

Università degli Studi di Padova

DIPARTIMENTO DI MATEMATICA "TULLIO LEVI-CIVITA"

CORSO DI LAUREA IN INFORMATICA



Beryllium, una piattaforma per il software asset management

Tesi di laurea triennale

Relatore

Prof. Mauro Conti

Tutor esterno

Jary Busato

Laureando

Lorenzo Ferrarin

ANNO ACCADEMICO 2016-2017

Dedicato ai miei genitori e ai miei amici, senza i quali non sarei mai arrivato fino a qui

Sommario

Nel presente documento viene descritto il lavoro svolto presso l'azienda NETCOM srl di Padova dal laureando Lorenzo Ferrarin durante il periodo di stage.

Lo stage è consistito nello sviluppo di un applicazione per la semplificazione del Software Asset Management. In particolare la parte affrontata è stata quella riguardante il licensing dei prodotti Microsoft. Allo studente è stato infatti richiesto di sviluppare una serie di programmi il cui scopo era quello di determinare il numero di licenze di uno specifico prodotto Microsoft che era necessario acquistare. Tali informazioni dovevano essere ricavate in base ad una serie di dati estratti dal computer sul quale è eseguito il programma. Compito dello studente è stato studiare i diversi prodotti da analizzare, individuare le informazioni che era necessario estrarre e trovare un modo per ottenerle.

Allo studente è stato inoltre richiesto di sviluppare un prototipo dell'interfaccia web finale del progetto.

Ringraziamenti

Innanzitutto, vorrei esprimere la mia gratitudine al Prof. Mauro Conti, relatore della mia tesi, e a Daniele Lain, per l'aiuto e il sostegno fornitomi durante la stesura del lavoro.

Desidero inoltre ringraziare con affetto i miei genitori e i miei amici per il continuo sostegno, il grande aiuto e per essermi stati vicini durante gli anni di studio.

Padova, Settembre 2017

Lorenzo Ferrarin

Indice

1	Introduzione	1
1.1	L'idea	1
1.2	L'azienda	1
1.2.1	Organizzazione aziendale	3
1.2.2	Tecnologie utilizzate dall'azienda	4
1.3	Struttura del documento	6
1.4	Convenzioni tipografiche	6
2	Il progetto aziendale	9
2.1	Proposta di stage	9
2.2	Beryllium	9
2.3	Motivazioni	11
2.4	Obiettivi	11
2.5	Vincoli	12
3	Lo stage	13
3.1	Ambiente e strumenti di sviluppo	13
3.2	Piano di Lavoro	15
3.2.1	Obiettivi pianificati	15
3.2.2	Pianificazione attività e scadenze	15
3.3	Analisi	17
3.3.1	Casi d'uso	17
3.3.2	Requisiti	19
3.4	Progettazione	21
3.5	Documentazione	22
3.6	Test	22
4	Musician	25
4.1	Microsoft Exchange	25
4.2	Microsoft Remote Desktop Services	28
4.3	Microsoft SharePoint	29
4.4	Dynamic Host Configuration Protocol	31
4.5	Microsoft Domain Name System	32
4.6	Microsoft Office	34
4.7	Microsoft Project	35
4.8	Microsoft Visio	37
5	Opera House	39

6	Valutazione retrospettiva	43
6.1	Raggiungimento degli obiettivi	43
6.2	Conoscenze acquisite	44
6.3	Conclusioni	46
	Glossario	47
	Acronimi e abbreviazioni	51
	Bibliografia	53

Elenco delle figure

1.1	Framework ITIL.	2
1.2	Struttura aziendale NETCOM.	4
1.3	Landesk.	5
1.4	Funzionamento di EasyVista.	5
1.5	Schema di Greenlight.	6
3.1	Diagramma di Gantt: attività e scadenze definite dal Piano di Lavoro.	16
3.2	Classe Musician generica.	21
5.1	Opera House - Homepage.	40
5.2	Opera House - Overview Page.	41
5.3	Opera House - Inventory Page.	41
5.4	Opera House - Report Page.	41
5.5	Opera House - Edit Page.	42
5.6	Opera House - Settings Page.	42

Elenco delle tabelle

3.1	Requisiti di vincolo.	20
3.2	Requisiti funzionali.	21
3.3	Riepilogo dei requisiti.	21

Capitolo 1

Introduzione

In questo capitolo viene introdotta l'idea generale dello stage e successivamente viene descritta in dettaglio l'azienda dove è stato svolto.

1.1 L'idea

Il presente documento descrive il lavoro svolto durante il periodo di stage, della durata di 320 ore, dal laureando Lorenzo Ferrarin presso l'azienda NETCOM srl.

Il laureando è stato inserito in un team di sviluppo che lavora ad un progetto di nome Beryllium. Tale progetto si propone di sviluppare una piattaforma unica di gestione per il Software Asset Management (SAM)^[gl] aziendale.

Questa piattaforma permette di assemblare, rielaborare e visualizzare in maniera chiara e concisa tutte le informazioni raccolte da specifici *software* di monitoraggio delle risorse. Lo scopo di tali *software* è quello di raccogliere dati riguardanti programmi utilizzati da un utente e ricavarne la lista di licenze necessarie ad utilizzarli. I dati raccolti da questi *software* possono dunque poi venire utilizzati dall'azienda per capire la quantità di licenze che deve possedere.

Durante lo stage il laureando si è occupato dello sviluppo di alcuni dei *software* di monitoraggio delle risorse, chiamati Musician, per lo sviluppo dei quali è stata necessaria la sua formazione nelle tecnologie che i *software* andranno a monitorare e in particolare nelle loro regole di licensing^[gl].

Lo studente ha inoltre collaborato allo sviluppo di un prototipo dell'interfaccia web dell'intero applicativo, chiamata Opera House, che permette agli utenti di accedere alle informazioni raccolte dai Musician in maniera facile e comprensibile.

1.2 L'azienda

NETCOM è un'azienda nata nel 1999 con l'obiettivo di proporre al mercato di imprese di medie e grandi dimensioni soluzioni e servizi di eccellenza di consulenza informatica, di IT Operations Management^[gl] e di IT Life Cycle Management^[gl]. Quest'ultimo viene inteso dall'azienda come l'insieme di processi, procedure, competenze e strumenti utilizzati nella gestione dei *Configuration Item*^[gl] che compongono il sistema informativo in tutte le fasi del loro ciclo di vita. In particolare per gli aspetti di:

- **Gestione Tecnica:** per la soluzione di tutti quei problemi che i clienti possono riscontrare nel corso dell'utilizzo del sistema;

- **Governance:** per la gestione di tutte le operazioni IT di livello manageriale ed allinearle alle richieste del *business*.

L'IT Life Cycle Management si occupa della gestione, ottimizzazione e continuo miglioramento del sistema informativo di un'organizzazione, permettendo di ridurre costi, rischi e complessivamente aumentandone la qualità. Per fornire il miglior servizio possibile il personale del settore tecnico di NETCOM vanta molte certificazioni di *best practice* (come Information Technology Infrastructure Library (ITIL)^[8], una serie di linee guida per la gestione dei servizi IT schematizzate in Figura 1.1) e standard (*ISO 9001* e *ISO 20000*) per la gestione dei sistemi informativi. Essendo inoltre l'azienda partner di LANDESK, SNOW, EasyVsta e Microsoft il personale dispone di certificazioni che attestano la loro competenza nei rispettivi *software*.

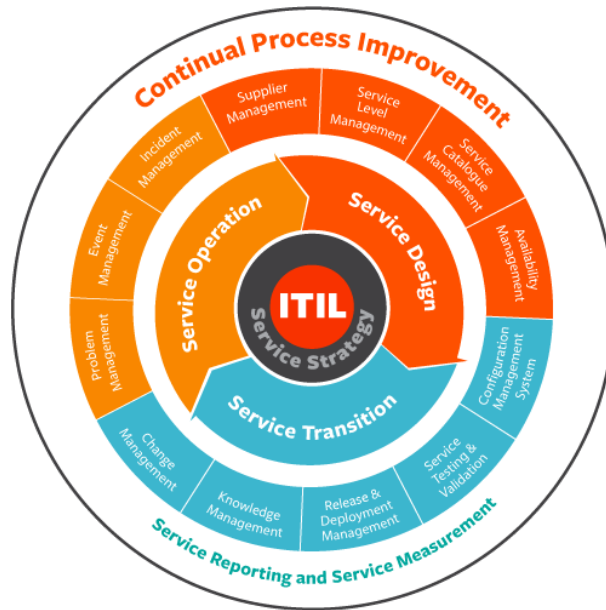


Figura 1.1: Framework ITIL.

La tipologia dei principali servizi offerti da NETCOM colloca l'azienda nel mercato delle soluzioni di *System Management*, dedicato alla gestione di tutti gli elaborati presenti all'interno di un'organizzazione. L'azienda tuttavia si focalizza prevalentemente su aspetti di più alto livello di quello *hardware*, come: la distribuzione dell'intero strato *software* necessario all'utilizzo di un terminale, l'installazione distribuita di applicazioni, la gestione remota di aggiornamenti ed eventuali vulnerabilità, l'inventario dello stato e della configurazione delle diverse macchine all'interno di un'organizzazione e il controllo remoto degli *host*.

