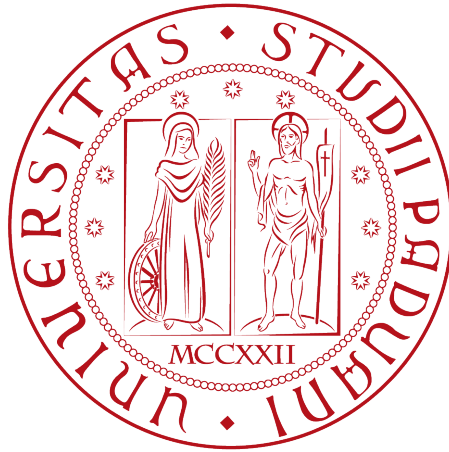


**Università degli Studi di Padova**

DIPARTIMENTO DI MATEMATICA

CORSO DI LAUREA IN INFORMATICA



**Realtà virtuale per colmare il divario tra  
e-commerce e negozio fisico**

*Tesi di laurea triennale*

*Relatore*

Prof. Tullio Vardanega

*Laureando*

Simone Magagna  
1009467

---

ANNO ACCADEMICO 2015-2016



Dedica



# Sommario

Questo documento rappresenta la relazione finale di stage tenuta presso l'azienda The White Dog s.r.l. della durata complessiva pari a 308 ore.

Il primo capitolo tratta dell'azienda ospitante, dei prodotti e dei servizi che offre, in particolare all'azienda Diana Corp. della quale rappresenta il reparto ricerca e sviluppo. Il secondo capitolo descrive le strategie aziendali riguardanti l'attività di stage, di come il mio progetto si integri in esse, del piano di lavoro propostomi e degli obiettivi aziendali e personali.

Il terzo capitolo parla dell'effettivo sviluppo del progetto, descrivendo le fasi di pianificazione e ricerca effettuate con il mio tutor aziendale e il team di sviluppo, per arrivare a trattare dell'analisi dei requisiti, dei principi di progettazione perseguiti, dello sviluppo di alcune delle parti più peculiari e interessanti del progetto e della verifica e validazione.

Il quarto ed ultimo capitolo analizza i risultati ottenuti confrontandoli con gli obiettivi prefissati, valuta le conoscenze acquisite e discute quali tra queste debbano essere integrate nel corso di studi. Il capitolo si conclude con alcune considerazioni di carattere personale.



# Ringraziamenti

In questa sezione ci saranno i ringraziamenti.





# Indice

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>The White Dog s.r.l.</b>                                   | <b>1</b>  |
| 1.1      | Chi è The White Dog s.r.l. . . . .                            | 1         |
| 1.2      | Prodotti e servizi . . . . .                                  | 1         |
| 1.3      | Processi interni . . . . .                                    | 2         |
| 1.4      | Strumenti e tecnologie . . . . .                              | 6         |
| 1.4.1    | Ambienti di sviluppo . . . . .                                | 6         |
| 1.4.2    | Gestione dei progetti . . . . .                               | 6         |
| 1.4.3    | Versionamento . . . . .                                       | 7         |
| 1.4.4    | Tecnologie di sviluppo . . . . .                              | 7         |
| 1.5      | Ricerca e innovazione . . . . .                               | 7         |
| <b>2</b> | <b>Il quadro strategico</b>                                   | <b>9</b>  |
| 2.1      | Strategie aziendali di stage . . . . .                        | 9         |
| 2.2      | Il progetto di stage proposto . . . . .                       | 9         |
| 2.2.1    | Piano di lavoro proposto . . . . .                            | 10        |
| 2.2.2    | Obiettivi aziendali . . . . .                                 | 11        |
| 2.2.3    | Obiettivi personali . . . . .                                 | 11        |
| <b>3</b> | <b>Il progetto di e-commerce VR</b>                           | <b>13</b> |
| 3.1      | Pianificazione del lavoro . . . . .                           | 13        |
| 3.2      | Ricerca e sperimentazione . . . . .                           | 13        |
| 3.3      | Tecnologie adottate . . . . .                                 | 13        |
| 3.4      | Analisi dei requisiti . . . . .                               | 13        |
| 3.4.1    | Caratteristiche degli utenti . . . . .                        | 13        |
| 3.4.2    | Casi d'uso . . . . .  | 13        |
| 3.4.3    | Requisiti . . . . .   | 14        |
| 3.5      | Progettazione . . . . .                                       | 14        |
| 3.5.1    | Portabilità dell'applicazione . . . . .                       | 14        |
| 3.5.2    | Usabilità dell'applicazione . . . . .                         | 14        |
| 3.5.3    | Costruzione della scena 3D . . . . .                          | 14        |
| 3.5.4    | Interazione con gli oggetti all'interno della scena . . . . . | 14        |
| 3.5.5    | Progettazione e integrazione con AWS API Gateway . . . . .    | 14        |
| 3.6      | Sviluppo . . . . .  | 14        |
| 3.6.1    | Sviluppo degli oggetti interattivi . . . . .                  | 14        |
| 3.6.2    | Creazione a runtime di oggetti interattivi . . . . .          | 14        |
| 3.6.3    | Dati persistenti attraverso le scene . . . . .                | 15        |
| 3.6.4    | Unity e il protocollo HTTP . . . . .                          | 15        |
| 3.6.5    | Creazione e parsing di oggetti JSON in Unity . . . . .        | 15        |

