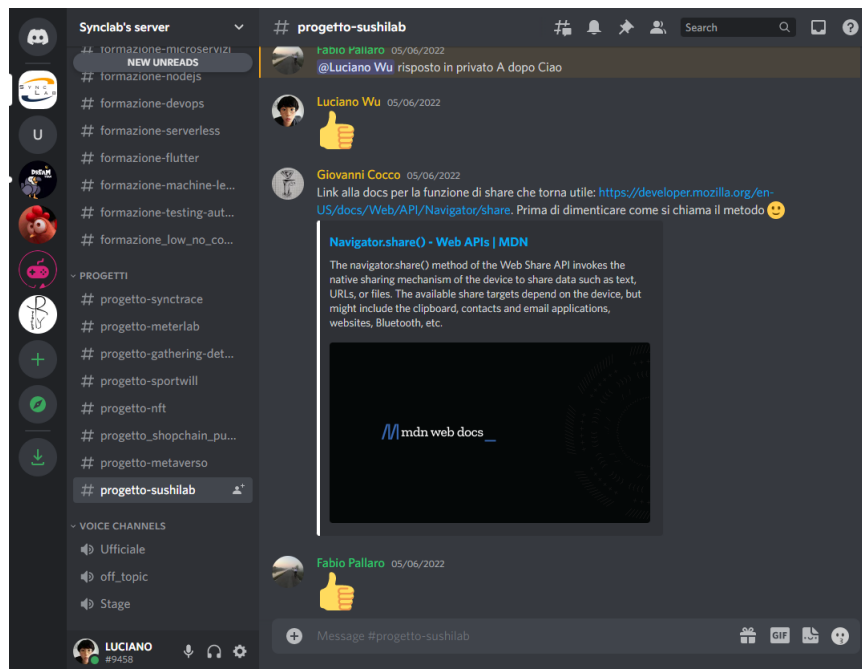


Figura 2.2: Canale SyncLab su Discord

Trello:

un software gestionale in stile Kanban^G, in cui è possibile pianificare il progetto, condividere lo stato di svolgimento di una card con altri collaboratori, spostare vari card tra le liste e assegnare ad un utente.

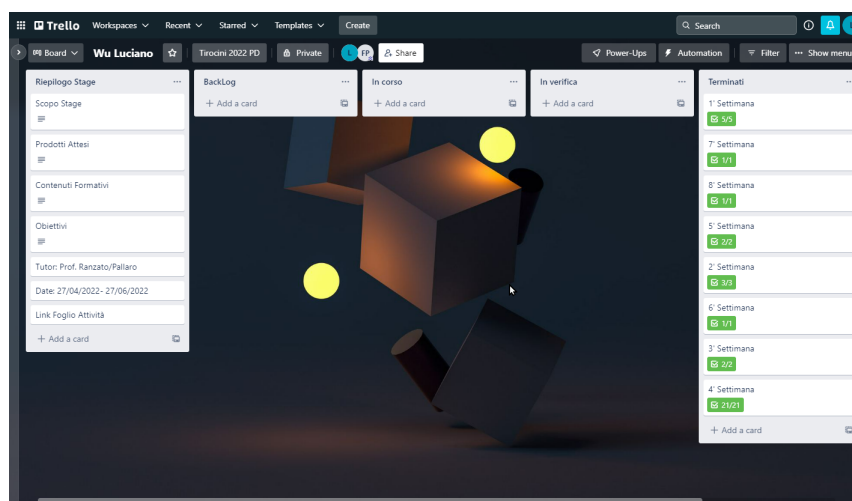


Figura 2.3: Borad di Trello per il progetto sushi-lab

Google Sheets:

una web-app che fornisce tutte le funzionalità di un foglio elettronico, lo abbiamo utilizzato come un diario giornaliero, dove vengono descritti il compito svolto durante una certa giornata.