NETCOM conta più di 150 clienti di varie tipologie e dimensioni sparsi su tutto il territorio nazionale, in particolare nel nord e centro Italia, comprendenti aziende operanti nel settore sia pubblico che privato. Per la natura dei servizi offerti NETCOM si rivolge alla medio-grande impresa italiana che, se misurata attraverso il numero dei dispositivi posseduti, possiamo collocare in un range tra le 250 e le 10000 unità. Data l'ampia gamma di servizi NETCOM è un'azienda che si adatta a molte tipologie di clien-

tela, a partire dalle aziende che ricercano un mantenimento e miglioramento della loro situazione attuale fino ad arrivare alle aziende che desiderano investire sull'innovazione, in particolare con soluzioni di *System Management* e di sicurezza. Vediamo ora i principali servizi offerti da NETCOM analizzando per ciascuno il tipo di cliente a cui si rivolge:

- **Assistenza tecnica:** rappresenta l'offerta base ed è un servizio che viene fornito quando ne viene fatta richiesta. Quando un'organizzazione si trova di fronte ad un problema potrebbe non avere le conoscenze o le risorse necessarie per risolverlo. Per evitare i rischi derivati da una tale situazione certe aziende decidono di pagare per potersi appoggiare a personale esterno più esperto e qualificato. Questo servizio non richiede grossi investimenti, quindi si adatta ad aziende con un budget limitato ma che hanno bisogno di risolvere in maniera efficace ed efficiente eventuali problemi tecnici.
- **Gestione delle vulnerabilità:** le aziende al giorno d'oggi non possono permettersi di perdere produttività o dati a causa di incidenti di sicurezza. La gestione delle vulnerabilità informatiche è quindi un processo critico per il *business* ma spesso le aziende non dispongono del personale necessario a gestirlo con la dovuta attenzione. NETCOM dunque offre un servizio di gestione delle vulnerabilità, il cui obiettivo è quello di supportare il dipartimento IT di un'organizzazione a gestire tutte le operazioni relative alla sicurezza delle stazioni di lavoro, dei notebook e dei server. Il servizio è composto da un processo di *vulnerability assessment* che consente alle organizzazioni di valutare in modo affidabile lo stato delle vulnerabilità del sistema, i relativi rischi, e allo stesso tempo tenere sotto controllo i costi delle azioni di rimedio e di mantenimento del livello di sicurezza richiesto. Questo servizio può adattarsi a molte tipologie di aziende, sia da quelle con un dipartimento IT non sufficientemente sviluppato da gestire le problematiche, sia a quelle che preferiscono delegare questo compito ad esterni.
- **Software Asset Management:** spesso le organizzazioni non sono a conoscenza di quali e quante delle licenze *software* acquistate sono effettivamente utilizzate. Il SAM è un approccio strutturato e sistematico per la gestione dell'intero ciclo di vita del *software* che consiste in una serie di processi, *best practice*, strumenti e competenze per la gestione del *software* aziendale che portano a numerosi vantaggi, come la riduzione dei costi, la semplificazione della pianificazione del budget, l'evitare sanzioni e danni d'immagine causati dalla non conformità delle licenze e da litigi con i vendor. NETCOM offre servizi multilivello che aiutano le aziende a proteggere, gestire e valorizzare le risorse *software*, supportando il dipartimento IT nelle verifiche di conformità e nell'ottimizzazione del *software* utilizzato a supporto del *business* aziendale.

1.2.1 Organizzazione aziendale

NETCOM è un'azienda di circa 30 dipendenti con sede a Padova. Si riporta, in Figura 1.2, l'organigramma della sua struttura interna. Durante lo stage lo studente è stato inserito all'interno del dipartimento di Development, che si occupa dello sviluppo di soluzioni proprie interne all'azienda, dove ha collaborato allo sviluppo del progetto.

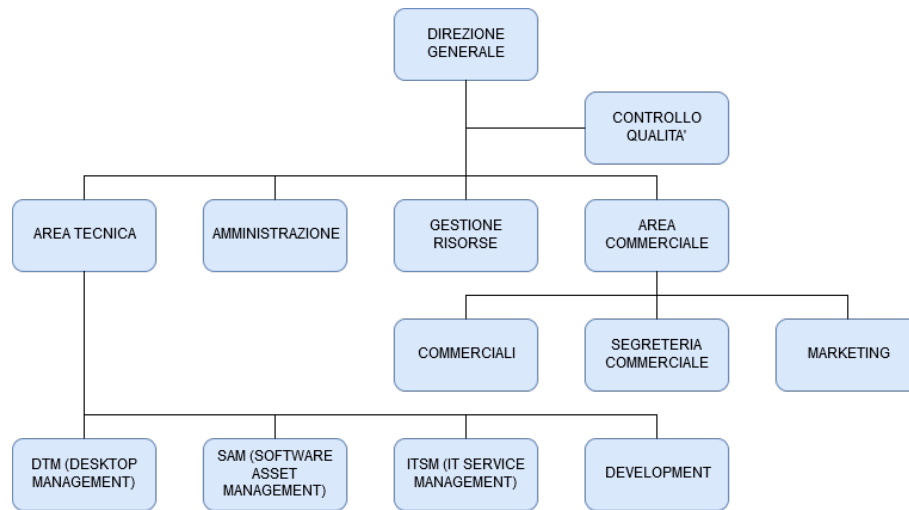


Figura 1.2: Struttura aziendale NETCOM.

1.2.2 Tecnologie utilizzate dall'azienda

L'azienda utilizza diverse tecnologie per fornire i suoi servizi. Qui di seguito verranno presentate le principali.

- **Landesk:** Il servizio Landesk, presentato in Figura 1.3, offre un approccio multilivello alla sicurezza per la salvaguardia della produttività e la protezione in maniera integrata degli endpoint. In particolare aiuta gli amministratori IT ad automatizzare il *deployment* di *software* e di sistemi operativi oltre che a rendere stabile e sicuro l'ambiente informatico della propria azienda con un sistema di *patching* automatizzato. Landesk permette anche un tracciamento costante dei dispositivi e delle risorse *software* di un'azienda, fornendo un servizio di protezione dei dati e dei dispositivi costantemente aggiornato, evidenziando rischi, facilitando l'adattamento alle attuali norme vigenti e integrandosi con la soluzione antivirus selezionata dall'azienda.
- **EasyVista:** EasyVista offre soluzioni complete integrate e modulari che consentono alle organizzazioni di aumentare la propria efficienza per supportare l'adozione dell'ITIL e per rispondere alle sempre più complesse esigenze di *business*. L'approccio EasyVista mette a disposizione dei clienti tecnologie in grado di supportare le strategie di ITIL e IT Service Management (ITSM)^[g] delle aziende, e di soddisfare le esigenze per l'adozione delle *best practice* ITIL, migliorando la qualità complessiva dei servizi IT. Si riporta, in Figura 1.4, uno schema del funzionamento di EasyVista, e della sua adozione all'interno di NETCOM.

Tra le funzionalità offerte ci sono:

- Apertura di ticket^[g] riguardanti incidenti e richieste di servizio;
- Gestione da parte del reparto tecnico di tutti i ticket e della loro scalabilità, permettendone l'assegnazione a persone competenti per una loro soluzione rapida e definitiva;
- Amministrazione degli asset^[g] aziendali in un'ottica di gestione integrata ad alto livello degli asset IT, del *software* e dei processi di *business* ad essi

correlati.

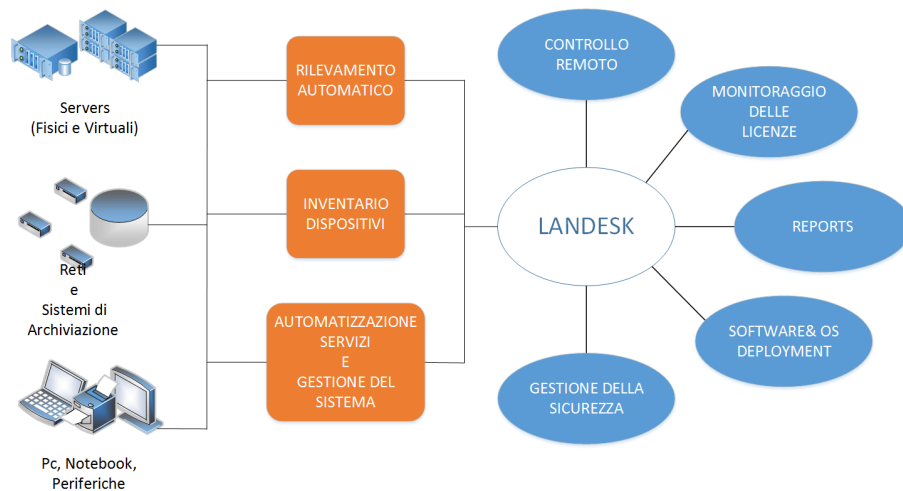


Figura 1.3: Landesk.