|          |  |           |
|----------|--|-----------|
| 3.7      | Verifica e validazione . . . . .   | 15        |
| <b>4</b> | <b>Analisi retrospettiva</b>   | <b>17</b> |
| 4.1      | Bilancio dei risultati rispetto agli obiettivi prefissati . . . . .        | 17        |
| 4.2      | Bilancio formativo . . . . .   | 17        |
| 4.3      | Analisi critica del rapporto formativo tra stage e corso di laurea . . . . | 17        |
| 4.4      | Valutazioni personali . . . . .  | 17        |
|          | <b>Glossario</b>   | <b>19</b> |
|          | <b>Bibliografia</b>  | <b>23</b> |

# Elenco delle figure

|     |  |    |
|-----|--|----|
| 1.1 | Logo dell'azienda The White Dog s.r.l. . . . .   | 1  |
| 1.2 | Metodologia di sviluppo Agile . . . . .  | 3  |
| 1.3 | Competenze necessarie alla metodologia di sviluppo DevOps . . . . .  | 4  |
| 1.4 | Metodologia di sviluppo software Extreme Programming . . . . .   | 5  |
| 1.5 | Esempio di <i>scrum board</i> all'interno del software Jira . . . . .  | 6  |
| 2.1 | Schema rappresentativo della differenze tra <i>single-channel</i> , <i>multi-channel</i> ,<br><i>cross-channel</i> e <i>omni-channel</i> . . . . . | 10 |

# Elenco delle tabelle





# Capitolo 1

## The White Dog s.r.l.

### 1.1 Chi è The White Dog s.r.l.

The White Dog s.r.l. è una realtà aziendale nata nel 2008 con sede a Torreglia, in provincia di Padova. Essa è stata fondata dal signor Stefano Mocellini, fondatore e CEO di Diana Corp.<sup>1</sup>, con la volontà di creare un *team* di lavoro focalizzato sulla ricerca e sviluppo.

The White Dog s.r.l. coordina e gestisce società tutte affini al settore *e-commerce*, come Diana Corp. e LiveStory<sup>2</sup>. L'azienda possiede un reparto di ricerca e sviluppo denominato R&D, il quale esplora nuove tecnologie da applicare poi alle società figlie nel caso di esito positivo o facendo nascere nuovi progetti separati.



figura 1.1: Logo dell'azienda The White Dog s.r.l.

### 1.2 Prodotti e servizi

Il principale servizio che l'azienda offre alle aziende che coordina e gestisce, come Diana Corp. e Live Story, è la ricerca e lo sviluppo di nuove tecnologie da applicare nell'ambito del *fashion e-commerce*. Essa svolge l'attività di *testing*<sub>G</sub> delle nuove tecnologie web disponibili, le valuta attentamente in termini di prestazioni e costi, per poi renderle disponibili. Ad essa oltretutto vengono commissionati progetti che Diana Corp., per competenze e tempistiche, non può portare a termine, come ad esempio applicazioni *mobile* legate agli *e-commerce* prodotti.

---

<sup>1</sup><http://www.dianacorp.com/>

<sup>2</sup><http://www.livestory.nyc/>

The White Dog s.r.l. inoltre ha creato il *concept* di *Live Story*, *social management system*<sub>G</sub> che gestisce contenuti *social* e li rende acquistabili, *concept* che è diventato azienda nel 2015 con sede a New York. *Live Story* colleziona foto degli utenti dei *social network* marcate con un particolare *hashtag* che rappresenta l'azienda che vuole utilizzare il servizio. Il sistema accoppia la foto ad un particolare prodotto presente nel catalogo e genera automaticamente le richieste di permesso di utilizzo della foto e la invia all'utente interessato. Se l'utente approva e il moderatore ritiene conforme la foto, l'azienda può utilizzare il contenuto nel proprio sito o *e-commerce*.