	A	B	C	D	E	F
17	2022-05-18	Studio su lifecycle hooks e comunicazione tra i componenti di angular	✓			
18	2022-05-19	Studio su chiamate http, rest, build e deploy	✓			
19	2022-05-20	Studio su lazy loading e reactive forms	✓			
20	2022-05-23	Progettazione maschera accesso	✓			
21	2022-05-24	Progettazione maschera tavolo	✓			
22	2022-05-25	Progettazione maschera ordini	✓			
23	2022-05-26	Progettazione maschera password dimenticata	✓			
24	2022-05-27	Progettazione guard	✓			
25	2022-05-30	Progettazione maschera gestione tavolo(creazione sessione tavolo)	✓			
26	2022-05-31	Progettazione maschera gestione tavolo(unione sessione tavolo)	✓			
27	2022-06-01	Progettazione maschera gestione ordini del tavolo	✓			
28	2022-06-02	Progettazione maschera gestione ordini personali	✓			
29	2022-06-03	Progettazione maschera gestione ordini in arrivo	✓			
30	2022-06-06	Progettazione merge ordini tavolo	✓			
31	2022-06-07	Progettazione visualizza ordine unico	✓			
32	2022-06-08	Progettazione sposta ordini in arrivo	✓			
33	2022-06-09	Abbellimento css maschera tavolo	✓			
34	2022-06-10	Abbellimento css maschera ordini	✓			
35	2022-06-14	Abbellimento css maschera password dimenticata	✓			
36	2022-06-15	Controllo completo su i file di stile	✓			
37	2022-06-16	Controllo completo su file di guard	✓			
38	2022-06-17	Controllo completo su file di service	✓			
39	2022-06-20	Controllo completo su file di type script	✓			
40	2022-06-21	Stesura documento analisi tecnica	✓			
41	2022-06-22	Completamento documento analisi tecnica	✓			
42	2022-06-23	Preparazione collaudo finale	✓			
43	2022-06-24	Collaudo finale	✓			
44						
45						
46						

Figura 2.4: Calendario personale per il progetto sushi-lab

2.5.1 Google Meet:

un software nata per le videochiamate sviluppato da Google, in cui è possibile mandare messaggi e condividere lo schermo e video contemporaneamente, lo abbiamo utilizzato per alcuni incontri con alcuni collaboratori esterni per analisi dei requisiti.

GitHub:

servizio di hosting per il sviluppo software, è implementato insieme con lo strumento di controllo versione distribuito Git^G, nel mio caso è utilizzato per condividere e tracciare i file con gli altri collaboratori del progetto.