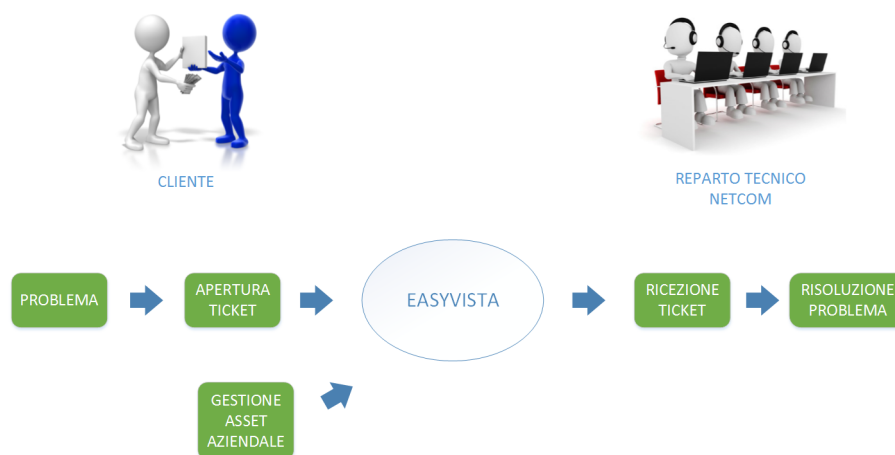


Figura 1.4: Funzionamento di EasyVista.

- **Snow License Manager:** Snow License Manager è una soluzione progettata per ridurre i rischi, i costi e la complessità associati alla gestione del *software*, evitando spese eccessive per l'acquisto di licenze e garantendo al contempo la piena conformità ai diritti d'uso. La soluzione sviluppata da Snow Software consente di portare l'automazione del SAM ad un livello molto elevato, grazie alla capacità di identificare le tipologie di *software* utilizzati, fornendo una vista unificata di tutte le risorse *software* e *hardware*, i diritti di licenza e le metriche di utilizzo delle applicazioni.
- **GREENLIGHT:** GREENLIGHT è una soluzione *self-service* sviluppata da NETCOM per il reset e la gestione delle *password* che permette di ridurre drasticamente il numero dei ticket aperti all'Help Desk^[6], diminuisce i costi e migliora la

soddisfazione degli utenti che possono evitare i tempi di inattività dovuti a inutili attese. GREENLIGHT rende disponibile un processo di reimpostazione della *password* semplice ed affidabile, attivabile dalla schermata di login di Windows o dal Web che consente agli utenti di sbloccare autonomamente il proprio account ogni volta si presenta la necessità, tramite qualsiasi dispositivo e ovunque si trovino. Questa soluzione offre inoltre la possibilità di individuare con precisione l'identità dell'utente che richiede il ripristino della *password*. Un esempio del funzionamento di GREENLIGHT è riportato in Figura 1.5. Il processo di reset della *password* si articola nei seguenti quattro passi:

- **Account check:** verifica della presenza dell'account in Active Directory^[8];
- **Enrollment check:** verifica dell'iscrizione;
- **Identity check:** verifica dell'identità;
- **Password reset:** reset della *password*.

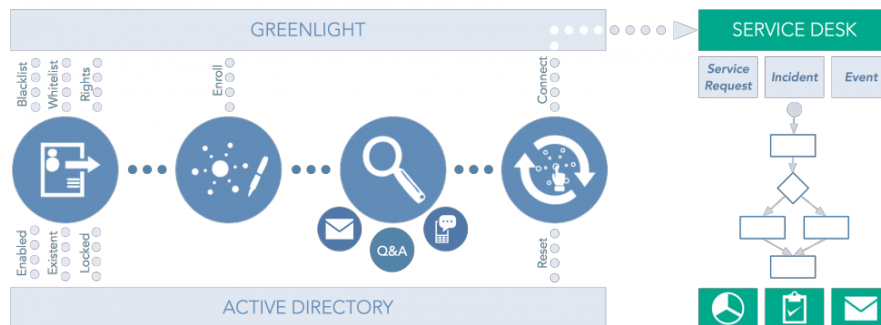


Figura 1.5: Schema di Greenlight.

1.3 Struttura del documento

Il resto del documento è suddiviso nei seguenti sei capitoli:

Il secondo capitolo approfondisce il progetto aziendale, descrivendone in dettaglio gli obiettivi, le motivazioni della sua scelta ed eventuali vincoli.

Il terzo capitolo descrive il periodo e il lavoro svolto durante lo stage.

Il quarto capitolo fornisce un approfondimento su ciascuno dei diversi *software* realizzati.

Il quinto capitolo descrive in maggiore dettaglio il prototipo dell'interfaccia web realizzato durante lo stage.

Nel sesto capitolo viene analizzato retrospettivamente il lavoro svolto per valutare il raggiungimento degli obiettivi e le conoscenze acquisite.

1.4 Convenzioni tipografiche

Durante la stesura del testo sono state adottate le seguenti convenzioni tipografiche:

- per tutti gli acronimi, abbreviazioni e termini ambigui o di uso non comune menzionati viene fornita una definizione nel glossario, situato alla fine del presente documento;
- per tutte le prime occorrenze 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*.

Capitolo 2

Il progetto aziendale

Questo capitolo presenta gli aspetti principali dello stage, gli obiettivi prefissati e le motivazioni per la scelta del progetto.

2.1 Proposta di stage

I primi contatti con l'azienda sono avvenuti durante l'evento Stage-IT^[8], durante il quale mi sono potuto incontrare con dei rappresentanti di NETCOM e mi sono state presentate delle proposte di stage. Le proposte delle diverse aziende erano state rese disponibili ai partecipanti prima dell'evento e l'accurata descrizione e motivazione dei progetti offerti dall'azienda mi aveva spinto ad informarmi in maniera più approfondita. In aggiunta a questo avevo già avuto modo di ottenere informazioni su NETCOM e le sue attività. L'anno precedente infatti ho seguito il corso di Amministrazione di Sistema dell'università di Padova tenuto dal Dott. Francesco Clabot, il quale è anche uno dei titolari dell'azienda. Il corso si focalizzava sull'insegnamento di ITIL, un insieme di linee guida e *best practice* per la gestione dei servizi IT. Avendo trovato il corso particolarmente interessante ero curioso di vedere applicate tutte quelle metodologie che avevo fino a quel momento appreso soltanto dal punto di vista teorico, sperando anche di poterle imparare meglio vedendone un esempio di applicazione pratica. Dopo l'evento di Stage-IT ho dunque fissato un colloquio con l'azienda che poi ha portato a concordarci su una particolare offerta di stage. La proposta di stage offertami dall'azienda NETCOM è stata quella di essere inserito all'interno del loro dipartimento DEV per partecipare allo sviluppo di Beryllium, una piattaforma unica per il SAM aziendale. Lo stage ha previsto un impegno della durata di 320 ore, che prevedevano sia una buona componente di formazione su nuove tecnologie, sia una parte di sviluppo dove avrei potuto mettere in pratica le conoscenze acquisite fino a quel momento.

2.2 Beryllium

Beryllium è il nome assegnato dall'azienda NETCOM al progetto al quale mi sono unito per la durata dello stage.

Scopo del prodotto

Lo scopo del progetto Beryllium è quello di sviluppare un'interfaccia unica di gestione per la raccolta, l'utilizzo e la consultazione delle informazioni ricavate da programmi di monitoraggio delle risorse. Questo viene fatto al fine di analizzare automaticamente i *software* utilizzate dagli utenti, per poi controllare se gli utilizzatori dispongono delle licenze necessarie ad utilizzare tali *software*.

Funzioni del prodotto

Il prodotto è composto da diverse parti, suddivise in base alle funzioni che svolgono. A ciascuna componente è stato assegnato il nome di una delle parti di un'orchestra in modo da fornire un nome semplice da utilizzare ma evocativo delle funzionalità della specifica componente.

Conductor - Il programma principale

Il Conductor rappresenta il programma principale. Si occupa di gestire singolarmente tutti i diversi componenti di monitoraggio, raggruppandone poi i risultati per renderli disponibili in cloud. Il Conductor è in grado di controllare l'esecuzione delle diverse componenti, permettendo di stabilire degli *schedule* di esecuzioni per le stesse, o di attivarle quando si desidera.

Opera House - L'interfaccia web

L'Opera House è l'interfaccia web tramite la quale i diversi fornitori possono gestire i loro clienti. I fornitori avranno la possibilità di visualizzare, aggiungere o rimuovere clienti. La visualizzazione dei clienti presenta due viste. La prima presenta una tabella dove ciascuna riga rappresenta un Stock Keeping Unit (SKU)^[§] rilevati e ciascuna colonna un mese di storico. Una cella contiene il numero massimo di licenze usate per il prodotto rilevate in quel mese. La seconda tabella invece ha i dettagli dell'inventario *Hardware/Software*. Tramite questa interfaccia i fornitori avranno anche la possibilità di scaricare i Conductor da installare sui server dei clienti. L'Opera House è accessibile da Internet e risiede in NETCOM.

