

Università degli Studi di Padova

DIPARTIMENTO DI MATEMATICA "TULLIO LEVI-CIVITA"

CORSO DI LAUREA IN INFORMATICA



## Sviluppo di un modulo middleware per la gestione di lettori RFID

*Tesi di laurea*

*Relatore*

Prof. Tullio Vardanega

*Laureando*

Samuele Vanini

---

ANNO ACCADEMICO 2020-2021



Computer Science is no more about computers than astronomy is about telescopes.

— Edsger Wybe Dijkstra



# Sommario

Il presente documento descrive il lavoro svolto durante il periodo di stage, della durata di 320 ore, dal laureando Samuele Vanini presso l'azienda Aton S.p.A.

L'obiettivo principale del progetto di stage consisteva nella scrittura di un modulo software attuo a comandare un lettore RFID Kathrein in modo che questo potesse operare in autonomia e tramite l'interfacciamento con una piattaforma middleware proprietaria di Aton, chiamata da ora in poi AMP. Per fare questo, è stato necessario in primo luogo studiare parte della teoria riguardante la comunicazione RFID, per poi applicarla alle varie feature offerte dal lettore in esame. Il passo successivo ha riguardato la progettazione di un'architettura ad-hoc che andasse ad integrare la libreria proposta dal produttore con le API del middleware AMP.

L'intero documento sarà suddiviso nei quattro capitoli di seguito introdotti:

- \* **Primo capitolo:** presenta la realtà aziendale andando a descriverne alcune delle componenti chiave, come le tecnologie utilizzate ed il mercato di riferimento.
- \* **Secondo capitolo:** presenta il progetto nel suo insieme, andando a descriverne obiettivi e vincoli.
- \* **Terzo capitolo:** presenta l'architettura sviluppata per il modulo, illustrando le scelte implementative più rilevanti.
- \* **Quarto capitolo:** presenta una valutazione retrospettiva delle attività svolte andando ad analizzare il gap formativo tra il bagaglio di conoscenze proposte dal mondo universitario e quello richiesto dal progetto svolto.



# Ringraziamenti

*Padova, Dicembre 2021*

Samuele Vanini





# Indice

<b>1</b>	<b>Il contesto aziendale</b>	<b>1</b>
1.1	Storia e ambiti di interesse . . . . .	1
1.2	Organizzazione dei processi aziendali . . . . .	1
1.3	Tecnologie utilizzate . . . . .	3
1.3.1	Sistemi operativi . . . . .	3
1.3.2	Ambienti di sviluppo . . . . .	4
1.3.3	Sistemi di versionamento . . . . .	4
1.3.4	Suite per la produttività e comunicazione . . . . .	4
1.3.5	Linguaggi di sviluppo . . . . .	4
1.3.6	Framework . . . . .	5
1.4	Propensione all'innovazione . . . . .	5
<b>2</b>	<b>Il progetto Kathrein</b>	<b>7</b>
2.1	Descrizione . . . . .	7
2.2	Obiettivi . . . . .	7
2.2.1	Obiettivi aziendali . . . . .	7
2.2.2	Obiettivi personali . . . . .	8
2.3	Vincoli . . . . .	9
2.3.1	Vincoli tecnologici . . . . .	9
2.3.2	Vincoli temporali . . . . .	9
2.4	Futuro del progetto . . . . .	10
<b>3</b>	<b>Resoconto del progetto</b>	<b>11</b>
3.1	Pianificazione delle attività . . . . .	12
3.1.1	Comunicazioni . . . . .	12
3.1.2	Revisioni di progetto . . . . .	12
3.2	Analisi del progetto . . . . .	12
3.2.1	Funzionalità . . . . .	12
3.2.2	Requisiti chiave . . . . .	12
3.2.3	Casi d'uso . . . . .	12
3.2.4	Tecnologie coinvolte . . . . .	12
3.3	Sviluppo delle componenti critiche . . . . .	12
3.3.1	Design del modulo . . . . .	12
3.3.2	Interfacciamento con librerie native . . . . .	12
3.3.3	Ottimizzazione dei tempi di lettura . . . . .	12
3.3.4	Linguaggio di dominio specifico . . . . .	12
3.3.5	Controllo delle porte GPIO . . . . .	12
3.3.6	Tolleranza ai guasti . . . . .	12