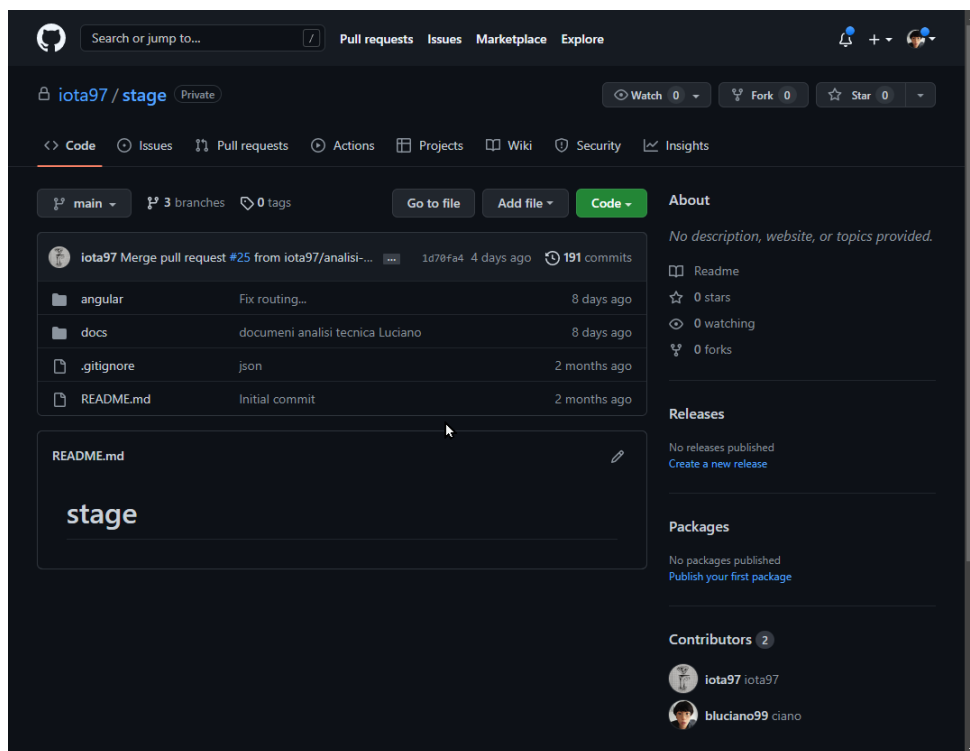


Figura 2.5: Repo su GitHub per il progetto sushi-lab

Capitolo 3

Analisi dei requisiti

In questo capitolo vengono trattati le analisi dei requisiti del modulo di front-end della web-app, con i vari casi d'uso ed elenco dei requisiti.

3.1 Descrizione generale

3.1.1 interfacce della web-app

Interfaccia menù

L'utente può visualizzare il menù di un ristorante dopo aver scansionato il QR-code di un ristorante presente sul tavolo. Il menù è composto da un insieme di categorie, in cui ci sono tutti i piatti appartenenti a quella categoria. I piatti sono ordinati in base al suo id che è un numero univoco dentro ogni menù.

Interfaccia lista ordini

L'utente può vedere i piatti ordinati del suo tavolo di appartenenza, dove ci sono anche i piatti ordinati dalle altre persone del tavolo, inoltre può vedere i suoi piatti personali e i piatti in arrivo.

Interfaccia gestione tavolo

In questa maschera utente può creare una sessione di tavolo se non appartiene ad nessuna sessione, altrimenti può visualizzare il suo QR-code in modo da fare entrare gli altri nella sua sessione di tavolo.

Interfaccia area personale

Qui l'utente può effettuare la login, di seguito se ha degli allergeni potrà inserire degli ingredienti nella blacklist^G in modo tale di non visualizzare i piatti contenenti quegli ingredienti nella sezione menù.

3.1.2 Caratteristiche degli Utenti

In questa sezione vengono descritti tutte le Caratteristiche degli utenti che possono utilizzare la web-app.

Utente non autenticato

Con il termine utente non autenticato ci si riferisce ad una qualsiasi persona non autenticata nel sistema, che può sfruttare le funzionalità di base offerte dalla piattaforma, ossia:

- * Visualizzare il menù del ristorante;
- * Visualizzare i singoli piatti in modalità dettaglio;
- * Creare una sessione di tavolo;
- * Unire ad una sessione di tavolo già esistente;
- * Uscire dalla sessione di tavolo;
- * Aggiungere piatti negli ordini;
- * Aggiungere note ai piatti ordinati;
- * Spostare ordini in arrivo;
- * Marcare i piatti in arrivo come arrivato;
- * Registrare nella piattaforma;
- * Effettuare la login.

Utente autenticato

Invece, con il termine “utente autenticato” ci si riferisce ad una persona registrata nel database e che ha effettuato l’accesso nella piattaforma, la quale, oltre a sfruttare le funzionalità dell’utente non autenticato, può anche:

- * Aggiungere piatti nei preferiti;
- * Rimuovere piatti dai preferiti;
- * Dare un recensione ad un piatto;
- * Aggiungere ingredienti non voluti;
- * Rimuovere gli ingredienti non voluti;
- * Effettuare logout.