Musician - Programma di analisi delle risorse

I Musician sono i piccoli programmi che si occupano di ottenere informazioni su un singolo prodotto *software* allo scopo di raccogliere informazioni sul numero di licenze necessarie per ciascuno di essi.

Speaker - WebService

Gli Speaker rappresentano il punto di contatto con il Conductor. Sviluppati seguendo lo standard Simple Object Access Protocol (SOAP)^[§] si occupano di ricevere uno *.zip* contenente i dati raccolti dal Conductor (identificato tramite l'ID fornitore e l'ID cliente, entrambe stringhe) e di salvarne le informazioni su un *Database*. Gli Speaker non effettuano alcun processing delle informazioni ricevute. Questi Web Service^[§] si occupano anche del controllo degli aggiornamenti per il Conductor e i Musician.

Composer - Processing dei dati

Il Composer ha il compito di processare le informazioni ricevute come dati grezzi in modo da renderle leggibili e facilmente interpretabili. Viene eseguito ogni 15 minuti, fornendo i dati presentabili all'interfaccia web. Nel caso sia necessario il Composer ha anche la possibilità di inviare email di report.

2.3 Motivazioni

NETCOM offre diversi servizi multilivello che aiutano le aziende a proteggere, gestire e valorizzare il loro patrimonio, sostenendo il dipartimento IT nelle verifiche di conformità e nell'ottimizzazione del *software* utilizzato in funzione dei requisiti di *business*. L'azienda supporta già i clienti in ogni fase del processo di implementazione del SAM, dall'esecuzione dell'inventario alla realizzazione del piano di consolidamento e ottimizzazione, nel rispetto delle normative vigenti e delle condizioni contrattuali di licenza del vendor. Beryllium si propone di offrire un ulteriore strumento per facilitare la gestione del SAM in particolare a quelle aziende che si occupano di offrire servizi a terzi.

La piattaforma Beryllium infatti permette ad un vendor di recuperare automaticamente e da remoto un gran numero di dati riguardanti le risorse che egli ha reso disponibili ai suoi clienti, fornendo strumenti di monitoraggio avanzati ed elaborandone automaticamente le informazioni mostrandole all'utente finale sotto forma di grafici e tabelle semplici da interpretare.

Beryllium si propone in particolare di semplificare notevolmente il processo di conta delle licenze per i prodotti Microsoft, un campo spesso particolarmente complicato e ostico per le aziende. La soluzione NETCOM si propone di risolvere i problemi causati da un'errata conta delle licenze, infatti il prodotto è in grado di recuperare autonomamente e automaticamente i dati necessari a determinare il numero di licenze necessarie per ciascun prodotto Microsoft per poi successivamente rielaborare questi dati e renderli disponibili all'utente finale in maniera semplice e chiara.

2.4 Obiettivi

Il progetto di stage prevede principalmente lo sviluppo da parte dello studente di una serie di Musician, ovvero dei programmi di monitoraggio che recuperano le informazioni necessarie alla conta delle licenze. Oltre a questo è prevista una parte di formazione su prodotti Microsoft, sul SAM aziendale e infine lo sviluppo di un prototipo di interfaccia web. Prima dell'inizio dello stage è stato redatto con il tutor aziendale un Piano di Lavoro nel quale sono stati definiti i seguenti obiettivi:

- Studio di *business practice* per SAM con riferimenti alla ISO/IEC 19779-1:2012 e licensing Services Provider License Agreement (SPLA)^[g];
- Studio, installazione e gestione di diversi prodotti e servizi Microsoft incluse le tecnologie relative come Exchange^[g], SharePoint^[g], Active Directory, Structured Query Language (SQL)^[g], Domain Name System (DNS)^[g], Dynamic Host Configuration Protocol (DHCP)^[g], Remote Desktop Services (RDS)^[g];
- Studio dell'infrastruttura virtuale e metodi di versionamento di un sistema aziendale;

- Progettazione, sviluppo, test e documentazione di Musician volti alla raccolta di vari dati da *software* più complessi;
- Progettazione e sviluppo di un'interfaccia web, l'Opera House.

2.5 Vincoli

Quando è stato definito il Piano di Lavoro sono anche stati imposti dei vincoli sul lavoro svolto. Questi possono essere divisi in tipologie:

- **Vincoli lavorativi:** in questa categoria rientrano i prodotti attesi al termine dello stage e le limitazioni imposte riguardanti la modalità di lavoro.

I prodotti attesi al termine dello stage erano:

- Da due a sei Musician completamente sviluppati;
- Documento di Analisi dei Requisiti
- Documentazione dei Musician sviluppati, in particolare il codice di ciascun Musician doveva essere accuratamente documentato e, in un documento a parte, dovevano essere date le seguenti informazioni:
 - * Breve descrizione del prodotto Microsoft analizzato;
 - * Licensing del prodotto;
 - * Descrizione delle attività del Musician;
 - * Descrizione dettagliata dell'output del Musician;
 - * Permessi Utente e diritti di accesso richiesti per eseguire il programma.
- Prototipo dell'Opera House, ospitato online su un server aziendale.

- **Vincoli tecnologici:** questa categoria rappresenta tutte le condizioni e i limiti sulla tipologia di tecnologie da utilizzare per sviluppare i prodotti.

I vincoli imposti sono stati:

- Lo sviluppo dei Musician deve avvenire con il .NET framework^[g] 4, è ammesso utilizzare altre versioni del framework in caso di non compatibilità con vecchie versioni dei programmi da analizzare;
- Il linguaggio con cui sviluppare i Musician è il C#, in caso non sia possibile effettuare le operazioni necessarie tramite questo linguaggio è permesso integrare all'interno del codice in C# del codice PowerShell;
- I Musician devono essere realizzati come librerie .dll;
- I Musician devono seguire la struttura decisa dall'azienda per potersi interfacciare con le altre componenti del programma;
- L'interfaccia web deve seguire il mockup^[g] fornito allo studente.

Il rispetto di ciascuno dei vincoli imposti è stato controllato dal tutor aziendale durante lo svolgimento dello stage man mano che venivano realizzati i relativi prodotti.

Capitolo 3

Lo stage

In questo capitolo vengono definite le modalità di svolgimento dello stage, come gli strumenti utilizzati per lo sviluppo, la pianificazione delle attività e degli obiettivi.

3.1 Ambiente e strumenti di sviluppo

Per lo sviluppo dei diversi prodotti è stato scelto di utilizzare l'ambiente Windows, essendo infatti tutti i prodotti da analizzare dei prodotti Microsoft utilizzare Windows e i suoi strumenti ha permesso di semplificare notevolmente il lavoro e il raggiungimento degli obiettivi prefissati.

Di seguito vengono presentati i diversi linguaggi, *software* e framework utilizzati durante lo sviluppo e la produzione della documentazione di progetto.

Linguaggi

- **C#:** C# è un linguaggio di programmazione orientato agli oggetti sviluppato da Microsoft. La sintassi e la struttura del C# riprendono numerosi elementi da linguaggi precedenti, come Delphi, C++ e Java. È stato scelto come linguaggio principale per la realizzazione dei Musician in quanto fortemente integrato nell'ambiente Microsoft. Essendo nato come parte dell'iniziativa .NET il linguaggio si integra perfettamente con il *framework* e presenta numerose librerie utili che interagiscono direttamente con i prodotti Microsoft che devono essere analizzati dai Musician.
- **PowerShell:** PowerShell è una shell con interfaccia a riga di comando che utilizza un linguaggio di *scripting* sviluppato da Microsoft. È basato fortemente sulla programmazione ad oggetti e sul *framework* .NET. L'utilizzo di PowerShell per la realizzazione dei Musician è stato fortemente sconsigliato, e ne è stato permesso l'utilizzo solo in caso di necessità, nel qual caso il codice PowerShell doveva essere invocato da codice C#. PowerShell si è rivelato estremamente utile vista la grande semplicità e le numerosissime funzionalità che offre per interagire con i vari prodotti Microsoft, fornendo quindi un ottimo strumento per effettuare test e verifiche.
- **HTML5:** HTML5 è l'ultima versione dell'HyperText Markup Language (HTML)^[g], un linguaggio di *markup* utilizzato per la formattazione e impaginazione del-

le pagine web. È stato scelto in quanto è lo standard più recente di HTML, presentando numerosi vantaggi sulle precedenti versioni.

- **CSS:** il Cascading Style Sheets (CSS)^[g] è un linguaggio utilizzato per definire la formattazione di documenti HTML, eXtensible HyperText Markup Language (XHTML)^[g] e eXtensible Markup Language (XML)^[g], in particolare per siti e pagine web. È stato utilizzato per definire lo stile delle pagine web dell'Opera House ed è stato scelto in quanto è il linguaggio più diffuso, supportato e comunemente utilizzato per questo scopo.