3.4	Risultati raggiunti . . . . .	12
3.4.1	Requisiti soddisfatti . . . . .	12
3.4.2	Prodotti creati . . . . .	12
4	<b>Valutazione retrospettiva e conclusioni</b>	<b>13</b>
4.1	Obbiettivi raggiunti . . . . .	13
4.2	Progressione personale . . . . .	13
4.3	Gap formativo tra università e stage . . . . .	13
	<b>Glossary</b>	<b>15</b>
	<b>Acronyms</b>	<b>17</b>
	<b>Bibliografia</b>	<b>19</b>

## Elenco delle figure

1.1	Illustrazione del modello Agile - <b>fonte</b> <a href="http://urly.it/3g72a">urly.it/3g72a</a> . . . . .	3
1.2	Architettura con OSGi - <b>fonte</b> <a href="http://urly.it/3g72y">urly.it/3g72y</a> . . . . .	6
2.1	Schema di un sistema RFID - <b>fonte</b> <a href="http://urly.it/3gc8k">urly.it/3gc8k</a> . . . . .	8

## Elenco delle tabelle



# Capitolo 1

## Il contesto aziendale

*Nel corso di questo capitolo si presenta l'azienda in cui è stato svolto il progetto di stage, si illustrano i processi interni di questa e la sua posizione sul mercato. Vengono inoltre discusse le tecnologie utilizzate e la sua propensione nei confronti dell'innovazione.*

### 1.1 Storia e ambiti di interesse

L'azienda ospitante prende il nome di Aton S.P.A., società del Trevigiano di importante rilevanza per le sue relazioni commerciali intraprese con aziende leader italiane e numerose multinazionali. Operante in numerosi ambiti, Aton fornisce soluzioni e servizi IT, in linea con i principi Industry 4.0, per le vendite multicanale, le catene di negozi, la grande distribuzione e la gestione degli *asset* nei settori CPG (*Consumer Packaged Goods*), Retail, Fashion ed Energy. Proprio nel contesto di gestione degli *asset* vengono introdotte due delle applicazioni tecnologiche di punta di Aton, le applicazioni [Internet Of Things](#) e [Machine-to-machine](#).

Data la grande varietà di ambiti in cui l'azienda è impiegata, anche il team che la compone presenta un cospicuo numero di professionisti, all'incirca 150. Aggiungendo a questi anche i dipendenti operanti in una rete di società partecipate, si raggiungono circa le 200 persone. Queste società, specializzate per area geografica e per soluzioni verticali, sono: Blue Mobility e Aton White in Italia, mentre, per Spagna e Portogallo si trova Aton Allspark Iberica. Non è possibile infine non citare il proficuo clima lavorativo presente all'interno dell'azienda, che in aggiunta allo spirito di innovazione e alle numerose possibilità di crescita personale offerta dall'azienda stessa, ha permesso ad Aton di ricevere per il suo terzo anno consecutivo la certificazione "Great Place To Work", che la colloca di diritto nella classifica delle 50 migliori aziende per cui lavorare in Italia.

### 1.2 Organizzazione dei processi aziendali

L'azienda al fine di gestire al meglio i propri processi interni, segue rigidamente un elenco ben definito di processi. All'interno del reparto RFID questa pratica viene ritenuta di primaria importanza. Questo è dovuto dalla natura stessa del reparto, lavorando su diversi progetti cliente ognuno differente dall'altro, una struttura solida con cui ogni sviluppatore possa orientarsi è l'unica possibilità per distribuire prodotti

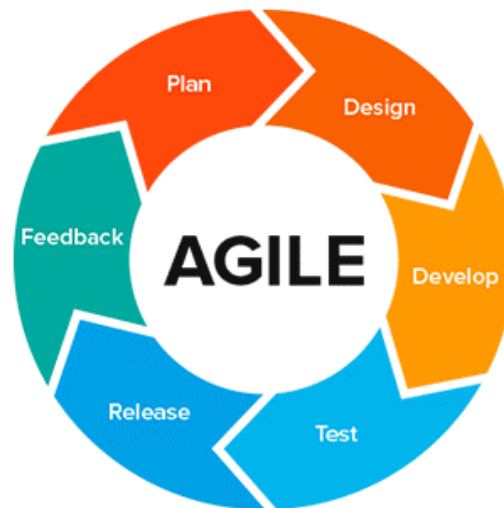
affidabili e che soddisfino tutte le aspettative del cliente. I processi cardine citati comprendono:

- \* **Pianificazione e progettazione:** il reparto si occupa della creazione di flussi di lavoro coinvolgenti la tecnologia RFID sulla base delle esigenze specifiche espresse dal cliente, servendosi di un commerciale e di un tecnico specializzato. Una volta effettuato questo incontro viene creata un'offerta commerciale che viene inviata al cliente in modo che questo possa accettarla o meno. In caso di approvazione si inizia a definire in dettaglio i requisiti tecnici per lo sviluppo del *software* e sui dispositivi *hardware* che dovrebbero essere utilizzati se già in possesso del cliente. In seguito si definiscono i dettagli del flusso di lavoro, questo è definito come l'insieme di tutti gli *step* che un *tag RFID* effettuerà e che eventi questi debbano scatenare. Una volta che il flusso è definito si passa all'implementazione, partendo da una piattaforma proprietaria dell'azienda vengono realizzate tutte le personalizzazioni necessarie in un determinato linguaggio di programmazione. Concluso ciò si passa alla fase di *test* e collaudo in cui si verifica che l'applicazione realizzata rispetti tutte le specifiche di progetto, così come collaudi funzionali e di performance. Infine il prodotto *software* che ha superato il collaudo viene installato all'interno dell'infrastruttura di rete del cliente e grazie ad interventi di tecnici specializzati la piattaforma viene resa utilizzabile dagli utenti.
- \* **Assistenza e manutenzione *software*:** il reparto fornisce un servizio di assistenza e supporto al cliente nell'utilizzo e nella manutenzione di tutte le soluzioni sviluppate internamente, o da aziende terze con cui sono stati stretti appositi accordi commerciali. Il cliente nello specifico può inoltrare una richiesta di assistenza tramite l'apertura di un così detto *ticket* o tramite chiamata al centralino aziendale apposito. Il *ticket* verrà quindi preso in carico da un apposito tecnico. Una volta svolto il lavoro necessario viene poi inviata una notifica di completamento al cliente.
- \* **Gestione e monitoraggio infrastruttura installata:** il reparto fornisce un servizio di monitoraggio sull'infrastruttura del cliente in cui il *software* sviluppato viene installato. Questo comprende diagnostiche sullo stato della rete, sulle performance dei dispositivi *hardware* utilizzati e di tutte le possibili anomalie causate dall'errore di un operatore umano. Grazie a questo è possibile fornire *feedback* tempestivi al cliente in modo da poter intervenire sulla causa del problema, andando ad arginare le perdite economiche dovute ad un arresto della produzione.

A fare da collante ai processi descritti è l'utilizzo di un *way of working* orientato al modello Agile. Questo permette al *Project Manager* di suddividere il lavoro complessivo in molti piccoli incrementi, ognuno di questi formato da diverse fasi, queste sono illustrate in figura 1.1. Questo approccio garantisce che il processo di sviluppo sia facilmente controllabile e misurabile, ciò permette al responsabile di progetto di rispondere in maniera efficace ad imprevisti e modifiche identificate a sviluppo avviato. Olte a ciò, viene data la possibilità al cliente di seguire l'avanzamento dello sviluppo del progetto, incremento dopo incremento, potendo così dare *feedback* su quanto si è già prodotto e su cosa si andrà ad implementare.

Al fine di raggiungere questi obiettivi è di vitale importanza la comunicazione, ritenuta attività fondamentale della gestione di progetto. Il gruppo di lavoro tende ad avere più contatti possibili tra gli stessi membri, sia durante lo *smartworking* che in ufficio, così da rimanere aggiornati in modo costante riguardo il lavoro svolto.

Ad inizio di ogni giornata inoltre si effettua una breve riunione riguardante il progresso dei prodotti in sviluppo e si espongono eventuali difficoltà riscontrate nella giornata precedente. La medesima cosa viene fatta col cliente: una volta a settimana viene fatta una *call* collettiva tra tutti i partecipanti al progetto ed i referenti dell'azienda committente, in cui vengono illustrati tutti i progressi effettuati.



**Figura 1.1:** Illustrazione del modello Agile - fonte [urly.it/3g72a](http://urly.it/3g72a)

## 1.3 Tecnologie utilizzate

Come è facile intuire dai numerosi ambiti in cui l'azienda offre soluzioni e servizi, anche il numero di tecnologie utilizzate risulta essere consistente ed in continua evoluzione. Di seguito verranno riportate le tecnologie principali utilizzate all'interno del reparto [Radio Frequency Identification](#) con cui ho potuto venire in contatto durante il periodo di stage.