3.1.3 Tecnologie utilizzate

Per sviluppare la piattaforma verranno utilizzare le seguenti tecnologie:

- * HTML5^G: per creare la struttura dell’interfaccia utente;
- * CSS3^G: per lo stile dell’interfaccia, viene utilizzato la sintassi SCSS;
- * StopLight^G: per simulare le chiamate Rest API;
- * Angular^G: per la creazione dell’interfaccia utente.

3.2 Casi D'uso

3.2.1 Introduzione

In questa sezione verranno presentati i casi d'uso individuati durante la fase di analisi dei requisiti, i quali fanno riferimento a tutte le funzionalità che la web-app SushiLab dovrà offrire ad ogni utente che vorrà interfacciarsi con essa.

3.2.2 Attori primari

- * **Utente non Autenticato:** utente che non ha ancora effettuato la fase di autenticazione sulla piattaforma. Può essere in possesso o meno delle credenziali per l'autenticazione. Avrà funzionalità limitate rispetto ad un utente autenticato;
- * **Utente Autenticato:** utente che ha effettuato l'autenticazione alla piattaforma tramite le proprie credenziali. Ha accesso ad ogni funzionalità messa a disposizione dalla piattaforma;
- * **Utente Generico:** può essere sia un utente autenticato che un utente non autenticato.

3.2.3 Utente generico

3.2.4 UC1 - Visualizza menù

- * **Descrizione:** L'utente visualizza il menù del ristorante.
- * **Attore Primario:** Utente generico.
- * **Precondizione:** L'utente si trova dentro la web-app sushiLab.
- * **Postcondizione:** Viene visualizzato il menù del ristorante.
- * **Scenrio principale:**
 - L'utente si trova dentro il sistema;
 - L'utente clicca sul bottone menù.

3.2.5 UC1.1 - Visualizza categorie

- * **Descrizione:** L'utente visualizza le categorie del menù.
- * **Attore Primario:** Utente generico.
- * **Precondizione:** L'utente si trova dentro la sezione menù.
- * **Postcondizione:** Viene visualizzato i nomi delle categorie.
- * **Scenrio principale:**
 - L'utente si trova sezione menù;
 - Viene mostrato le categorie del menù.

3.2.6 UC1.2 - Visualizza piatto

- * **Descrizione:** L'utente visualizza i piatti del menù mostrando il numero, nome, prezzo, ingredienti^G, allergeni, limitazioni^G e la quantità. La quantità di default^G è 0 che vuole dire non è stato ordinato.
- * **Attore Primario:** Utente generico.
- * **Precondizione:** L'utente si trova dentro la sezione menù.
- * **Postcondizione:** Viene visualizzato i piatti del menù.
- * **Scenrio principale:**
 - L'utente si trova sezione menù;
 - Viene mostrato i piatti del menù.

3.2.7 UC1.3 - Aumenta quantità

- * **Descrizione:** L'utente aumenta la quantità di un piatto nel menù.
- * **Attore Primario:** Utente generico.
- * **Precondizione:** L'utente si trova dentro la sezione menù.
- * **Postcondizione:** Viene aggiunto il piatto specifico con la quantità aggiornata negli ordini.
- * **Scenrio principale:**
 - L'utente si trova sezione menù;
 - L'utente clicca sul bottone + di un piatto;
 - Viene aggiunto il piatto negli ordini.
- * **Scenrio alternativo:**
 - L'utente si trova sezione menù;
 - L'utente clicca sul bottone + di un piatto che è già presente negli ordini;
 - Viene aumentato la quantità del piatto negli ordini.

3.2.8 UC1.4 - Diminuisce quantità

- * **Descrizione:** L'utente dimiuisce la quantità di un piatto nel menù.
- * **Attore Primario:** Utente generico.
- * **Precondizione:** L'utente si trova dentro la sezione menù.
- * **Postcondizione:** Viene dimiuito la quantità del piatto specifico negli ordini.
- * **Scenrio principale:**
 - L'utente si trova sezione menù;
 - L'utente clicca sul bottone - di un piatto con quantità maggiore di 1;
 - Viene diminuito la quantità del piatto negli ordini.