Framework

- **.NET Framework:** Il *framework* .NET è un ambiente per la creazione, distribuzione ed esecuzione di applicativi. È formato da tre componenti principali: una serie di compilatori per i principali linguaggi supportati da Microsoft; un ambiente di esecuzione chiamato Common Language Runtime (CLR)^[g]; e una libreria di classi. Il suo utilizzo è stato scelto per la sua integrazione nativa con i vari prodotti Microsoft che sono stati utilizzati durante lo stage.
- **Bootstrap:** Bootstrap è un *framework* per la progettazione della parte *front end* di siti e applicazioni web. Esso rende disponibili numerosi template HTML e CSS di comprovata efficienza ed efficacia, garantendone un buon supporto da parte dei vari browser. È stato scelto al fine di semplificare la realizzazione dell'Opera House grazie all'utilizzo dei vari template.

Software di sviluppo

- **Visual Studio 2013 Express for Desktop:** Microsoft Visual Studio è un ambiente di sviluppo integrato sviluppato da Microsoft, che supporta diversi tipi di linguaggio come C, C++, C#, e .NET. Visual Studio è stato scelto come ambiente principale di sviluppo dei Musician del progetto a causa della sua stretta integrazione con il *framework* .NET.
- **Notepad++:** Notepad++ è un semplice e leggero *editor* di testo per Microsoft Windows, che fornisce un supporto per molti linguaggi di programmazione. È stato utilizzato per la scrittura del codice HTML e CSS dell'Opera House in quanto forniva uno strumento rapido e semplice da utilizzare che bene si adattava alle necessità della realizzazione del prototipo.

Software per la documentazione

- **Microsoft Word 2016:** Microsoft Word è un programma di videoscrittura prodotto da Microsoft facente parte della *suite* di Office. È stato utilizzato per scrivere e condividere la documentazione realizzata per l'azienda.
- **TeXstudio:** TeXstudio è un *editor* per L^AT_EX costantemente aggiornato che offre una serie completa di funzionalità per la realizzazione di documenti. È stato utilizzato per realizzare le versioni definitive dei documenti.

3.2 Piano di Lavoro

3.2.1 Obiettivi pianificati

Durante la redazione del Piano di Lavoro sono stati identificati i diversi obiettivi dello stage, che sono poi stati elaborati e divisi secondo la seguente notazione:

- **min:** per i requisiti minimi, vincolanti in quanto obiettivo primario richiesto dal committente;
- **max:** per i requisiti massimi, non vincolanti o strettamente necessari, ma dal riconoscibile valore aggiunto;
- **for:** per i requisiti formativi, rappresentanti valore aggiunto in termini culturali e di conoscenze da acquisire dallo stagista.

Alle sigle precedentemente indicate è stata aggiunta una coppia sequenziale di numeri, creando l'identificativo finale del requisito.

Ora vengono presentati i requisiti così come sono stati definiti e suddivisi nel Piano di Lavoro:

- **Minimi**
 - *min01*: Comprensione del mondo SAM, SPLA e dei prodotti Microsoft di interesse;
 - *min02*: Progettazione, sviluppo, test e documentazione di almeno due Musician.
- **Massimi**
 - *max01*: Progettazione, sviluppo, test e documentazione di sei Musician;
 - *max02*: Miglioramento ed estensione delle funzionalità dei Musician per Exchange e Lync^{gl} già esistenti;
 - *max03*: Sviluppo di un prototipo dell'Opera House
- **Formativi**
 - *for01*: capacità di creare un ambiente microsoft con Exchange, SharePoint, Active Directory, SQL, DNS, DHCP e RDS;
 - *for02*: capacità di utilizzare sistemi di versionamento in contesti aziendali;
 - *for03*: capacità di progettare componenti modulari che dovranno interagire con sistemi di produzione;
 - *for04*: capacità di sviluppare parti di progetti di grandi dimensioni;
 - *for05*: capacità di utilizzare sistemi infrastrutturali VMware.

3.2.2 Pianificazione attività e scadenze

Per raggiungere gli obiettivi prefissati nel Piano di Lavoro sono state individuate una serie di attività suddivise in tre macro-attività principali:

- **Formazione e introduzione al contesto aziendale:** periodo iniziale di formazione dello studente;

- *Introduzione al progetto Beryllium*: durante questa attività verrà presentato allo studente il progetto a cui andrà a lavorare, con particolare attenzione a come le parti da lui sviluppate andranno ad interfacciarsi con il resto del prodotto;
- *Formazione su SAM, ISO/IEC 19779-1:2012, licensing SPLA e prodotti Microsoft*: in questo periodo lo studente verrà formato sui diversi argomenti elencati;
- *Formazione sul sistema aziendale di infrastruttura virtuale*: questa attività prevede la presentazione e l'istruzione dello studente all'utilizzo del sistema di virtualizzazione aziendale;
- *Creazione ambiente di test con Exchange, SharePoint, Active Directory, SQL, DNS, DHCP e RDS*: in questo periodo lo studente verrà formato sulle tecnologie indicate e creerà il suo ambiente di test e, con il supporto dell'azienda, provvederà ad installarvi i diversi programmi e strumenti che andrà ad utilizzare durante lo stage;
- *Introduzione a metodi di versionamento aziendali*: durante questa attività lo studente imparerà ad utilizzare gli strumenti di versionamento selezionati dall'azienda.
- **Sviluppo Musician**: periodo di sviluppo dei Musician, per ciascuno di essi sono state effettuate le seguenti attività.
 - *Progettazione Musician*: periodo di progettazione del Musician;
 - *Sviluppo Musician*: sviluppo effettivo del programma;
 - *Test Musician*: test delle diverse funzionalità del programma e della sua integrazione con il resto del progetto;
 - *Documentazione Musician*: scrittura della documentazione del Musician realizzato, insieme all'aggiunta di commenti al codice prodotto.
- **Realizzazione Opera House e presentazione del lavoro svolto**
 - *Progettazione Opera House*: progettazione del *design* e delle funzionalità offerte dall'interfaccia web;
 - *Sviluppo Opera House*: sviluppo effettivo del sito web;
 - *Presentazione del lavoro svolto*: presentazione finale al tutor aziendale del lavoro svolto.

Queste attività sono state pianificate durante la durata dello stage secondo il diagramma di Gantt riportato in Figura 3.1.

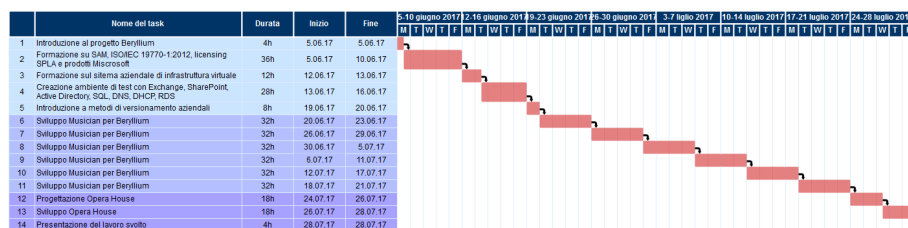


Figura 3.1: Diagramma di Gantt: attività e scadenze definite dal Piano di Lavoro.

3.3 Analisi

Al fine di determinare in maniera il più chiara e precisa possibile gli obiettivi dello stage è stata svolta un'analisi dei requisiti sui prodotti da sviluppare. È stata svolta prima un'analisi generale dell'intero progetto Beryllium, la quale tuttavia non sarà inclusa all'interno di questo testo in quanto solo marginalmente pertinente al lavoro svolto. Casi d'uso e requisiti dell'intero progetto infatti esulano da quelli che erano gli obiettivi posti dallo stage. Verrà invece presentata l'analisi dei requisiti di quelle componenti che sono state effettivamente realizzate durante lo stage, ovvero i Musician e l'Opera House.

3.3.1 Casi d'uso

I casi d'uso sono catalogati come:

UC[numero][caso]

Dove:

- *UC* specifica che si sta parlando di un caso d'uso;
- *Numero* è assoluto e rappresenta un riferimento univoco al caso d'uso in questione;

La breve descrizione di ciascun caso d'uso presenta:

- Gli attori del caso d'uso;
- Lo scopo e la descrizione del caso d'uso;

Di seguito viene presentata la lista degli attori dei casi d'uso:

- *Fornitore*: rappresenta un fornitore delle licenze dei diversi *software* analizzati;
- *Cliente*: rappresenta l'utente che ha acquistato le licenze e fa uso dei *software*;
- *Conductor*: rappresenta il componente Conductor;
- *Opera House*: rappresenta il componente Opera House;
- *Musician*: rappresenta il componente Musician;

Casi d'uso Opera House

Vengono di seguito riportati i casi d'uso dell'Opera House.

UC1 Login

- *Attori*: Opera House;
- *Descrizione*: Lo scopo di questo caso d'uso è di richiedere al Fornitore di effettuare il *login* per accedere all'interfaccia web. Il Fornitore richiede l'accesso all'Opera House, la quale effettua una richiesta di inserimento di *username* e *password*.