### 1.3.1 Sistemi operativi

Per quanto concerne i sistemi operativi, vi è una distinzione nel loro utilizzo in base al ruolo che il dipendente ricopre all'interno del reparto. Per gli sviluppatori viene adoperato Windows, in quanto, grazie alla sua robustezza e alla facile amministrazione remota permette la creazione di ambienti di lavoro omogenei per l'intero team. Per quanto riguarda invece la parte più amministrativa, la soluzione adottata è stata quella di dispositivi Apple e quindi aventi MacOS. Per ultimo ma non per importanza troviamo Linux, usato come piattaforma per ospitare le soluzioni sviluppate da Aton presso le sedi dei loro clienti.

### 1.3.2 Ambienti di sviluppo

Per la scrittura del codice non viene espressa una preferenza specifica, agli sviluppatori viene lasciata la possibilità di utilizzare una delle seguenti *IDE*: Eclipse, IntelliJ Idea o Visual Studio. La scelta di una di queste ricade in primo luogo sulle esigenze specifiche del progetto su cui lo sviluppatore si trova a lavorare, ed in secondo luogo sulla propria preferenza personale. Ognuno di questi programmi porta con sé un numero molto elevato di funzionalità, queste se combinate con ambienti di lavoro pre-configurati allestiti negli anni e disponibili agli sviluppatori, vanno a formare delle soluzioni pronte che rendono lo spostamento da un progetto ad un altro semplice ed immediato. Oltre ai prodotti già citati, non è possibile non citare la rapida diffusione negli ultimi anni di un altro strumento dalla grande popolarità, l'*editor* Visual Studio Code. Questo programma multiplatforma nasce con lo scopo di essere un *editor* "tutto fare", di base infatti è provvisto solo di strumenti avanzati per la scrittura e la formattazione del testo. Grazie però ad un vasto e ricco market di estensioni, è possibile una sua profonda e completa personalizzazione integrando quindi il supporto a tutti i linguaggi di programmazione più rilevanti presenti sul mercato. Questo porta alla creazione di un ambiente dinamico che possa rispondere a tutte le esigenze di ogni sviluppatore.

### 1.3.3 Sistemi di versionamento

Dato il numeroso team di lavoro risulta indispensabile fare affidamento su un sistema di versionamento, questo tiene traccia del flusso di lavoro e di tutte le modifiche che vengono apportate al codice sorgente. Storicamente, all'interno dell'azienda, il punto di riferimento per questo compito è stato il *software open source* Subversion. Questo però sta lentamente venendo abbandonato per effettuare una massiccia migrazione verso Git, altro *software* per il controllo di versione del codice distribuito che è diventato di fatto uno standard per il mercato odierno.

### 1.3.4 Suite per la produttività e comunicazione

Per quanto riguarda l'organizzazione e la comunicazione viene fatto un ampio uso dei servizi proposti da Microsoft. Di questi, vengono usati principalmente:

- \* **Outlook**: servizio di posta elettronica che comprende anche un *client* desktop ed un calendario condivisibile. Viene utilizzato sia internamente tra i dipendenti che coi clienti.
- \* **Onedrive**: servizio *web* per la memorizzazione, sincronizzazione e condivisione di file online.
- \* **Office Suite**: pacchetto di applicazioni *software* che presenta numerose funzionalità, come la creazione e modifica di *file* di testo, fogli di calcolo, presentazioni e reportistica varia.
- \* **Teams**: piattaforma di comunicazione e collaborazione unificata che combina chat di lavoro persistente, teleconferenza, condivisione di contenuti e integrazione delle applicazioni esterne.