* **Scenrio alternativo:**

- L'utente si trova sezione menù;
- L'utente clicca sul bottone - di un piatto con quantità uguale a 1;
- Viene rimosso il piatto dagli ordini.

3.2.9 UC1.5 - Visualizza dettaglio piatto

* **Descrizione:** L'utente visualizza i dettagli di un piatto nel menù, mostrando la recensione del piatto e il text-box^G per inserire una nota.

* **Attore Primario:** Utente generico.

* **Precondizione:** L'utente si trova dentro la sezione menù.

* **Postcondizione:** Viene visualizzato i dettagli di un piatto specifico.

* **Scenrio principale:**

- L'utente si trova sezione menù;
- L'utente clicca sul bottom mostra dettagli;
- Viene mostrato i dettagli di un piatto del menù.

3.2.10 UC1.6 - Nascondi dettaglio piatto

* **Descrizione:** L'utente nasconde i dettagli di un piatto specifico.

* **Attore Primario:** Utente generico.

* **Precondizione:** L'utente si trova dentro la sezione menù con un piatto in modalità dettaglio.

* **Postcondizione:** Viene nascosto i dettagli del piatto specifico.

* **Scenrio principale:**

- L'utente si trova sezione menù;
- L'utente clicca sul bottom nascondi dettagli;
- Viene mostrato i dettagli di un piatto del menù.

3.2.11 UC2 - gestione tavolo

* **Descrizione:** L'utente visualizza la maschera di gestione tavolo.

* **Attore Primario:** Utente generico.

* **Precondizione:** L'utente si trova dentro la web-app sushiLab.

* **Postcondizione:** Viene visualizzato la maschera di gestione tavolo.

* **Scenrio principale:**

- L'utente si trova dentro il sistema;
- Viene mostrato la maschera di gestione tavolo.

3.2.12 UC2.1 - Generazione sessione tavolo

- * **Descrizione:** L'utente genera la sessione del tavolo.
- * **Attore Primario:** Utente generico.
- * **Precondizione:** L'utente si trova dentro la sezione gestione tavolo.
- * **Postcondizione:** L'utente entra nella sessione generata del tavolo.
- * **Scenrio principale:**
 - L'utente si trova dentro la sezione gestione tavolo;
 - L'utente clicca sul bottone crea sessione;
 - L'utente viene inserito nella sessione creata.

3.2.13 UC2.2 - Unione sessione tavolo

- * **Descrizione:** L'utente si unisce alla sessione del tavolo.
- * **Attore Primario:** Utente generico.
- * **Precondizione:** L'utente si trova dentro la sezione gestione tavolo.
- * **Postcondizione:** L'utente entra nella sessione che è stata inserita.
- * **Scenrio principale:**
 - L'utente si trova dentro la sezione gestione tavolo;
 - L'utente clicca sul bottone unisciti a una sessione;
 - L'utente inserisce il numero della sessione;
 - L'utente clicca sul bottone unisciti;
 - L'utente viene inserito nella sessione.
- * **Scenrio alternativo:**
 - L'utente si trova dentro la sezione gestione tavolo;
 - L'utente clicca sul bottone unisciti a una sessione;
 - L'utente inserisce il numero della sessione inesistente;
 - L'utente clicca sul bottone unisciti;
 - L'utente non viene inserito nella sessione.

3.2.14 UC2.3 - Uscita sessione tavolo

- * **Descrizione:** L'utente esce dalla sessione del tavolo.
- * **Attore Primario:** Utente generico.
- * **Precondizione:** L'utente si trova dentro la sezione gestione tavolo ed è dentro ad una sessione.
- * **Postcondizione:** L'utente esce dalla sessione generata del tavolo.

