

Università degli Studi di Padova

DIPARTIMENTO DI MATEMATICA "TULLIO LEVI-CIVITA"

CORSO DI LAUREA IN INFORMATICA



Progettazione e sviluppo dell'applicazione
per la gestione amministrativa di VIC SRL

Tesi di laurea triennale

Relatore

Dott. Silvia Crafa

Laureando

Luca Allegro

ANNO ACCADEMICO 2017-2018

Resentment is like taking poison and waiting for the other person to die.

– Malachy McCourt

Dedicato ad Alessio e nonno Lorenzo

Sommario

Il presente documento descrive il lavoro svolto durante il periodo di stage dal laureando Luca Allegro presso l'azienda VIC s.r.l.

Il tirocinio si è tenuto presso il Dipartimento di Ricerca e Sviluppo (R&D) durante i mesi di luglio e agosto 2018 e ha avuto la durata di 310 ore.

L'obiettivo dello stage è stato lo sviluppo di un'applicazione web per la gestione amministrativa dell'azienda che comprende il tenere traccia dei movimenti bancari e la gestione delle fatture attive e delle fatture passive.

L'elaborato ha lo scopo di illustrare:

- il contesto aziendale dove è stato svolto lo stage ([Capitolo 1](#));
- la descrizione del progetto sviluppato ([Capitolo 2](#));
- le attività svolte durante esso ([Capitolo 3](#));
- una valutazione finale sull'esperienza e le competenze acquisite([Capitolo 4](#)).

Ringraziamenti

Innanzitutto, vorrei esprimere la mia gratitudine alla Professoressa Silvia Crafa, relatrice della mia tesi, per la l'attenzione e l'aiuto fornitomi durante la stesura del lavoro.

Desidero ringraziare con affetto la mia famiglia per essermi stata vicina e per avermi dato fiducia durante tutti gli anni di studio.

Ringrazio l'Associazione Italiana Arbitri per tutte le persone che mi ha permesso di conoscere e per le infinite occasioni di crescita e confronto che mi ha offerto.

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

Ringrazio tutte le altre persone a me care che conosco e ho conosciuto e che hanno contribuito, anche solo per un momento, a farmi diventare la persona che sono oggi.

Padova, Dicembre 2018

Luca Allegro

Indice

1	L'Azienda	1
1.1	Chi è VIC	1
1.2	Il Dipartimento di Ricerca e Sviluppo	2
1.3	Opportunità di Stage	2
2	Il Progetto	3
2.1	L'applicativo	3
2.1.1	Fatture attive	3
2.1.2	Fatture passive	4
2.1.3	Banche	4
2.2	Tecnologie Adottate	5
2.3	Strumenti	5
3	Svolgimento	7
3.1	Pianificazione	7
3.2	Studio soluzione esistente	8
3.3	Analisi dei Requisiti	8
3.3.1	Casi d'uso	8
3.3.2	Tracciamento dei requisiti	8
3.4	Progettazione	10
3.5	Realizzazione	10
3.6	Verifica e Validazione	10
3.6.1	Metriche	10
3.6.2	Risultati	10
3.7	Risultato	10
3.7.1	Modifiche Migliorative	10
4	Analisi retrospettiva	11
4.1	Considerazioni sulle Tecnologie	11
4.2	Considerazioni sugli Strumenti	11
4.3	Esperienza	11
4.4	Valutazioni sul rapporto stage-laurea	11
4.5	Valuazioni sulla Laurea Triennale	11
A	Appendice A	13
	Glossario	15

Acronimi	17
Bibliografia	19

Elenco delle figure

1.1	Il logo di VIC	1
3.1	Use Case - UC0: Scenario principale	8

Elenco delle tabelle

3.1	Tabella del tracciamento dei requisiti funzionali	10
3.2	Tabella del tracciamento dei requisiti qualitativi	10
3.3	Tabella del tracciamento dei requisiti di vincolo	10

1 L'Azienda

Questo capitolo descrive la realtà aziendale e il suo ambito di interesse

1.1 Chi è VIC

VIC s.r.l.[1] offre servizi di ispezione delle merci in tutto il mondo per individuare eventuali non conformità delle merci al fine di tutelare i propri clienti e ridurre al minimo le perdite.