### 1.3 Processi interni

Lo sviluppo del software a The White Dog s.r.l. segue una metodologia tipicamente Agile<sup>3</sup>.

La Metodologia Agile si riferisce ad un insieme di metodi di sviluppo software fondati su principi comuni, direttamente o indirettamente derivati dai principi del *Manifesto per lo Sviluppo Agile di Software*:

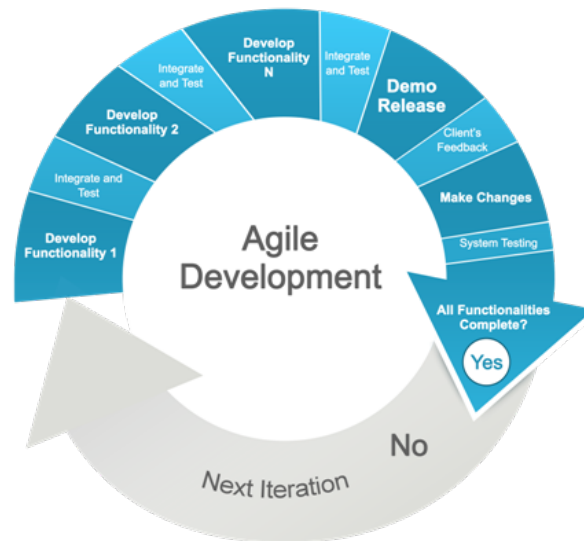
- \* **Gli individui e le interazioni** più che i processi e gli strumenti;
- \* **Il software funzionante** più che la documentazione esaustiva;
- \* **La collaborazione col cliente** più che la negoziazione dei contratti;
- \* **Rispondere al cambiamento** più che seguire un piano.

I metodi agili si contrappongono al modello a cascata e altri processi software tradizionali, proponendo un approccio meno strutturato e focalizzato sull'obiettivo di consegnare al cliente, in tempi brevi e frequentemente, software funzionante e di qualità. Fra le pratiche promosse dai metodi agili ci sono la formazione di team di sviluppo piccoli, cross-funzionali e auto-organizzati, lo sviluppo iterativo e incrementale, la pianificazione adattiva e il coinvolgimento diretto e continuo del cliente nel processo di sviluppo.

---

<sup>3</sup><http://agilemanifesto.org/>





**figura 1.2:** Metodologia di sviluppo Agile

L'azienda The White Dog s.r.l. è composta da un *team* di sviluppo piccolo, nel quale ogni componente ha competenze sia software, sia decisionali che di controllo della qualità.

Essa dà molta importanza agli individui presenti e all'interazione tra di loro, eliminando completamente la gerarchia lavorativa classica e impegnandosi molto per mantenere un ambiente lavorativo di reciproco rispetto.

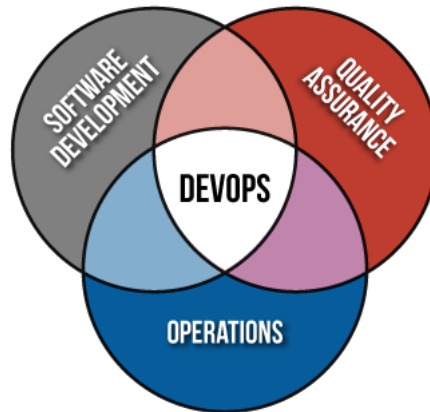
La collaborazione col cliente avviene in maniera assidua e giornaliera tramite riunioni nell'ufficio R&D o trasferte. Se entrambe le soluzioni non sono attuabili, vengono optate sessioni di video conferenze.

Infine la natura dell'azienda la obbliga a rispondere al cambiamento in maniera molto veloce e repentina, dovendo pianificare così molti cicli di *refactoring*<sub>G</sub> per lo sviluppo software.

Le tre principali metodologie di sviluppo, derivanti da quella Agile, che l'azienda adotta sono:

### DevOps

Metodologia di sviluppo software che punta alla comunicazione, collaborazione e integrazione tra gli sviluppatori e addetti alle *operations*<sub>G</sub> dell'*information technology*. DevOps vuole rispondere all'interdipendenza tra sviluppo software e IT *operations*<sub>G</sub>, puntando ad aiutare un'organizzazione a sviluppare in modo più rapido ed efficiente prodotti e servizi.