- * **Scenrio principale:**

- L'utente si trova dentro la sezione gestione tavolo;
- L'utente clicca sul bottone esci dalla sessione;
- L'utente viene rimosso dalla sessione.

3.2.15 UC2.4 - Generazione QR-code sessione tavolo

- * **Descrizione:** L'utente genera il QR-code dalla sessione del tavolo per mostrarlo agli altri, che li permetterà di unire alla sessione direttamente scansionando il QR-code.

- * **Attore Primario:** Untente generico.

- * **Precondizione:** L'utente si trova dentro la sezione gestione tavolo ed è dentro ad una sessione.

- * **Postcondizione:** L'utente genera il QR-code dalla sessione del tavolo.

- * **Scenrio principale:**

- L'utente si trova dentro la sezione gestione tavolo;
- L'utente genera il QR-code dalla sessione.

3.2.16 UC3 - lista ordini

- * **Descrizione:** L'utente visualizza la maschera di gestione ordini.

- * **Attore Primario:** Untente generico.

- * **Precondizione:** L'utente si trova dentro ad una sessione di tavolo.

- * **Postcondizione:** Viene visualizzato la maschera di gestione ordini.

- * **Scenrio principale:**

- L'utente si trova dentro il sistema con una sessione di tavolo attiva;
- Viene mostrato la maschera di gestione ordini.

3.2.17 UC3.1 - Visualizza lista ordini del tavolo

- * **Descrizione:** L'utente visualizza la lista degli ordini della sessione di tavolo in cui si trova.

- * **Attore Primario:** Untente generico.

- * **Precondizione:** L'utente si trova dentro la sezione lista ordini.

- * **Postcondizione:** Viene visualizzato la lista degli ordini del tavolo.

- * **Scenrio principale:**

- L'utente si trova dentro la sezione gestione ordini;
- L'utente clicca sul bottone "tavolo";
- Viene mostrato la lista dei piatti ordinati del tavolo.

3.2.18 UC3.2 - Visualizza lista ordini personali

- * **Descrizione:** L'utente visualizzato la lista degli ordini personali.
- * **Attore Primario:** Utente generico.
- * **Precondizione:** L'utente si trova dentro la sezione lista ordini.
- * **Postcondizione:** Viene visualizzato la lista degli ordini personali.
- * **Scenrio principale:**
 - L'utente si trova dentro la sezione gestione ordini;
 - L'utente clicca sul bottone "personali";
 - Viene mostrato la lista dei piatti ordinati dall'utente stesso.

3.2.19 UC3.3 - Visualizza lista ordini in arrivo

- * **Descrizione:** L'utente visualizza la lista degli ordini in arrivo.
- * **Attore Primario:** Utente generico.
- * **Precondizione:** L'utente
- * **Postcondizione:** Viene visualizzato la lista lista degli ordini in arrivo.
- * **Scenrio principale:**
 - L'utente si trova dentro la sezione gestione ordini;
 - L'utente clicca sul bottone "in arrivo";
 - Viene mostrato la lista dei piatti in arrivo.

3.2.20 UC3.4 - Visualizza in dettaglio lista ordini personali

- * **Descrizione:** L'utente visualizza la lista degli ordini personali in modalità dettaglio.
- * **Attore Primario:** Utente generico.
- * **Precondizione:** L'utente si trova dentro la sezione lista ordini personali.
- * **Postcondizione:** Viene visualizzato la lista degli ordini personali con i piatti in modalità dettaglio.
- * **Scenrio principale:**
 - L'utente si trova dentro la sezione gestione ordini personali;
 - L'utente clicca sul bottone "lente" con il +;
 - Viene mostrato la lista dei piatti ordinati personali in modalità dettaglio.