VIC nasce a Venezia nel 2006 ed ora è una delle più grandi compagnie del mercato globale in quanto è presente con 22 diverse aziende e più di 50 laboratori in tutti i più grandi snodi di scambio del mondo.

La chiave del successo di VIC s.r.l. è l'attenzione che pone alla riduzione del tempo tra ispezione e reporting al cliente. Questo è stato possibile grazie all'automazione e alla semplificazione di molte attività svolte dai dipendenti mediante moderni sistemi informatici e allo sviluppo di diversi servizi web ed applicazioni mobile con l'obiettivo di fornire più informazioni possibili (inclusi foto e video) ai clienti.



Figura 1.1: Il logo di VIC

1.2 Il Dipartimento di Ricerca e Sviluppo

Il Dipartimento di Ricerca e Sviluppo (abbreviato: R&D) ha sede a Padova e si occupa di realizzare e mantenere tutti gli strumenti che l'azienda offre, ad esempio:

- il portale *VIC Online* per i clienti, mediante cui è possibile controllare lo stato dei propri ordini, ottenere informazioni utili, immagini e video riguardanti la merce ordinata;
- le applicazioni rivolte agli ispettori per raccogliere i dati sulla merce e realizzare dei resoconti sulle analisi effettuate e lo stato della merce.

Il dipartimento R&D è alla continua ricerca di metodi di innovazione per velocizzare, automatizzare e standardizzare le attività svolte dall'azienda nell'ambito del controllo qualità e quantità delle merci.

Tutti i servizi sviluppati dal dipartimento R&D sono composti dalle componenti front-end (client) e back-end (server). La comunicazione tra queste due componenti avviene utilizzando le quattro operazioni CRUD (Create, Read, Update, Delete) di HTTP, in modo da essere conformi ai principi [REST](#) (REpresentational State Transfer).

1.3 Opportunità di Stage

VIC s.r.l. è favorevole ad ospitare laureandi per lo stage perché crede che questa attività porti molteplici benefici all'azienda, nello specifico:

- il confronto con persone nuove e la loro esperienza può offrire spunti di miglioramento per i sistemi e le metodologie;
- per lo studio di nuove tecnologie e sul loro possibile utilizzo in progetti aziendali;
- la conoscenza di possibili candidati per l'assunzione.

2 Il Progetto

Descrizione del progetto di stage, motivazioni, tecnologie interessate, obiettivi.

2.1 L'applicativo

L'azienda utilizza un software gestionale ricco di funzionalità, ma molto datato. Esso infatti è stato sviluppato interamente in [PHP](#)[2] ed è basato sul framework [QCubed](#)[3] che sta per concludere il suo ciclo di vita in quanto è supportato solo da [PHP](#) 5.6, una versione ormai sorpassata da [PHP](#) 7 che lo ha sostituito come pacchetto predefinito nel sistema operativo Ubuntu a partire dalla versione 16.04, che vincola il mantenimento del sistema operativo 14.04.

Lo scopo dello stage è stato quello di sviluppare un nuovo gestionale, con le stesse funzionalità, ma basato su tecnologie moderne, prestando cura anche al rinnovamento dell'interfaccia grafica.

Nello specifico il software da realizzare rappresenta il Modulo Amministrativo: esso è uno dei moduli principali in cui è diviso il software di gestione in uso attualmente presso l'azienda. Entra in gioco subito dopo il modulo operativo e si occupa di gestire tutti gli aspetti contabili degli ordini, una volta che questi sono portati a compimento dallo staff. Si divide a propria volta in tre sottomoduli riguardanti le fatture attive, le fatture passive e la gestione dei conti bancari.

2.1.1 Fatture attive

Il modulo delle fatture attive si occupa di tenere traccia di tutti gli ordini completati e in attesa di fatturazione, della creazione della fattura stessa con la possibilità di specificare i parametri richiesti (data fattura, scadenza, tipo fattura...), della possibilità di inviare le fatture al cliente e la gestione dello scadenziario delle fatture scadute e non ancora incassate.

Dovrà offrire le seguenti funzionalità:

- R1: lista fatture da creare;
- R2: creazione nuova fattura;
- R3: modifica fattura esistente non ancora incassata;
- R4: rimozione fattura esistente non ancora incassata;
- R5: lista fatture create non ancora inviate al cliente;
- R6: invio fattura al cliente tramite email;
- R7: scadenziario fatture emesse da incassare
 - R7.1: già scadute;
 - R7.2: in scadenza.