### 1.3.5 Linguaggi di sviluppo

Avendo l'esigenza di sviluppare progetti e soluzioni *software* che si interfaccino frequentemente con dispositivi *hardware* e che possano operare andando ad interfacciarsi con



un importante numero di applicativi scritti da terzi, viene fatto uso di diversi linguaggi di programmazione:

- \* **Java:** basato su Java Virtual Machine(JVM), ormai da svariati anni è il più utilizzato per il suo vastissimo numero di librerie. Questo, combinato alla possibilità di eseguire il codice in modo parallelo e concorrente lo rende uno strumento solido ed affidabile. Un altro vantaggio di tale linguaggio riguarda l'esecuzione del codice stesso, questo infatti risulta indipendente dall'*hardware* della macchina in quanto virtualizzato dalla piattaforma stessa. In tal modo il codice è reso portabile verso altre piattaforme con *hardware* diversi tra loro.
- \* **C#:** linguaggio di programmazione multiparadigma, viene sviluppato da Microsoft e presenta numerose somiglianze con Java. Negli ultimi anni ha avuto una forte crescita dovuta ad alcune sue caratteristiche uniche come la possibilità di interfacciare nativamente codice scritto in linguaggi compilati come C e C++. Questo, unito ad un solido ecosistema di librerie e dalle sue notevoli performance, ha spinto numerosi produttori ad adottarlo come strumento per interfacciarsi con i propri dispositivi *hardware*.
- \* **Javascript:** linguaggio di programmazione orientato agli oggetti tramite prototipi e agli eventi. Grazie alla sua grande popolarità ed alla produttività nella scrittura del codice viene utilizzato per prototipizzazione o per specifici ambiti.

### 1.3.6 Framework

Il principale e sicuramente più utilizzato è OSGi Knopflerfish, un *framework open-source* che si propone di implementare un modello a componenti completo e dinamico per la piattaforma Java. I concetti fondamentali dietro questa tecnologia sono essenzialmente 3:

- \* Definizione del concetto di modulo (*bundle*)
- \* Gestione automatica delle dipendenze
- \* Gestione del ciclo di vita del codice (configurazione e distribuzione dinamica)

Grazie a ciò viene reso possibile creare un sistema dinamico, che consente l'installazione, l'avvio, lo stop e la rimozione dei moduli a *runtime*, senza quindi necessitare di riavvii. Un esempio di come il codice venga organizzato tramite il concetto di modulo è visibile nella figura 1.2. Non a caso quindi è stato adottato per la creazione della principale piattaforma utilizzata dal reparto RFID, AMP, che offrendo la possibilità di creare nuovi moduli si rende una perfetta base di partenza per progetti orientati alla personalizzazione per il singolo cliente.

## 1.4 Propensione all'innovazione

L'azienda da molti anni ha deciso di mettere la formazione continua come uno dei suoi pilastri principali, operando in mercati dinamici e con tecnologie in continua evoluzione risulta quindi indispensabile continuare ad ampliare le proprie conoscenze, in modo da non essere superati dai vari *competitor*. A tal fine vengono messi a disposizione di ogni dipendente numerosi corsi di formazione, questi spaziano da certificazioni per tecnologie specifiche, corsi sulla gestione di progetto fino ad arrivare a corsi di lingue

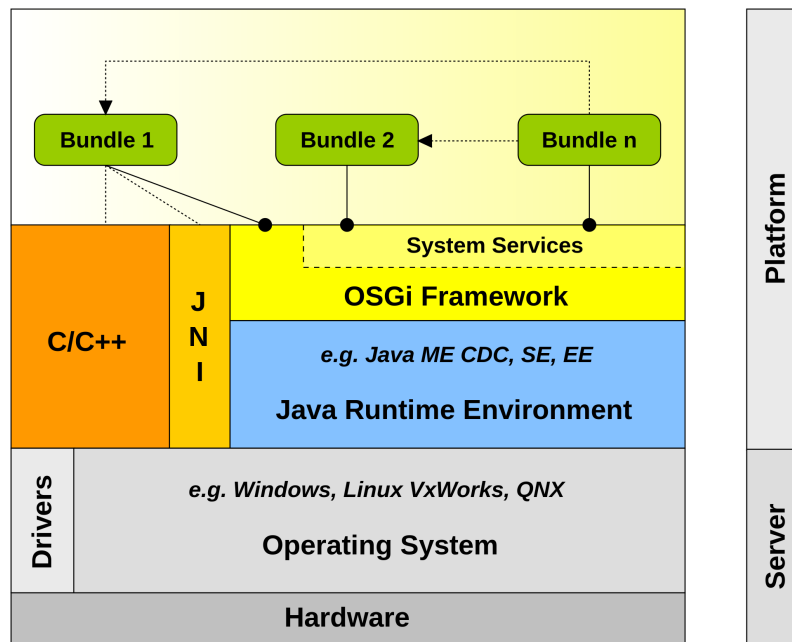


Figura 1.2: Architettura con OSGi - fonte [urly.it/3g72y](http://urly.it/3g72y)

straniere.