3.2.21 UC3.5 - Ricezione piatto

- * **Descrizione:** L'utente marca un piatto in arrivo come ricevuto.
- * **Attore Primario:** Utente generico.
- * **Precondizione:** L'utente si trova dentro la sezione gestione ordini in arrivo e ha almeno un piatto nella lista in arrivo.
- * **Postcondizione:** L'utente marca il piatto come arrivato diminuendo di 1 la sua quantità.
- * **Scenrio principale:**
 - L'utente si trova dentro la sezione gestione lista ordini in arrivo;
 - L'utente clicca sul bottone "v" di un piatto;
 - Viene diminuito di 1 la sua quantità.
- * **Scenrio alternativo:**
 - L'utente si trova dentro la sezione gestione lista ordini in arrivo;
 - L'utente clicca sul bottone "v" di un piatto con quantità uguale a 1;
 - Viene diminuito di 1 la quantità del piatto e viene disabilitato il bottone.

3.2.22 UC3.6 - Sposta la lista in arrivo

- * **Descrizione:** L'utente sposta la lista degli ordini in arrivo.
- * **Attore Primario:** Utente generico.
- * **Precondizione:** L'utente si trova dentro la sezione lista ordini del tavolo.
- * **Postcondizione:** Viene spostato la lista degli ordini del tavolo in arrivo.
- * **Scenrio principale:**
 - L'utente si trova dentro la sezione gestione ordini del tavolo;
 - L'utente clicca sul bottone sposta la lista in arrivo;
 - Viene spostato la lista degli piatti ordinati personali in modalità dettaglio;
 - Viene mostrato all'utente il messaggio "ordini spostati correttamente".

3.2.23 UC3.7 - Mostra QR-code ordini

- * **Descrizione:** L'utente genera il QR-code della lista ordini per dopo mostrarlo al cameriere.
- * **Attore Primario:** Utente generico.
- * **Precondizione:** L'utente si trova dentro la sezione lista ordini del tavolo.
- * **Postcondizione:** Viene mostrato il QR-code della lista degli ordini.
- * **Scenrio principale:**
 - L'utente si trova dentro la sezione gestione ordini del tavolo;
 - L'utente clicca sul bottone QR-code;
 - Viene generato il QR-code degli ordini.

3.3 Utente Non Autenticato

3.3.1 UC4 - Area personale

- * **Descrizione:** L'utente visualizza la maschera dell'area personale.
- * **Attore Primario:** Utente non autenticato.
- * **Precondizione:** L'utente si trova dentro la web-app sushiLab.
- * **Postcondizione:** Viene visualizzato la maschera dell'area personale.
- * **Scenrio principale:**
 - L'utente si trova dentro il sistema;
 - Viene mostrato la maschera dell'area personale.

3.3.2 UC4.1 - Registrazione

- * **Descrizione:** L'utente viene registrato nella piattaforma.
- * **Attore Primario:** Utente non autenticato.
- * **Precondizione:** L'utente si trova dentro la web-app sushiLab.
- * **Postcondizione:** Viene salvato i dati dell'utente inseriti durante la fase di registrazione nel data-base.
- * **Scenrio principale:**
 - L'utente si trova dentro l'area personale;
 - L'utente clicca sul bottone registrati;
 - Viene mostrato il form di registrazione;
 - L'utente inserisce l'email;
 - L'utente inserisce la password;
 - L'utente ripete la password;
 - L'utente clicca sul bottone registrati;
 - Viene registrato correttamente l'account.
- * **Estensioni:**
 - L'utente inserisce l'email già esistente nel data-base;
 - Non viene registrato l'account.

3.3.3 UC4.2 - Login

- * **Descrizione:** L'utente effettua login nella piattaforma.
- * **Attore Primario:** Utente non autenticato.
- * **Precondizione:** L'utente si trova dentro la web-app sushiLab.
- * **Postcondizione:** Viene effettuato il login.