2.1.2 Fatture passive

Il modulo delle fatture passive si occupa di tenere traccia di tutte le passività relative all'attività. Si dividono in gruppi: dirette e indirette. Le fatture passive dirette rappresentano delle passività direttamente riconducibile ad un preciso ordine (fornitori, agenzie, laboratori, spedizionieri), mentre le indirette rappresentano dei costi generici di gestione (elettricità, bollette telefoniche, affitti, . . .). Le fatture passive indirette si suddividono a propria volta in due categorie: una tantum e cicliche.

Dovrà offrire le seguenti funzionalità:

- R1: Lista passive dirette non pagate raggruppate per fornitore
 - R1.1: visione lista fatture per ogni fornitore
 - R1.2: visione dettaglio singola fattura
 - R1.3: modifica fattura
 - R1.4: eliminazione fattura
- R2: Lista passive indirette (divise tra cicliche e una tantum)
 - R2.1: visione lista fatture per ogni fornitore
 - R2.2: visione dettaglio singola fattura
 - R2.3: modifica fattura
 - R2.4: eliminazione fattura

2.1.3 Banche

Il modulo banche si occupa di tenere sotto controllo la situazione contabile dei vari conti correnti aziendali.

Dovrà offrire le seguenti funzionalità:

- R1: Visione saldi conti correnti banche
- R2: Visualizzazione fido di cassa
- R3: Visualizzazione saldo reale e saldo disponibile

2.2 Tecnologie Adottate

2.3 Strumenti

3 Svolgimento

Il presente capitolo descrive le varie fasi dello sviluppo dell'applicazione

3.1 Pianificazione

Come modello per la gestione del progetto ho adottato la metodologia [Agile](#). Tale modello mi ha permesso di adottare una certa flessibilità e velocità nel rispondere alle esigenze e ai feedback del Dipartimento di Amministrazione.

Il modello Agile prevede iterazioni continue, della durata massima di due settimane, entro le quali si susseguono attività di:

1. analisi dei requisiti che emergono dall'interazione con il cliente, in questo caso Wintech SpA;
2. pianificazione delle funzionalità da includere nello sprint, suddivisione in task e loro assegnazione ai membri del team;
3. progettazione delle funzionalità da implementare nell'iterazione corrente;
4. codifica e sviluppo delle funzionalità previste;
5. rilascio;
6. monitoraggio continuo e verifica dello stato dell'applicazione.

Il punto di partenza di ogni ciclo è il risultato raggiunto con il precedente e il feedback ricevuto tramite la presentazione della soluzione raggiunta fin'ora agli stakeholder.

Ho pianificato uno sprint della durata di 2 settimane per ogni sottomodulo da realizzare:

TODO dettaglio pianificazione:

3.2 Studio soluzione esistente

3.3 Analisi dei Requisiti

3.3.1 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.

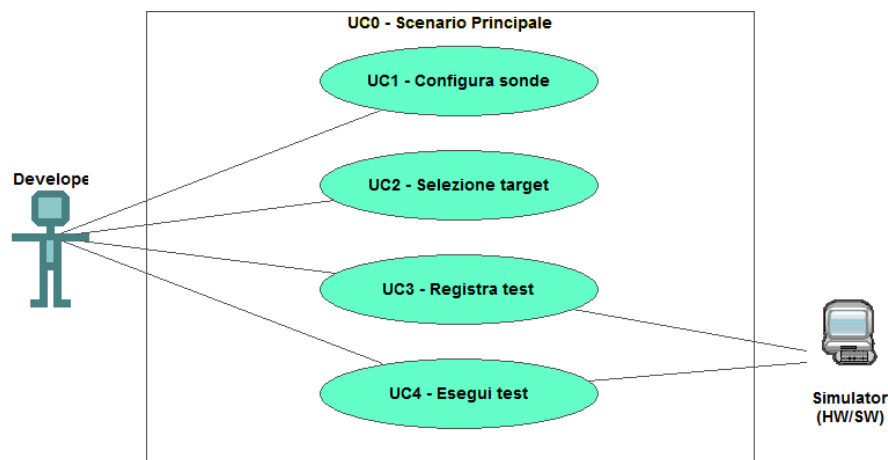


Figura 3.1: Use Case - UC0: Scenario principale

UC0: Scenario principale

Attori Principali: Sviluppatore applicativi.