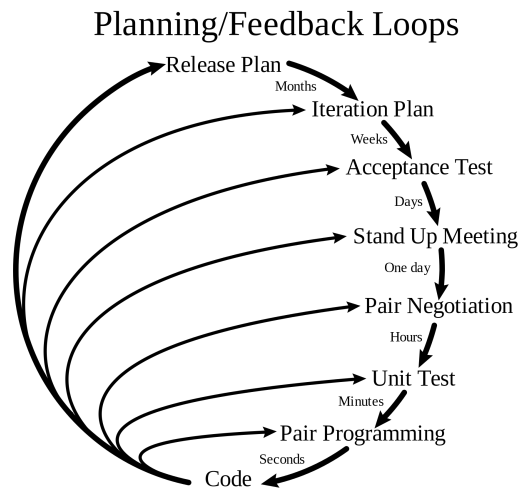


**figura 1.3:** Competenze necessarie alla metodologia di sviluppo DevOps

In The White Dog s.r.l. questo principio è concretizzato dal fatto che ogni membro possiede sia le competenze di sviluppo, sia amministrative che di controllo della qualità, migliorando così di molto l'efficienza e l'agilità nello sviluppo del software e nel suo rilascio.

### **Extreme Programming**

Metodologia di sviluppo software che enfatizza la scrittura di codice di qualità e la rapidità di risposta ai cambiamenti di requisiti. Prescrive lo sviluppo iterativo e incrementale, soprattutto in brevi cicli di sviluppo. Suggerisce inoltre l'uso sistematico di *unit testing*<sub>G</sub> e *refactoring*<sub>G</sub>, vietando ai programmatori di sviluppare codice non strettamente necessario. Sostiene la chiarezza e la semplicità del codice, preferisce strutture gestionali non gerarchiche e dà molta importanza alla comunicazione diretta e frequente fra sviluppatori e cliente e fra gli sviluppatori stessi.



**figura 1.4:** Metodologia di sviluppo software Extreme Programming

Il *team* di sviluppo di The White Dog s.r.l. fa ampio utilizzo di questa metodologia, spingendo molto sulla semplicità del codice prodotto, che dovrà poi essere utilizzato dagli sviluppatori Diana Corp. e Live Story, e sulla giornaliera comunicazione diretta tra gli sviluppatori e con i loro principali clienti. Questa comunicazione è facilitata dal fatto che The White Dog s.r.l. ha sede nello stesso stabilimento di Diana Corp..

### Scrum

*Framework*<sub>G</sub> agile di sviluppo software, iterativo ed incrementale, concepito per gestire progetti e prodotti software. Esso enfatizza tutti gli aspetti di gestione di progetto legati a contesti in cui è difficile pianificare in anticipo. Vengono utilizzati meccanismi propri di un processo di controllo empirico, in cui i cicli di *feedback*, che ne costituiscono le tecniche di *management* fondamentali, risultano in opposizione alla gestione basata sul concetto tradizionale di *command-and-control*<sub>G</sub>. Il suo approccio alla pianificazione e gestione dei progetti è quello di portare l'autorità decisionale al livello di proprietà e certezze operative.