* **Scenrio principale:**

- L'utente si trova dentro l'area personale;
- L'utente inserisce l'email;
- L'utente inserisce la password;
- L'utente clicca sul bottone login;
- Viene effettuato il login correttamente.

* **Estensioni:**

- L'utente inserisce l'email non esistente nel data-base o una password errata;
- Non viene effettuato il login.

3.3.4 UC4.2 - Password dimenticata

* **Descrizione:** L'utente reimposta la password del proprio account.

* **Attore Primario:** Utente non autenticato.

* **Precondizione:** L'utente si trova dentro la web-app sushiLab.

* **Postcondizione:** Viene aggiornato la nuova password nel data-base.

* **Scenrio principale:**

- L'utente si trova dentro l'area personale;
- L'utente clicca sul bottone password dimenticata;
- Vine mostrato il form di recupero password;
- L'utente inserisce l'email;
- L'utente clicca sul bottone ottieni codice;
- L'utente arriva nel secondo form tramite il link mandato tramite email;
- L'utente inserisce la password;
- L'utente ripete la password;
- L'utente clicca sul bottone cambia password;
- Viene cambiato correttamente la password.

* **Estensioni:**

- L'utente inserisce l'email non esistente nel data-base;
- Non viene effettuato il cambio password.

3.4 Utente Autenticato

3.4.1 UC4.3 - Logout

* **Descrizione:** L'utente effettua logout.

* **Attore Primario:** Utente autenticato.

- * **Precondizione:** L'utente si trova dentro la web-app sushiLab ed ha effettuato la login.
- * **Postcondizione:** Viene effettuato il logout.
- * **Scenrio principale:**
 - L'utente si trova dentro l'area personale;
 - L'utente inserisce l'email;
 - L'utente inserisce la password;
 - L'utente clicca sul bottone login;
 - Viene effettuato il login correttamente.
- * **Estensioni:**
 - L'utente inserisce l'email non esistente nel data-base o una password errata;
 - Non viene effettuato il login.

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.

3.5 Requisiti e obiettivi

3.6 Pianificazione

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.

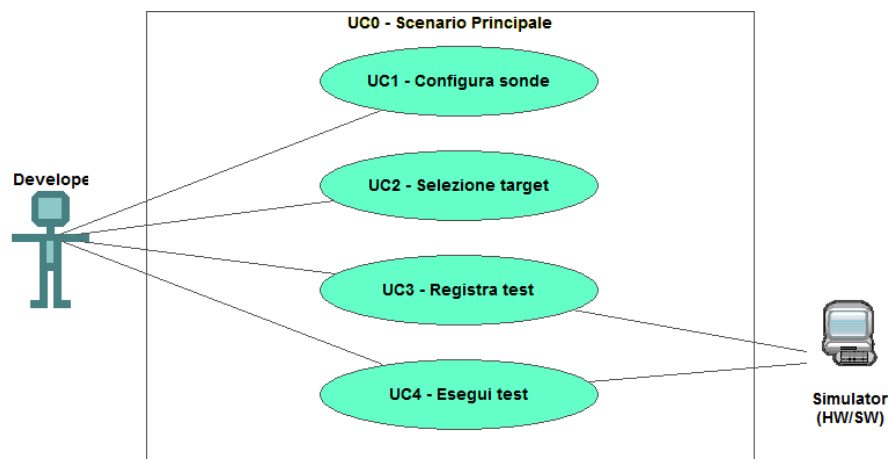


Figura 4.1: Use Case - UC0: Scenario principale

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.

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.

Tecnologia 1

Descrizione Tecnologia 1.

Tecnologia 2

Descrizione Tecnologia 2

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

Riferimenti bibliografici

James P. Womack, Daniel T. Jones. *Lean Thinking, Second Editon*. Simon & Schuster, Inc., 2010 (cit. a p. [1](#)).

Siti web consultati

Manifesto Agile. URL: <http://agilemanifesto.org/iso/it/> (cit. a p. [1](#)).