UC2 Ricerca clienti

- *Attori:* Fornitore;
- *Descrizione:* Lo scopo di questo caso d'uso è di permettere al fornitore di effettuare una ricerca tra i suoi clienti. L'Opera House fornisce al Fornitore una barra di ricerca dove egli può trovare i suoi clienti.

UC3 Aggiunta/eliminazione/modifica dei clienti

- *Attori:* Fornitore;
- *Descrizione:* Lo scopo di questo caso d'uso è di permettere al fornitore di aggiungere, rimuovere o modificare i clienti. L'Opera House offre al Fornitore un'interfaccia dalla quale aggiungere nuovi clienti.

UC4 Visualizzazione SKU di un cliente

- *Attori:* Fornitore;
- *Descrizione:* Lo scopo di questo caso d'uso è di permettere al fornitore di vedere le informazioni relative agli SKU di un singolo cliente sotto forma di tabella. La tabella ha sulle righe l'elenco degli SKU rilevati e sulle colonne i mesi di storico. Ogni cella contiene il massimo numero di licenze usate per quel prodotto rilevato in quel mese.

UC5 Visualizzazione inventario

- *Attori:* Fornitore;
- *Descrizione:* Lo scopo di questo caso d'uso è di permettere al fornitore di vedere le informazioni di inventario di un singolo cliente. L'Opera House offre al Fornitore una vista dove può controllare le informazioni di inventario.

UC6 Download Conductor

- *Attori:* Fornitore;
- *Descrizione:* Lo scopo di questo caso d'uso è di permettere al fornitore di scaricare dall'Opera House un Conductor da installare sui *server* del cliente.

Casi d'uso Musician

Vengono di seguito riportati i casi d'uso dei Musician.

UC7 Recupero dei dati

- *Attori:* Musician;
- *Descrizione:* Lo scopo di questo caso d'uso è di permettere al Musician di raccogliere le informazioni del prodotto Microsoft di cui si occupa. Il Musician, una volta avviata la scansione, interroga il computer per ricavarne le informazioni. Al termine delle operazioni il Musician indica al Conductor se il recupero delle informazioni ha avuto successo o meno.

UC8 Attivazione/Disattivazione file di log

- *Attori*: Conductor;
- *Descrizione*: Lo scopo di questo caso d'uso è di permettere al Conductor di attivare o disattivare la creazione di file di log durante l'esecuzione del Musician. Il Conductor, quando esegue il Musician, gli passa come condizione se generare o meno un file di log.

UC9 Salvataggio delle informazioni ricavate

- *Attori*: Musician;
- *Descrizione*: Lo scopo di questo caso d'uso è di permettere al Musician di salvare le informazioni che ha recuperato. Il Musician, una volta che ha ottenuto le informazioni, le inserisce in un file .csv.

3.3.2 Requisiti

Vengono ora presentati i requisiti emersi durante l'analisi dei casi d'uso e i requisiti discussi nelle riunioni interne con l'azienda. Si è deciso di inserire i requisiti in una tabella dei requisiti per permettere una consultazione agevole degli stessi. La tabella dei requisiti li presenta fino al massimo livello di dettaglio insieme alle loro caratteristiche, in particolare ne specifica:

- Codice (secondo le regole indicate successivamente);
- Categoria di appartenenza fra:
 - Obbligatorio, per i requisiti irrinunciabili;
 - Desiderabile, per i requisiti non strettamente necessari ma che offrono un valore aggiunto riconoscibile;
 - Opzionale, per i requisiti relativamente utili o contrattabili in seguito.
- Una descrizione esaustiva del requisito;
- La fonte dal quale il requisito ha avuto origine, sia essa l'analisi diretta dei casi d'uso o il dialogo con il tutor aziendale.

Catalogazione requisiti

I requisiti sono catalogati come segue:

$$R[\text{Categoria}][\text{tipo}][\text{numero}]$$

Dove:

- *R* specifica che si tratta di un requisito;
- *Categoria* indica se si tratta di un requisito:
 - Obbligatorio: 1;
 - Desiderabile: 2;
 - Opzionale: 3;
- *Tipo* indica la tipologia del requisito, che può essere:

- Di vincolo: V;
- Funzionale: F;
- Di qualità: Q;
- *Numero* è assoluto e rappresenta un riferimento univoco al requisito in questione.

Tabella dei requisiti

Vengono di seguito presentate le tabelle dei requisiti, suddivise per tipologia.

Requisiti di vincolo

Codice	Categoria	Descrizione	Fonti
R1V1	Obbligatorio	I Musician devono essere sviluppati utilizzando il linguaggio C#	Riunione interna
R1V2	Obbligatorio	I Musician devono essere sviluppati come librerie .dll	Riunione interna
R1V3	Obbligatorio	I Musician devono essere sviluppati con il <i>framework</i> .NET 4.0	Riunione interna

Tabella 3.1: Requisiti di vincolo.

Requisiti funzionali

Codice	Categoria	Descrizione	Fonti
R1F4	Obbligatorio	L'Opera House richiede il <i>login</i> al fornitore prima di garantirgli l'accesso	UC1
R1F5	Obbligatorio	L'Opera House da al fornitore la possibilità di effettuare una ricerca tra i suoi clienti	UC2
R1F6	Obbligatorio	L'Opera House permette al fornitore di aggiungere, eliminare e modificare i suoi clienti	UC3
R1F7	Obbligatorio	L'Opera House offre al fornitore una vista degli SKU di un cliente	UC4
R1F8	Obbligatorio	L'Opera House offre al fornitore una vista dell'inventario di un cliente	UC5
R3F9	Desiderabile	L'Opera House permette al fornitore di scaricare un'istanza del Conductor da installare sui <i>server</i> del cliente	UC6

R1F10	Obbligatorio	Il Musician effettua il recupero dei dati riguardanti il prodotto Microsoft di cui è responsabile, indicando al Conductor se l'operazione ha avuto successo o meno	UC7
R1F11	Obbligatorio	Il Conductor indica al Musician se salvare o meno un file di log	UC8
R1F12	Obbligatorio	Il Musician salva le informazioni ricavate su un file .csv	UC9

Tabella 3.2: Requisiti funzionali.

Riepilogo

Categoria	Obbligatorio	Opzionale	Desiderabile
Funzionale	8	0	1
di Vincolo	3	0	0

Tabella 3.3: Riepilogo dei requisiti.

3.4 Progettazione

Per la progettazione dei Musician è stata stabilita, insieme all'azienda, una struttura comune per tutti i diversi programmi.

Ciascun Musician è rappresentato da una classe (Figura 3.2), il cui nome corrisponde a quello del prodotto Microsoft che deve analizzare. Viene di seguito presentata la struttura generale.

Musician
<u>-musicianName : string</u>
<u>-Logs_active : boolean</u>
<u>+Log(line : string) : void</u>
<u>+Gather(args : string[], Logs_status : boolean = false) : boolean</u>

Figura 3.2: Classe Musician generica.

Descrizione

Questa classe rappresenta un singolo Musician. Ciascuno di essi è dedicato ad un singolo prodotto Microsoft e il loro scopo è raccogliere le informazioni di licensing del programma di cui si occupano.

Utilizzo

La classe viene compilata per formare una libreria .dll le cui funzionalità saranno poi invocate da una parte del Musician chiamata Tester.

Attributi

- - musicianName : string: campo dati statico che contiene il nome del Musician. Il nome del Musician è uguale a quello del prodotto che deve analizzare.
- - Logs_active : boolean: campo dati statico che indica se i log sono attivi o meno. Un valore *true* indica di al programma di generare il file di log, un valore *false* indica di non generarlo.

Metodi

- + Log(line : string) : void: questa funzione statica viene utilizzata per generare il file di log. Quando viene invocata per prima cosa controlla se i log sono attivi. In caso lo siano procede a creare il file di log nel caso non esista e poi ad inserirci la stringa di testo contenuta nel parametro *line*.
- + Gather(args : string[], Logs_status : boolean = false) : boolean: questa è la funzione principale di tutti i Musician. Gather si occupa della raccolta dei dati sul prodotto Microsoft che il Musician deve analizzare quando viene invocata gli vengono passati i parametri args, che possono per esempio contenere un nome alternativo per il file di *output* diverso da quello di *default*, e il booleano Logs_status che indica se creare o meno il file di log durante questa esecuzione del Musician. La funzione ritorna un booleano che indica il risultato dell'esecuzione del Musician, un valore *true* indica che l'esecuzione ha avuto successo, un valore *false* indica che ci sono stati dei problemi.

3.5 Documentazione

Durante lo svolgimento dello stage è stato richiesto di produrre della documentazione approfondita dei diversi prodotti sviluppati. In particolare è stato richiesto:

- Documento di Analisi dei Requisiti per l'intero progetto;
- Un documento, in inglese, che forniva per ciascun Musician realizzato:
 - Descrizione generale del prodotto;
 - Licensing del prodotto;
 - Descrizione del Musician e delle sue attività;
 - Descrizione dettagliata dell'*output*;
 - Permessi Utente e diritti di accesso richiesti;
- Tutto il codice prodotto doveva essere accuratamente commentato.