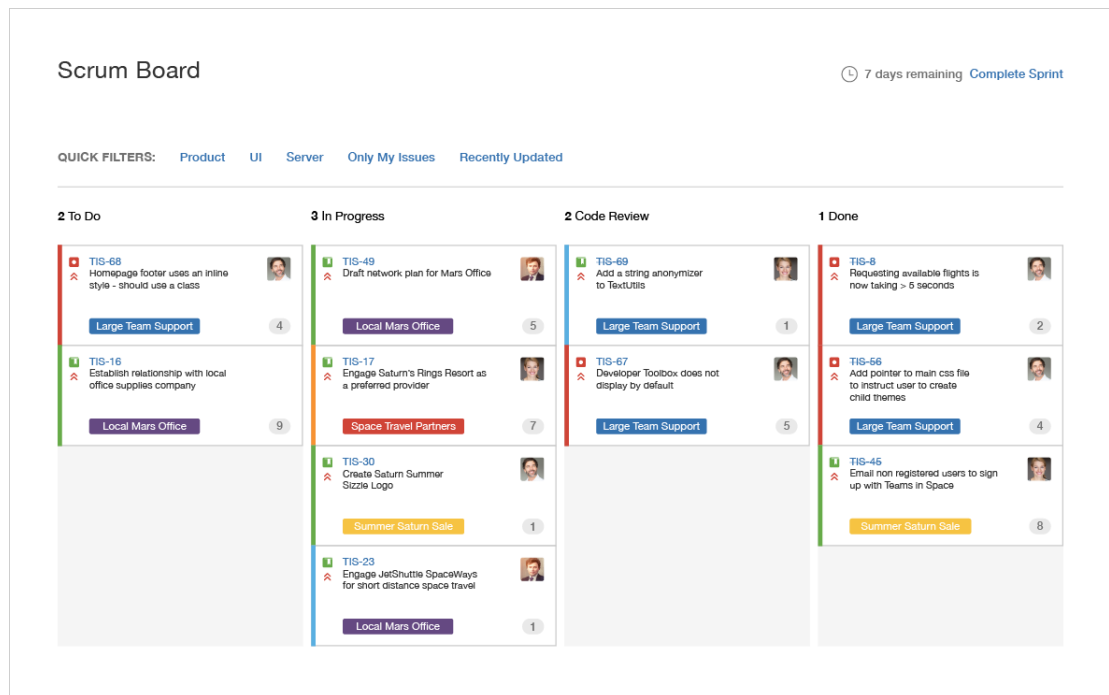


figura 1.5: Esempio di *scrum board* all'interno del software Jira

The White Dog s.r.l. sfrutta questa metodologia di sviluppo software utilizzando ampiamente strumenti di *project management* che supportano il metodo Scrum, come Wrike<sup>4</sup> e Jira<sup>5</sup>.

## 1.4 Strumenti e tecnologie

### 1.4.1 Ambienti di sviluppo

Il sistema operativo adottato dall'azienda è Mac OS X installato su macchine iMac. L'ambiente di sviluppo, data la natura aziendale, non è standardizzato, ma varia in base al prodotto in fase di sviluppo, che può cambiare in maniera repentina.

### 1.4.2 Gestione dei progetti

I due principali strumenti utilizzati da The White Dog s.r.l. per il *project management* e l'*issue tracking* sono rispettivamente Wrike e Jira.

Wrike, sviluppato dall'omonima casa, è uno strumento per la collaborazione e il *project management*. Permette ai suoi utenti di modificare progetti, classificare le attività per importanza, tenere traccia dei programmi e collaborare con altri utenti dello stesso gruppo.

Jira, prodotto dall'azienda Atlassian, è un software di *bug tracking*, *issue tracking* e *project management*. Esso permette di tenere traccia delle azioni e dei problemi degli

<sup>4</sup><https://www.wrike.com/it/it/>

<sup>5</sup><https://www.atlassian.com/software/jira>

utenti, di distribuire i compiti all'interno del *team*, discutere del lavoro in atto con una visibilità completa e migliorare le prestazioni della squadra visualizzando dati in tempo reale.

### 1.4.3 Versionamento

Il principale software di controllo di versione distribuito utilizzato dall'azienda è Git<sup>6</sup>. Git supporta lo sviluppo non lineare con diramazione e fusioni rapide e continue e comprende strumenti specifici per visualizzare e navigare una cronologia di sviluppo non lineare. Permette ad ogni sviluppatore una copia locale dell'intera cronologia di sviluppo e le modifiche vengono importate da un *repository*<sub>G</sub> ad un altro. I *repository*<sub>G</sub> possono essere pubblicati facilmente tramite protocolli HTTP, FTP, SSH, RSYNC o uno speciale protocollo git.

### 1.4.4 Tecnologie di sviluppo

Vista la varietà delle ricerche e dei prodotti sviluppati dall'azienda The White Dog s.r.l., le tecnologie di sviluppo sono sempre in continua evoluzione e cambiamento. Le principali sono comunque Java, JavaScript, Node.js, MongoDB, PHP, HTML, CSS e AWS.

## 1.5 Ricerca e innovazione

R&D rappresenta il reparto di ricerca e sviluppo dell'azienda The White Dog s.r.l.. Ha a disposizione diversi dispositivi per la ricerca come *smartphone* di ultima generazione, *Smart TV*, *smartwatch* e numerosi dispositivi per lo sviluppo *AR*<sub>G</sub> e *VR*<sub>G</sub> come *Google Glass*, *Oculus Rift Development Kit 2*, *Google Cardboard* e *Leap Motion*. Attraverso questi dispositivi l'azienda studia e sviluppa nuove modalità di interazione che l'utente finale può utilizzare nell'acquisto nei propri *stores* digitali.