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

Descrizione: La finestra di simulazione mette a disposizione i comandi per configurare, registrare o eseguire un test.

Postcondizioni: Il sistema è pronto per permettere una nuova interazione.

3.3.2 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 [3.1](#), [3.2](#) e [3.3](#) sono riassunti i requisiti e il loro tracciamento con gli use case delineati in fase di analisi.

Tabella 3.1: 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 3.2: 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	-

3.4 Progettazione

3.5 Realizzazione

3.6 Verifica e Validazione

3.6.1 Metriche

3.6.2 Risultati

3.7 Risultato

3.7.1 Modifiche Migliorative

Tabella 3.3: Tabella del tracciamento dei requisiti di vincolo

Requisito	Descrizione	Use Case
RVO-1	La libreria per l'esecuzione dei test automatici deve essere riutilizzabile	-

4 Analisi retrospettiva

Considerazioni sull'esperienza maturata durante lo stage e, più in generale, attraverso l'intero percorso formativo.

4.1 Considerazioni sulle Tecnologie

4.2 Considerazioni sugli Strumenti

Valutazioni sullo stage

4.3 Esperienza

4.4 Valutazioni sul rapporto stage-laurea

Analisi critica del rapporto formativo tra stage e corso di laurea

4.5 Valuazioni sulla Laurea Triennale

A Appendice A

A Glossario

Agile Insieme di principi per lo sviluppo software sotto i quali requisiti e soluzioni evolvono tramite gli sforzi collettivi del gruppo auto-organizzato polifunzionale. L'obiettivo di questo modello è soddisfare il cliente, rilasciando software in maniera continua e in periodi brevi, in modo che possa osservare l'andamento dello sviluppo e i risultati ottenuti. Di conseguenza è anche lecito accogliere eventuali cambiamenti dei requisiti durante l'intero ciclo di sviluppo.. [7](#), [15](#)

PHP Linguaggio di scripting interpretato, con licenza open source e libera, originariamente concepito per la realizzazione di pagine web dinamiche.. [3](#)

REST Si tratta di un tipo di architettura software per i sistemi di ipertesto distribuiti come il World Wide Web. Un concetto importante in REST è l'esistenza di risorse (fonti di informazioni), a cui si può accedere tramite un identificatore globale (un URI). Per utilizzare le risorse, le componenti di una rete (componenti client e server) comunicano attraverso una interfaccia standard (ad es. HTTP) e si scambiano rappresentazioni di queste risorse.. [2](#)

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.. [8](#)

A Acronimi

PHP [PHP: Hypertext Preprocessor](#). 15

REST [REpresentational State Transfer](#). 15

UML [Unified Modeling Language](#). 15

A Bibliografia

Riferimenti bibliografici

- [4] Daniel T. Jones James P. Womack. *Lean Thinking, Second Editon*. Simon & Schuster, Inc., 2010.
- [7] Ian Sommerville. *Software Engineering, 10th Edition*. Pearson, 2015.

Siti web consultati

- [1] VIC s.r.l. URL: <http://vicworldwide.com/> (cit. a p. 1).
- [2] PHP. URL: <http://www.php.net/> (cit. a p. 3).
- [3] Qcubed. URL: <http://qcubed.github.io/> (cit. a p. 3).
- [5] OpenBSD. URL: <http://www.math.unipd.it/~bellio/openbsd-michele-marchetto.pdf>.
- [6] Wikimedia. URL: https://wiki.wikimedia.it/wiki/Chi_siamo.
- [8] Manifesto Agile. URL: <http://agilemanifesto.org/iso/it/>.
- [9] Python Virtual Environment. URL: <http://docs.python-guide.org/en/latest/dev/virtualenvs/>.
- [10] Django. URL: <https://www.djangoproject.com/>.
- [11] Django REST Framework. URL: <http://www.django-rest-framework.org/>.
- [12] JSON. URL: <https://www.json.org/json-it.html>.
- [13] JSON Web Tokens. URL: <https://jwt.io/>.
- [14] Angular. URL: <https://angular.io/>.