3.6 Test

La fase di test dei prodotti sviluppati ha seguito un andamento diverso dallo standard. I programmi sviluppati infatti non effettuano operazioni complesse, e la loro esecuzione è estremamente lineare. Le maggiori problematiche per il funzionamento dei Musician

derivano dunque da aspetti esterni al programma stesso, come permessi dell'utente, configurazioni strane dei prodotti da analizzare o versioni non compatibili. Per questo motivo i test svolti si sono focalizzati prevalentemente sull'eseguire ripetutamente il programma con varie tipologie di utenti, su dispositivi diversi e su versioni diverse dei *software* da analizzare. Vista l'enorme quantità di possibili varianti l'attività di test è stata per necessità incompleta. Ciascun Musician è stato ripetutamente eseguito su tutte le versioni dei programmi disponibili all'azienda e con numerose tipologie di utenti. Per citare un esempio il Musician di SharePoint è stato testato su tre versioni diverse del prodotto, su tre *server* diversi e con una complessa gamma di utenti con varie tipologie di permessi locali, di Active Directory, *database* SQL (su cui si basano i siti web realizzati con SharePoint), farm di SharePoint, Site Collections e siti web. Tutti questi fattori hanno quindi contribuito ad una fase di test non del tutto soddisfacente e programmabile, che rappresenta la principale problematica dei prodotti realizzati.

Capitolo 4

Musician

In questo capitolo verranno descritti in dettaglio tutti i Musician che sono stati sviluppati. Per ciascun Musician verrà fornita: una descrizione generale del prodotto che deve analizzare; un'introduzione al licensing del prodotto; una descrizione del lavoro svolto dal Musician; l'output del Musician; i permessi richiesti per far andare il programma.

4.1 Microsoft Exchange

Descrizione del prodotto

Microsoft Exchange Server è un *software* di *networking* che permette la collaborazione online tra utenti di un'organizzazione. Le funzionalità principali del *software* sono la gestione dell'email centralizzata, l'amministrazione di calendari e agende che possono essere condivise tra utenti della rete e la pianificazione condivisa di attività di *business*. Il *client* più comunemente utilizzato per connettersi ad un *server* di Exchange è Microsoft Outlook, che fa parte della suite di Office, attraverso il protocollo proprietario *MAPI* (*Messaging Application Programming Interface*). Vengono tuttavia offerti anche altri mezzi per connettersi al *server*, come l'interfaccia web *OWA* (*Outlook Web Access*) o *software* di terze parti che utilizzano protocolli standard. Una delle caratteristiche più importanti di Exchange è la sua stretta integrazione con Active Directory, che permette di gestire facilmente strutture complesse di utenti e gruppi di distribuzione implementati al livello di unità organizzative. [21]

Licensing

Il licensing di Microsoft Exchange può essere diviso in due parti, il licensing relativo al *server* dove è installato l'applicativo e il licensing relativo agli utenti che effettuano l'accesso al *server*.

Server licensing:

Per ottenere la licenza di un *server* Exchange è necessario acquistare una licenza per ogni istanza del *software* che viene fatta girare sul *server*. Ci sono due tipi di licenze *server*:

- **Standard:** adatta a gestire le caselle di posta per organizzazioni di medie dimensioni. Può anche essere appropriata per ruoli diversi in un *deployment* di maggiori dimensioni. Questa versione supporta da uno fino a cinque *database* di caselle di posta.
- **Enterprise:** adatta per organizzazioni più grandi che potrebbero richiedere un maggior numero di *database* per gestire le caselle di posta. Questa versione supporta fino a 100 *database*.

CAL (Client Access License) licensing:

Ogni utente o dispositivo che accede il *software* installato sul *server* necessita di una licenza CAL. Ci sono due tipologie di licenze CAL di Exchange, entrambe le quali possono essere utilizzate con entrambe le versioni della licenza *server*.

- **Standard:** questa licenza è fatta per permettere agli utenti comuni di essere più produttivi sfruttando i servizi resi disponibili dal *server* Exchange. Ogni utente o dispositivo che desidera accedere al *server* necessita di questa licenza.
- **Enterprise:** questa licenza offre l'accesso a strumenti e funzioni più avanzate, come la protezione e archiviazione delle informazioni integrata. Il CAL Enterprise di Exchange viene venduto come licenza additiva rispetto a quella standard, il che significa che un utente per poter accedere alle funzionalità Enterprise deve acquistare sia una licenza Standard sia una Enterprise.

Le licenze CAL possono essere ulteriormente divise tra *User CAL* e *Device CAL*. Un *Device CAL* è una licenza legata al dispositivo che viene utilizzato per effettuare l'accesso al *server*. Una singola licenza *Device CAL* permette ad un numero potenzialmente infinito di utenti di avere accesso al *server*, a patto che effettuino l'accesso attraverso il dispositivo che dispone della licenza e che le connessioni non siano essere concorrenti. Lo *User CAL* è una licenza legata all'identità dell'utente che effettua l'accesso al *server*. Una singola licenza *User CAL* permette ad un utente di accedere al *server* tramite un numero potenzialmente infinito di dispositivi a patto che le connessioni non siano concorrenti. [22]

Descrizione del Musician

Il Musician di Exchange utilizza gli strumenti forniti dai *cmdlet* di PowerShell per recuperare informazioni sull'installazione corrente del *server* di Exchange e per recuperare la lista di User o *Device CAL* di quali necessitano gli utenti.

Per via di una richiesta aggiuntiva di un cliente il Musician offre inoltre delle ulteriori funzionalità. Il cliente necessitava di recuperare le informazioni sugli accessi degli utenti alle diverse caselle di posta, in particolare l'identità dell'utente e il dispositivo dal quale è stato effettuato l'accesso. Sono state sviluppate tre diverse soluzioni per recuperare queste informazioni.

Active Sync

La prima soluzione sviluppata è stata basata su *Active Sync*. Exchange memorizza automaticamente alcune informazioni riguardanti tutti i dispositivi che si sono collegati ad una casella di posta specifica tramite il servizio *Active Sync*, il Musician quindi utilizza dei *cmdlet* di PowerShell per recuperare le informazioni. Questa soluzione

tuttavia fornisce risultati incompleti in quanto recupera informazioni soltanto dalle connessioni effettuate tramite *Active Sync*, e non da altri tipi di connessioni come per esempio quelle tramite OWA.

Audit Logs

La seconda soluzione sviluppata è basata sugli *Audit Log* di Exchange. Gli *Audit Log* sono dei record particolarmente dettagliati che memorizzano le operazioni svolte dai diversi utenti sulle caselle di posta. Possono essere impostati per memorizzare diversi tipi di informazioni effettuate sia dal possessore della casella di posta sia da altri utenti che hanno accesso alla casella di posta, come gli amministratori. Questi log mostrano chi ha effettuato l'accesso alla casella di posta, quando è stato effettuato e da che dispositivo. Il problema che è stato individuato con questa soluzione è che gli *Audit Log* sono di *default* disattivati. Questo significa che, sebbene recuperino tutte le informazioni richieste dal cliente, questa soluzione non è sempre applicabile visto che la maggior parte dei clienti non tengono attivati gli *Audit Log* per le caselle di posta, o potrebbero averli attivati solo per alcune di esse.

IIS Logs Parsing

L'ultima soluzione sviluppata è stata quella di effettuare il *parsing* dei log di IIS (Internet Information Services). IIS memorizza una grande quantità di log che includono anche le connessioni alle caselle di posta. Effettuando il *parsing* e filtrando i log è stato possibile recuperare solo le informazioni desiderate. Questa opzione è sempre applicabile e soddisfa pienamente i requisiti posti dal cliente, tuttavia, visto che i log di IIS oltre ad essere molto numerosi sono anche di dimensioni considerevoli, il processo di *parsing* è estremamente lungo e costoso.

Output del programma

L'*output* del Musician di Exchange è composto da diversi file:

- *ExchangeUsers.csv*: questo file contiene una lista dettagliata di tutti gli utenti del server Exchange, specificando per ciascuno di loro:
 - Domain e Username: lo *username* dell'utente e il dominio di cui fa parte;
 - Primary SMTP (Simple Mail Transfer Protocol) Address: l'indirizzo mail primario dell'utente;
 - Last Logon Timestamp: il timestamp dell'ultima volta in cui l'utente ha effettuato l'accesso alla casella di posta;
 - Need Enterprise: se l'utente necessita o meno di una CAL Enterprise.
- *Report-Exchange.csv*:
- *Report-IISLogs.csv*:
- *ActiveSync-Exchange.csv*:
- *Device-Exchange.csv*:
- *Exchange.log*: il log di Exchange memorizza informazioni sull'esecuzione del programma, incluso il verificarsi di eventuali errori.

Permessi Utente e diritti di accesso richiesti

Per riuscire a recuperare tutte le informazioni desiderate l'utente che esegue il programma deve disporre di diritti di accesso da amministratore sul *server* dove viene eseguito il programma.