---

<sup>6</sup><https://git-scm.com/>



## Capitolo 2

# Il quadro strategico

### 2.1 Strategie aziendali di stage

L'azienda The White Dog s.r.l. accoglie e offre l'attività di stage per due principali motivi:

- \* **Sperimentazione su progetti innovativi:** data la forte propensione alla ricerca dell'azienda, esistono numerosi campi che essa vorrebbe esplorare ma che a causa di altri progetti più prioritari e scarsità di tempo non può studiare. Offre quindi allo studente universitario un progetto di ricerca e sviluppo su tecnologie innovative ed interessanti, non pretendendo alcun risultato da subito inseribile nel mercato. Questo permette allo studente di vivere l'esperienza dello stage in piena libertà e serenità, riuscendo così a portare un notevole valore aggiunto personale che l'azienda è ben felice di accogliere;
- \* **Valutazione dello stagista:** l'azienda è in continua crescita e necessita di nuovo personale preparato e soprattutto capace di lavorare in costante sintonia col gruppo. Lo stage universitario permette all'azienda di scoprire persone che soddisfano questi due importanti requisiti per una futura assunzione.

Da parte sua The White Dog s.r.l. offre molto agli stagisti. I tutor aziendali supportano lo studente per tutto il periodo lavorativo, consigliandolo sia per quanto riguarda il piano di lavoro, sia sulle tecnologie da utilizzare sia effettuando proficue discussioni in vista della relazione finale. Allo studente viene offerto un ambiente di lavoro accogliente e strumenti aggiornati e all'avanguardia, supportandolo anche economicamente prevedendo un rimborso spese.

### 2.2 Il progetto di stage proposto

Il progetto propostomi nasce dalla costante volontà aziendale di ricercare nuove metodologie di interazione da proporre agli utenti dei suoi *e-commerce* in ottica *omni-channel*<sup>6</sup>. Lo studio sulle nuove tecnologie e strumentazioni presenti sul mercato, ha portato l'azienda a considerare la realtà virtuale un terreno interessante e degno di studio. Nasce così l'idea di un *e-commerce VR*, progetto in grado di colmare, in parte, quel divario che da sempre ha distanziato *store* virtuale e negozio fisico.

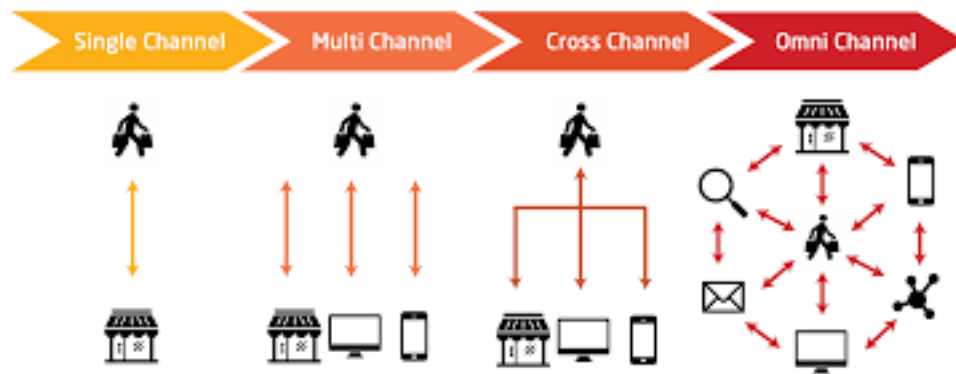


figura 2.1: Schema rappresentativo della differenze tra *single-channel*, *multi-channel*, *cross-channel* e *omni-channel*

L'obiettivo di stage è, dunque, un'esplorazione tecnologica nel campo della *virtual reality*<sub>G</sub>. Il progetto, diviso in due parti, mira ad arrivare ad un prototipo di *virtual showroom* dove poter esplorare ed interagire con i prodotti e permetterne l'acquisto. La prima parte è relativa alla progettazione e realizzazione del movimento in uno spazio 3D ed interazione con gli oggetti. La seconda parte tratta invece la progettazione e realizzazione in un'interfaccia di presentazione del prodotto, con integrazione al processo di acquisto mediante l'uso di sistemi *cloud* esterni.