Il reparto RFID non fa eccezione, viene richiesto infatti a tutti gli sviluppatori di seguire percorsi formativi su un'ampia scelta di tecnologie, identificate come cruciali per il continuo rinnovamento ed espansione dell'azienda. Questi comprendono ad esempio corsi su tecnologie riguardanti la persistenza dei dati fino all'utilizzo di *framework* di nuova generazione. Oltre a questo Aton ha dimostrato negli anni una forte connessione con l'ambiente universitario e quindi di conseguenza con la ricerca che si svolge in esso. Uno degli esempi più significativi è sicuramente la collaborazione avvenuta con l'Università di Torino con cui si sono svolte delle ricerche riguardanti proprio la tecnologia RFID. Queste hanno portato alla creazione delle basi della piattaforma proprietaria per tale tecnologia, AMP, utilizzata ancora oggi come base di partenza per tutti i progetti cliente sviluppati.

Correlata alla connessione con gli ambienti universitari appena citati, troviamo l'occupazione di un dipendente Aton, che oltre al suo lavoro in azienda svolge l'attività di professore nel corso "Altri paradigmi di programmazione" all'interno della laurea in Informatica presso l'Università di Padova.

## Capitolo 2

# Il progetto Kathrein

*Nel corso di questo capitolo si presenta il progetto di stage esponendo come queste attività siano interpretate dall'azienda. Ne vengono poi descritti gli obiettivi, i vincoli ed il futuro al termine della mia partecipazione.*

### 2.1 Descrizione

Il progetto di stage propostomi da Aton S.P.A., mediante la figura del mio *tutor* interno, è riconducibile alla visione d'insieme che l'azienda possiede nei confronti delle collaborazioni con le università. Queste opportunità non sono nuove all'azienda, nel corso degli anni sono state svolte numerose collaborazioni con diversi atenei che hanno portato alla realizzazione di prodotti ancora oggi utilizzati.

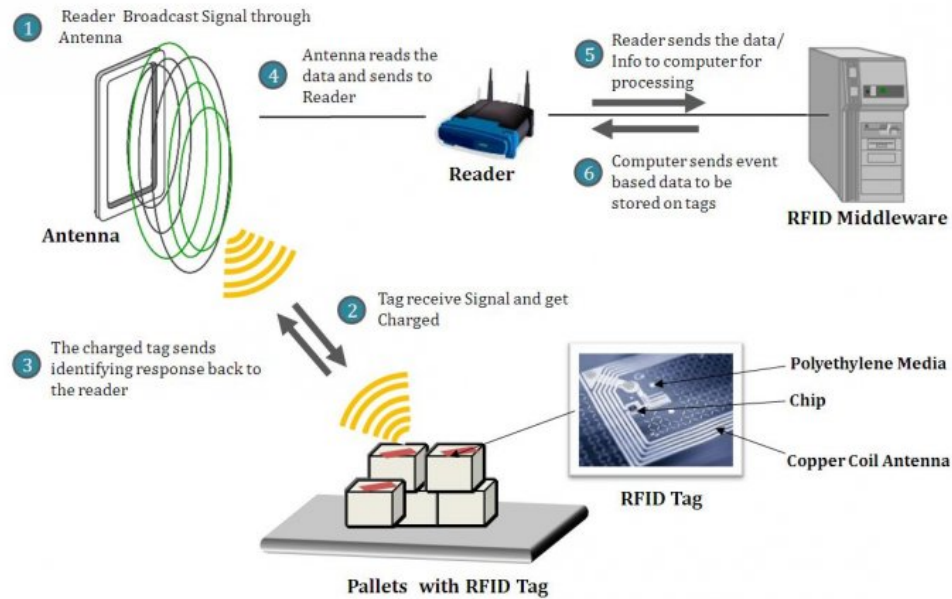
Tali percorsi vengono interpretati come mezzo per ampliare il bagaglio di conoscenze proprio dell'azienda riguardo a nuove tecnologie. Questo scopo viene perseguito tramite l'implementazione di *software* che possa contribuire alle soluzioni utilizzate nei vari progetti cliente. Oltre a questo, gli stage proposti si pongono come secondo fine la pubblicizzazione dell'azienda cercando di attirare nuovi sviluppatori che possano far crescere il team già esistente.

