

Università degli Studi di Padova

DIPARTIMENTO DI MATEMATICA "TULLIO LEVI-CIVITA"

CORSO DI LAUREA IN INFORMATICA



Tecnologie web per lo sviluppo di
applicazioni mobile ibride

Tesi di laurea

Relatore

Prof. Paolo Baldan

Laureando

Nicola Salvatore

ANNO ACCADEMICO 2019-2020

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 Salvatore Nicola presso l'azienda Alternative Studio di Costa Francesco.

L'obiettivo dello stage è stato il rifacimento della applicazione per smartphone a disposizione di UCIS (Unità Cinofile Italiane da Soccorso), che si occupa di registrare la geolocalizzazione delle unità cinofile durante esercitazioni, addestramenti, e anche operazioni di emergenza.

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

— Confucius

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, Settembre 2020

Nicola Salvatore

Indice

1	Introduzione	1
1.1	Convenzioni tipografiche	1
1.2	Scopo del documento	1
1.3	Organizzazione del testo	1
1.4	L'azienda	1
1.5	L'idea	2
1.6	Metodo di Lavoro	2
1.7	Tecnologie utilizzate	3
2	Processi e metodologie	5
2.1	Processo sviluppo prodotto	5
3	Descrizione dello stage	7
3.1	Introduzione al progetto	7
3.2	Analisi preventiva dei rischi	7
3.3	Requisiti e obiettivi	7
3.4	Pianificazione	7
4	Analisi dei requisiti	9
4.1	Casi d'uso	9
4.2	Tracciamento dei requisiti	10
5	Progettazione e codifica	13
5.1	Tecnologie e strumenti	13
5.2	Ciclo di vita del software	13
5.3	Progettazione	13
5.4	Design Pattern utilizzati	13
5.5	Codifica	13
6	Verifica e validazione	15
7	Conclusioni	17
7.1	Consuntivo finale	17
7.2	Raggiungimento degli obiettivi	17
7.3	Conoscenze acquisite	17
7.4	Valutazione personale	17
A	Appendice A	19

Acronyms	23
Glossary	25
Bibliografia	27

Elenco delle figure

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

Elenco delle tabelle

4.1	Tabella del tracciamento dei requisiti funzionali	11
4.2	Tabella del tracciamento dei requisiti qualitativi	11
4.3	Tabella del tracciamento dei requisiti di vincolo	11

Capitolo 1

Introduzione

1.1 Convenzioni tipografiche

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*.

1.2 Scopo del documento

Qui devo scrivere i contenuti di ciò che dirò.

1.3 Organizzazione del testo

Il secondo capitolo descrive ...

Il terzo capitolo approfondisce ...

Il quarto capitolo approfondisce ...

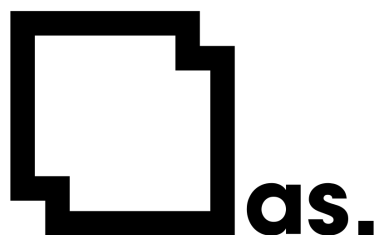
Il quinto capitolo approfondisce ...

Il sesto capitolo approfondisce ...

Nel settimo capitolo descrive ...

1.4 L'azienda

Alternative Studio è una web agency che fornisce soluzioni professionali su misura, costruite secondo le esigenze del cliente. Si occupa principalmente di sviluppo web e



marketing. È un'azienda piccola che raccoglie poche risorse umane, ma molte energie che continuano a spingere per crescere. Opera da appena sei anni nel settore del web development, ma ha abbracciato anche altre iniziative, collaborando in progetti più grandi con altre realtà. Negli ultimi anni l'azienda si è cimentata nello sviluppo di un gestionale per l'organizzazione [Unità Cinofile Italiane da Soccorso](#). Quest ultimo possiede un'api volta alla ricezione e all'elaborazione di attività registrate durante addestramento, soccorso o esercitazioni.

1.5 L'idea

L'attuale applicazione per smartphone che si occupa di registrare le attività succitate è stata sviluppata molti anni fa ed essendo molto instabile è nata l'esigenza di un refactoring di quest'ultima. Volendo utilizzare tecnologie moderne per lo sviluppo mobile e crossplatform è nata l'esigenza di un'indagine preventiva sulle tecnologie da utilizzare.

Alternative Studio ha pensato, quindi, di aprire una posizione perfetta per un laureando, che cerchi un progetto formante, che lo metta continuamente alla prova.