### 2.2.1 Piano di lavoro proposto

#### Piano temporale

In accordo col tutor aziendale, la durata massima dello stage è stata fissata a 320 ore, divise in 8 settimane lavorative di 5 giorni, 8 ore al giorno. Il piano lavorativo è stato dunque pianificato nel seguente modo:

- \* **Settimana 1:** analisi dei requisiti funzionali del sistema da sviluppare. Studio delle tecnologie e linguaggi disponibili riguardanti la realtà aumentata;
- \* **Settimana 2:** scelta dell'hardware da utilizzare in base ai requisiti. Scelta del *framework*<sub>G</sub> di sviluppo e primo prototipo di una scena 3D;
- \* **Settimana 3:** raffinamento della scena 3D, progettazione e sviluppo di oggetti e comportamento di essi nello spazio 3D. Primo prototipo di *user interaction*;
- \* **Settimana 4:** progettazione e sviluppo integrazione tra sistema *VR*<sub>G</sub> e *e-commerce*. Progettazione di *user interaction* per la fruizione dei contenuti provenienti dall'*e-commerce*;
- \* **Settimana 5:** approfondimento di *user interaction* e del comportamento degli oggetti nell'ambiente virtuale;
- \* **Settimana 6:** studio e prototipazione del possibile processo d'acquisto all'interno dell'ambiente virtuale;



- \* **Settimana 7:** Conclusione del prototipo e della relativa documentazione;
- \* **Settimana 8:** studio del modello emergente di *omni-channel<sub>G</sub>* e come la realtà virtuale e la realtà aumentata possono estendere questo modello.

### Piano metodologico

Assieme al tutor aziendale, abbiamo fin da subito concordato la mia presenza durante l'orario d'ufficio, permettendo così un'interazione intensa e costante. Il lavoro di ricerca e sviluppo che ho effettuato è stato totalmente autonomo, con giornaliere interazioni con il personale solo per raccogliere e analizzare la documentazione, requisiti e *feedback* sull'andamento del progetto. Le revisioni di progetto sono avvenute secondo la seguente metodologia:

- \* Riunione breve di 15 minuti ogni mattina;
- \* Riunione di 1 ora alla fine di ogni settimana come analisi retrospettiva.

### Piano tecnologico

In questa sezione descriverò lo stack tecnologico inizialmente propostomi dall'azienda e di come si sia evoluto nel tempo dopo le attività di ricerca.

#### 2.2.2 Obiettivi aziendali

In questa sottosezione elencherò gli obiettivi che l'azienda si pone di raggiungere con il mio stage.

#### 2.2.3 Obiettivi personali

In questa sottosezione tratterò degli obiettivi personali e delle motivazioni che mi hanno spinto a scegliere questo stage e questo progetto.



## Capitolo 3

# Il progetto di e-commerce VR

In questo capitolo andrò a trattare nel dettaglio tutte le fasi dello stage che hanno portato allo sviluppo del progetto.

### 3.1 Pianificazione del lavoro

In questa sezione tratterò della pianificazione del lavoro effettuata assieme al mio tutor, delle fasi che l'hanno caratterizzata e del ciclo di vita adottato.

### 3.2 Ricerca e sperimentazione

In questa sezione descriverò la fase di ricerca e sperimentazione delle tecnologie utilizzate, inizialmente a me sconosciute. Ho deciso di dedicare una sezione a questa fase perché ha avuto una rilevante importanza all'interno del mio stage e rappresenta uno dei principali obiettivi aziendali.

### 3.3 Tecnologie adottate

In questa sezione descriverò come le ricerche e le sperimentazioni effettuate mi hanno portato a scegliere un particolare stack tecnologico.

### 3.4 Analisi dei requisiti

All'interno di questa sezione tratterò dell'attività di analisi dei requisiti che il team ha effettuato prima della progettazione e dello sviluppo del progetto.

#### 3.4.1 Caratteristiche degli utenti

In questa sottosezione descriverò le tipologie di utenti previsti per l'applicazione.

#### 3.4.2 Casi d'uso

In questa sottosezione elencherò tutti i casi d'uso previsti per l'applicazione.

### 3.4.3 Requisiti

In questa sottosezione elencherò tutti i requisiti estratti che l'applicazione soddisfa.

## 3.5 Progettazione

In questa sezione andrò a descrivere le più importanti fasi di progettazione.

### 3.5.1 Portabilità dell'applicazione

In questa sottosezione tratterò di come la progettazione del software sia stata ampiamente influenzata dalla volontà di portabilità dell'applicazione su tutti i dispositivi VR Android.

### 3.5.2 Usabilità dell'applicazione

In questa sottosezione descriverò gli studi effettuati riguardo l'usabilità dell'applicazione VR e delle scelte di progettazione che hanno portato tali studi.

### 3.5.3 Costruzione della scena 3D

In questa sezione andrò a descrivere le fasi di progettazione della scena 3D presente nell'applicazione e visibile tramite dispositivo VR.

### 3.5.4 Interazione con gli oggetti all'interno della scena

All'interno di questa sottosezione parlerò della progettazione riguardante le modalità di interazione tra il visore VR e gli oggetti presenti all'interno della scena.