Nel mio caso specifico lo scopo del progetto era quello di studiare l'interazione con dei nuovi lettori RFID per poi applicare quanto imparato tramite lo sviluppo di un prodotto *software* apposito. Tale prodotto era quindi finalizzato a diventare un nuovo componente utilizzabile dai tecnici aziendali per la creazione di flussi di lavoro adoperanti i lettori analizzati, accrescendo di conseguenza le soluzioni offerte dall'azienda nei vari progetti cliente. Un esempio concreto della posizione di questo prodotto, all'interno di un sistema RFID classico, è rappresentato nei passi 5 e 6, riportati nella figura [2.1](#).

### 2.2 Obiettivi

#### 2.2.1 Obiettivi aziendali

L'azienda ha posto dei macro obiettivi principali per quanto riguarda le funzionalità dell'applicativo, sia a livello utente che a livello software. Questi sono stati esplicitati nei primi giorni in cui mi è stato presentato il progetto in dettaglio dal mio *tutor* interno.



**Figura 2.1:** Schema di un sistema RFID - fonte [urly.it/3gc8k](http://urly.it/3gc8k)

- \* L'applicativo avrebbe dovuto interfacciarsi con tutte le generazioni di lettori Kathrein;
- \* L'applicativo avrebbe dovuto funzionare in maniera autonoma, senza quindi la presenza di un operatore atto ad operare i *device*;
- \* L'applicativo avrebbe dovuto offrire le stesse funzionalità all'utente degli altri applicativi, dedicati ad altri marchi di lettori;
- \* L'applicativo avrebbe dovuto supportare la lettura di numerosi *tag* simultaneamente, nell'ordine di qualche centinaio;
- \* L'applicativo avrebbe dovuto essere fruibile direttamente all'interno della piattaforma AMP;
- \* Le anomalie dovute alla rete dovevano essere gestite e notificate se non risolvibili;
- \* L'applicativo avrebbe dovuto comandare altri dispositivi IoT come semafori o *buzzer* il cui operato dipendesse da quello del lettore.

### 2.2.2 Obiettivi personali

Tramite l'Università ho avuto accesso all'evento **Stage-IT**, offerto da AssIndustria Veneto-Centro, in cui vengono messi in diretto collegamento aziende in collaborazione con il Dipartimento di Padova e gli studenti laureandi, in modo da poter proporre stage pre-laurea. Durante l'evento numerosi progetti hanno attratto la mia attenzione, l'offerta proposta da Aton S.P.A. però è riuscita a catturare a pieno il mio interesse. Causa di questo è stata la possibilità di lavorare in un contesto a me nuovo ma in fortissima crescita nel mercato odierno. In particolare, l'utilizzo di codice nativo tramite Java, all'interno di un contesto in cui l'ottimizzazione ricopre un ruolo cruciale, mi ha

fin da subito affascinante. Questo mi ha permesso di applicare quanto imparato durante gli anni studio, garantendomi allo stesso tempo una grande occasione per apprendere nuove tecnologie. Sapevo, inoltre, che sarei stato inserito in un ambiente di lavoro giovane, dinamico e professionale dove la collaborazione viene ritenuta uno dei pilastri fondamentali per la buona riuscita di ogni progetto.

Gli obiettivi che ambivo a raggiungere a livello personale riguardavano principalmente la mia formazione riguardo tecnologie a me estranee. In particolare, ambivo a poter apprendere il funzionamento di un sistema RFID, andando poi ad applicare quanto imparato nello sviluppo di prodotto capace di essere resiliente ai guasti e che si interfacciasse con un *device* fisico. In aggiunta a questo, era mia intenzione osservare ed apprendere il più possibile le metodologie di lavoro utilizzate e le *best practices*, consolidate e utilizzate da anni dal gruppo di *tech lead* presente all'interno dell'azienda. Questo per poter conoscere le strategie che, partendo da un'idea, portano alla concretizzazione di un prodotto finito e funzionante.

## 2.3 Vincoli

### 2.3.1 Vincoli tecnologici

Essendo la soluzione basata sulla piattaforma AMP e comprendente delle librerie native proprie del lettore, i vincoli tecnologici hanno fatto sì che venissero utilizzate specifiche tecnologie:

- \* **Java 8**: linguaggio di programmazione utilizzato per lo sviluppo;
- \* **OSGi Knopflerfish**: *framework* utilizzato per la realizzazione di AMP, mette a disposizione la possibilità di creare moduli autonomi;
- \* **JNA**: libreria per accedere a codice nativo.