1.6 Metodo di Lavoro

Dato le contenute risorse umane a disposizione Alternative Studio adotta un ciclo di sviluppo software [Incrementale](#) con qualche introduzione di processi da quello [Agile](#), in particolare dal metodo [Scrum](#). Durante lo stage lo studente è stato incaricato anche di introdurre qualche concetto di questi cicli di vita all'interno del contesto aziendale, come quello di [Sprint](#) e di [Daily Scrum](#).

A supporto dello sviluppo del progetto sono stati utilizzati i seguenti strumenti:

- * Gitlab: software open source per la gestione di repository [Git](#) e supporto alla Continuous Integration. Utile anche per l'assegnazione di piccole tasks e fissare scadenze, milestone e monitoraggio delle statistiche del progetto.
- * Slack:
- * Telegram:



- * WebStorm: potente IDE, modulare, che con l'aggiunta di plugin acquista funzionalità, come il conteggio delle ore di programmazione, scadenze e gestione del repository [Git](#) locale.



1.7 Tecnologie utilizzate

Capitolo 2

Processi e metodologie

Brevissima introduzione al capitolo

2.1 Processo sviluppo prodotto

Per

Capitolo 3

Descrizione dello stage

Breve introduzione al capitolo

3.1 Introduzione al progetto

3.2 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.

3.3 Requisiti e obiettivi

3.4 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 [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.

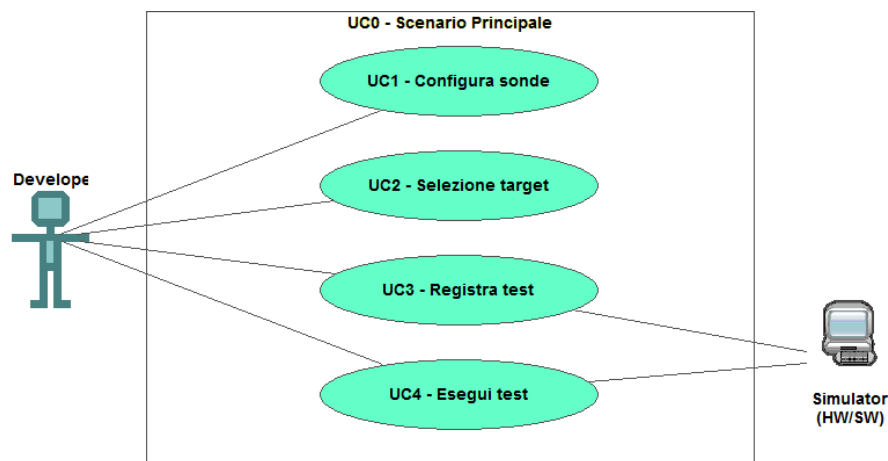


Figura 4.1: Use Case - UC0: Scenario principale

UC0: Scenario principale

Attori Principali: Sviluppatore applicativi.

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

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

Acronimi

BIG Best Intelligent Ground. [5](#)

Glossario

Agile FGGSDF . [3](#), [5](#)

api in informatica con il termine *Application Programming Interface API* (ing. interfaccia di programmazione di un'applicazione) si indica ogni insieme di procedure disponibili al programmatore, di solito raggruppate a formare un set di strumenti specifici per l'espletamento di un determinato compito all'interno di un certo programma. La finalità è ottenere un'astrazione, di solito tra l'hardware e il programmatore o tra software a basso e quello ad alto livello semplificando così il lavoro di programmazione. [1](#), [5](#)

Daily Scrum FGGSDF . [3](#), [5](#)

Git FGGSDF . [3-5](#)

Incrementale FGGSDF . [3](#), [5](#)

Java fsdf s sadf . [5](#)

Scrum FGGSDF . [3](#), [5](#)

Sprint FGGSDF . [3](#), [5](#)

UCIS È un'Associazione Nazionale di Volontariato, inserita nell'Albo istituito presso il Dipartimento di Protezione Civile. Raggruppa, tutela e coordina i Soccorritori Cinofili presenti sul Territorio Nazionale. [1](#), [5](#)

UCIS Report Tool Applicazione in dotazione attualmente a UCIS sviluppata 5 anni fa e pubblicata da Alternative Studio nel 2018. [5](#)

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. [5](#)

xml FGGSDF . [5](#)

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/>.