### 3.5.5 Progettazione e integrazione con AWS API Gateway

All'interno di questa sezione tratterò della progettazione riguardante l'API Mock creata tramite AWS API Gateway e della sua integrazione con l'applicazione.

## 3.6 Sviluppo

In questa sezione andrò a descrivere in dettaglio lo sviluppo delle più significative e peculiari funzionalità dell'applicazione.

### 3.6.1 Sviluppo degli oggetti interattivi

In questa sottosezione descriverò come si costruiscono degli oggetti interattivi in Unity per i dispositivi VR.

### 3.6.2 Creazione a runtime di oggetti interattivi

In questa sottosezione tratterò della creazione a runtime di oggetti interattivi in Unity.

### **3.6.3 Dati persistenti attraverso le scene**

In questa sezione spiegherò come si costruiscono oggetti persistenti che vivono attraverso le scene.

### **3.6.4 Unity e il protocollo HTTP**

In questa sottosezione parlerò di come Unity si integri con il protocollo HTTP.

### **3.6.5 Creazione e parsing di oggetti JSON in Unity**

In questa sottosezione parlerò di come si creino e si manipolino oggetti JSON in Unity.

## **3.7 Verifica e validazione**

All'interno di questa sezione parlerò della fase di verifica e validazione effettuata per questo progetto.



## Capitolo 4

# Analisi retrospettiva

In questo capitolo analizzerò i risultati ottenuti confrontandoli con gli obiettivi prefissati, le conoscenze acquisite e quali tra queste ritengo debbano essere integrate nel corso di laurea.

### 4.1 Bilancio dei risultati rispetto agli obiettivi prefissati

In questa sezione analizzerò i risultati ottenuti e li confronterò con gli obiettivi che l'azienda si era prefissata di ottenere.

### 4.2 Bilancio formativo

In questa sezione analizzerò le conoscenze, le abilità e le competenze apprese durante l'attività di stage.

### 4.3 Analisi critica del rapporto formativo tra stage e corso di laurea

In questa sezione discuterò quali conoscenze, apprese durante lo stage, ritengo debbano essere integrate nel corso di laurea.

### 4.4 Valutazioni personali

In questa sezione effettuerò delle valutazioni personali riguardo al progetto e allo stage.





# Glossario

## A

- \* **Augment reality:** arricchimento della percezione sensoriale umana mediante informazioni, in genere manipolate e convogliate elettronicamente, che non sarebbero percepibili con i cinque sensi.

## B

- \* **Bug tracking:** applicativo software usato generalmente dai programmatori per tenere traccia delle segnalazioni di bug all'interno dei software, in modo che tali errori siano mantenuti sotto controllo, con una descrizione della riproducibilità e dei dettagli ad essi correlati.

## C

- \* **Command-and-control:** principio di *management* dove si afferma il mantenimento dell'autorità in un processo decisionale distribuito.

## D

## E

## F

- \* **Framework:** architettura logica di supporto (spesso un'implementazione logica di un *design pattern*) su cui un software può essere progettato e realizzato.

## G

## H

## I

- \* **Issue tracking:** pacchetto software che gestisce e mantiene liste di problemi in maniera organizzata.

**J****K****L****M****N****O**

- \* **Operations:** funzioni di un'impresa coinvolte nella messa a disposizione per il cliente di un determinato prodotto o servizio.

**P****Q****R**

- \* **Refactoring:** tecnica strutturata per modificare la struttura interna di porzioni di codice senza modificarne il comportamento esterno, applicata per migliorare alcune caratteristiche non funzionanti del software.
- \* **Repository:** è un ambiente di un sistema informativo, in cui vengono gestiti i metadati, attraverso tabelle relazionali; l'insieme di tabelle, regole e motori di calcolo tramite cui si gestiscono i metadati prende il nome di metabase.

**S**

- \* **Social management system:** software che permette la gestione dei propri social network, collezionando contenuti presenti in essi o interagendovi automaticamente effettuando operazioni desiderate.

**T**

- \* **Testing:** il *software testing* è un'attività di investigazione condotta per fornire alle parti interessate informazioni sulla qualità del prodotto o del servizio in prova.

**U**

- \* **Unit testing:** per test di unità si intende l'attività di *testing* di singole unità software. Per unità si intende il minimo componente di un programma dotato di funzionamento autonomo.

**V**

- \* **Virtual reality:** realtà simulata attraverso dispositivi elettronici, come visori, cuffie e sensori di movimento.

**W****X****Y****Z**



# Bibliografia