### 2.3.2 Vincoli temporali

Il tempo determinato dallo stage, ovvero 320 ore, è stato il vincolo temporale più rilevante, in quanto è stato impiegato in larga parte per la formazione personale rispetto alle tecnologie da utilizzare e alla teoria riguardante la tecnologia RFID, indispensabile per capire il funzionamento dei lettori. Un altro aspetto indispensabile da considerare è stato riguardo la documentazione offerta dal produttore del lettore, questa a fine commerciali è stata considerevolmente scarsa ed ostica. Questo, unito alla presenza di alcuni bug o incompletezze che ho riscontrato all'interno del loro codice ha portato ad una frequente comunicazione fra me ed il loro reparto di sviluppo. Questo necessario scambio di informazioni ha inevitabilmente ridotto il tempo a mia disposizione per soddisfare le richieste fattemi.

**POSSIBILE MODIFICA:** Il tempo determinato dallo stage, ovvero 320 ore, è stato il vincolo temporale più rilevante, in quanto veniva incluso in esso la mia formazione autonoma riguardo: la teoria sulla tecnologia RFID, le tecnologie utilizzate e le librerie offerte dal produttore dei lettori. In aggiunta a questo, le revisioni di progetto settimanali, di cui si parla in maniera approfondita nella sezione [3.1.2](#), richiedono un puntuale sviluppo di ogni incremento in modo da facilitare l'organizzazione di tali incontri.

## 2.4 Futuro del progetto

In connessione a quanto riportato nella sezione [2.1](#), è facile vedere come il prodotto sviluppato fosse interpretato come uno strumento utilizzabile in futuri progetti cliente. La visione complessiva dell'azienda, rispetto al prodotto, risultava però essere più ampia. L'applicativo non avrebbe dovuto essere fine a se stesso, ma rappresentare una base di studio per qualsiasi altro programmatore intento nello sviluppo di una soluzione basata sui lettori Kathrein o, più in generale, necessitante di interfacciarsi con del codice nativo.

## Capitolo 3

# Resoconto del progetto

*Nel corso di questo capitolo si presenta un resoconto del progetto, descrivendone gli aspetti nel dettaglio. Si illustrano le metodologie di lavoro utilizzato e le scelte dietro lo sviluppo delle componenti ritenute più rilevanti o critiche*

### 3.1 Pianificazione delle attività

#### 3.1.1 Comunicazioni

#### 3.1.2 Revisioni di progetto

### 3.2 Analisi del progetto

#### 3.2.1 Funzionalità

#### 3.2.2 Requisiti chiave

#### 3.2.3 Casi d'uso

#### 3.2.4 Tecnologie coinvolte

### 3.3 Sviluppo delle componenti critiche

#### 3.3.1 Design del modulo

#### 3.3.2 Interfacciamento con librerie native

#### 3.3.3 Ottimizzazione dei tempi di lettura

#### 3.3.4 Linguaggio di dominio specifico

#### 3.3.5 Controllo delle porte GPIO

#### 3.3.6 Tolleranza ai guasti

### 3.4 Risultati raggiunti

#### 3.4.1 Requisiti soddisfatti

#### 3.4.2 Prodotti creati



## Capitolo 4

# Valutazione retrospettiva e conclusioni

*Breve introduzione al capitolo*

4.1 Obbiettivi raggiunti

4.2 Progressione personale

4.3 Gap formativo tra università e stage



# Glossario

**IoT** Internet delle cose, nelle telecomunicazioni è un neologismo riferito all'estensione di Internet al mondo degli oggetti e dei luoghi concreti.. [1](#), [17](#)

**M2M** Con *Machine-to-machine*, in generale ci si riferisce a tecnologie e applicazioni di telemetria e telematica che utilizzano le reti wireless. Machine-to-machine indica anche un insieme di software e applicazioni che migliorano l'efficienza e la qualità dei processi tipici di ERP, CRM e asset management.. [1](#), [17](#)

**RFID** Identificazione a radiofrequenza, in elettronica e telecomunicazioni, è una tecnologia di riconoscimento e validazione e/o memorizzazione automatica di informazioni a distanza.. [3](#), [17](#)



# Acronimi

**IoT** [Internet Of Things](#). 15

**M2M** [Machine-to-machine](#). 15

**RFID** [RFID](#). 15



# 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/>.