4.2 Microsoft Remote Desktop Services

Descrizione del prodotto

RDS, conosciuto precedentemente come *Terminal Services*, è un servizio che permette all'utente di connettersi e prendere il controllo di un computer remoto, una macchina virtuale o una connessione di rete. Con RDS vengono trasmessi al *server* soltanto gli *input* dell'utente, mentre l'esecuzione degli stessi avviene sul *server*.

RDS accelera ed estende l'accesso ai desktop e il *deployment* delle applicazioni a qualsiasi dispositivo, migliorando l'efficienza dei lavoratori in remoto, aiutando allo stesso tempo a mantenere sicure e private informazioni critiche e semplificando la conformità alle normative. RDS rende disponibile l'infrastruttura desktop virtuale (VDI), sessioni desktop e applicazioni, permettendo agli utenti di lavorare ovunque siano. Microsoft RDS fornisce tre forme di *deployment* per permettere ai clienti di avere flessibilità e utilizzare il tipo di VDI adatto ai propri utenti, tutto tramite una singola piattaforma. I clienti possono ospitare sessioni desktop, macchine virtuali condivise o personali. [31]

Licensing

Le policy per il licensing di Windows RDS richiedono che chi desideri usufruire di questi servizi, in aggiunta all'acquisto di una Windows Server Client Access License, acquisti anche una licenza RDS CAL per ciascun utente che acceda direttamente o indirettamente a qualsiasi funzionalità fornita da RDS, o che acceda, direttamente o indirettamente, al *software* del *server* per interagire con una interfaccia grafica utente (GUI) utilizzando funzionalità di RDS o qualsiasi tecnologia di terze parti. Nessun RDS CAL è richiesto per un massimo di due utenti che effettuino l'accesso a istanze del *software server* per motivi amministrativi.

Sono disponibili i seguenti tre tipi di CAL per RDS:

- **RDS Device CAL:** Permette ad un singolo dispositivo (utilizzato da qualunque utente) di accedere alle funzionalità di RDS su un *server* qualsiasi;
- **RDS User CAL:** Permette ad un singolo utente (utilizzando qualunque dispositivo) di accedere alle funzionalità di RDS su un *server* qualsiasi;
- **RDS External Connector:** Permette ad una moltitudine di utenti esterni di accedere ad un singolo *server* RDS. Nel caso l'organizzazione possieda più di un *server* sarà necessario acquistare più connettori esterni in aggiunta a qualsiasi Windows Server External Connector già richiesto.

Microsoft permette di combinare le licenze RDS *Device* e *User CAL* in modo che abbiano accesso contemporaneamente ad uno stesso *software server*. È possibile riassegnare permanentemente un *Device CAL* ad un altro dispositivo o uno *User CAL* ad un altro utente. È inoltre concesso riassegnare temporaneamente un *Device CAL*

ad un dispositivo mentre quello originario è fuori servizio, o riassegnare uno *User CAL* ad un lavoratore temporaneo nel caso quello originale sia assente. [32]

Descrizione del Musician

Il Musician di RDS utilizza i *Directory Services* per recuperare la lista di utenti che fanno parte del gruppo *Administrators* o *Remote Desktop Users*, in quanto soltanto loro hanno i permessi necessario per connettersi al *server* tramite RDS. In questo modo ottiene la lista completa di tutti coloro che possono effettuare l'accesso al *server* tramite RDS.

Output del programma

Il Musician di RDS restituisce in *output* due file:

- *RDS-Report.csv*: Questo file contiene tutte le informazioni recuperate dal Musician, quindi elenca tutti i *Sam Account Names* di tutti gli utenti che possiedono i diritti necessari ad effettuare l'accesso al *server* tramite RDS.
- *RDS.log*: Il log di RDS memorizza informazioni sull'esecuzione del programma, incluso il verificarsi di eventuali errori.

Permessi Utente e diritti di accesso richiesti

Per poter recuperare tutte le informazioni necessarie l'utente che esegue il programma deve far parte del dominio di Active Directory e disporre di diritti di accesso di livello amministratore sulla macchina dove viene eseguito il Musician.

4.3 Microsoft SharePoint

Descrizione del prodotto

Microsoft SharePoint è una piattaforma *software* CMS (Content Management System) che permette di creare e distribuire siti web con scopo di *business* (*Intranet*) che possono poi successivamente venire resi disponibili anche online come normali siti web. Il focus principale di SharePoint è la condivisione di informazioni e/o documenti. È possibile creare liste, *repository* di documenti, calendari sincronizzati tramite Outlook e molte altre cose. SharePoint è anche completamente integrato con il pacchetto Microsoft Office e offre diverse soluzioni come il versionamento dei documenti. [27]

Licensing

Il licensing di SharePoint varia in base alla versione di SharePoint che viene utilizzata.

- **SharePoint Online**: le licenze di SharePoint online sono basate sugli utenti. Possono essere acquistate singolarmente o come parte di Office 365.
- **On-Premises**: i siti Intranet sono basati su un modello di licenze Server/CAL. È necessario acquistare una licenza SharePoint *server* per ciascuna istanza funzionante del *software*, ed sono necessari dei CAL per ciascun utente o dispositivo che effettui l'accesso al *server*.

- *SharePoint Standard CAL*: CAL Standard che rende disponibili le funzionalità di base di SharePoint.
- *SharePoint Enterprise CAL*: CAL Enterprise che permette all'utente di accedere a tutte le funzionalità disponibili.

È importante far notare che le licenze CAL di SharePoint sono additive, ovvero per avere accesso alle funzionalità Enterprise un utente/dispositivo deve possedere sia una licenza SharePoint CAL standard, sia una Enterprise.

Per SharePoint 2016 nel caso si tratti di siti Extranet o Internet non è necessario acquistare dei CAL per utenti esterni. Sono considerati utenti esterni tutti quegli utenti che non sono impiegati, contrattori o agenti dell'azienda o di aziende affiliate. [28]

Descrizione del Musician

Il Musician di SharePoint inizia controllando le chiavi di registro per recuperare informazioni sulla versione di SharePoint correntemente installata sul *server*. Una volta fatto questo itera tra tutte le Web Application, i Site e i Web per recuperare informazioni sugli indirizzi URL di tutti i site e i web, i *database* del contenuto a loro associati, la loro zona (Intranet/Extranet/Internet), la possibilità o meno per utenti anonimi di accedere al sito e la lista di tutti i gruppi e utenti con diritti di accesso o modifica nel sito.

Output del programma

Il Musician di SharePoint restituisce in *output* due file:

- *SharePoint-Report.csv*: questo file contiene tutte le informazioni recuperate dal Musician, quindi elenca:
 - La versione di SharePoint installata;
 - La versione della build della Farm di SharePoint;
 - L'URL dei Siti;
 - I *database* del contenuto dei siti;
 - L'URL dei siti web;
 - L'Anonymous state dei siti web (se il sito permette o meno l'accesso ad utenti anonimi)
 - La lista dei gruppi e degli utenti dei siti web;
- *SharePoint.log*: Il log del Musician di SharePoint memorizza informazioni sull'esecuzione del programma, incluso il verificarsi di eventuali errori.

Permessi Utente e diritti di accesso richiesti

Per essere in grado di recuperare tutte le informazioni l'utente che esegue il programma deve avere i seguenti permessi:

- L'utente deve avere permessi di accesso di livello almeno SPDataAccess al *database* AdminContent, il *database* di configurazione e tutti i *database* del contenuto dei diversi siti web;
- L'utente deve essere un amministratore di tutte le collezioni di siti web delle quali si desiderano recuperare le informazioni;
- L'utente deve avere la possibilità di eseguire il programma come amministratore.

4.4 Dynamic Host Configuration Protocol

Descrizione del prodotto

Il DHCP è un protocollo *client/server* che fornisce automaticamente ad un *host* IP (Internet Protocol) il suo indirizzo IP e ulteriori informazioni di configurazione correlati come la maschera del *subnet* e il *gateway* di *default*. Il DHCP viene utilizzato per gestire semplicemente gli indirizzi IP dei dispositivi all'interno di una rete, visto che permette di configurare con facilità delle impostazioni generale e poi gestisce tutto quanto automaticamente. Il *server* DHCP mantiene una riserva di indirizzi IP e permessi e una lista di indirizzi per tutti i client che hanno attivato il servizio DHCP che si collegano alla rete. Dato che gli indirizzi IP sono dinamici e non statici quegli indirizzi che non sono più in uso vengono automaticamente restituiti alla riserva del DHCP per poter essere riallocati.

L'amministratore della rete determina quali *server* DHCP mantengano le informazioni sulla configurazione TCP/IP e forniscano gli indirizzi e configurazioni ai *client* come offerte di permessi.

Il *server* DHCP mantiene tutte le informazioni in un *database* che include:

- Parametri validi di configurazione TCP/IP per tutti i dispositivi presenti nella rete;
- Indirizzi IP validi, mantenuti in una riserva per poter essere assegnati ai *client*, e eventuali indirizzi esclusi.
- Indirizzi IP riservati o associati a particolari *client* DHCP. Questo permette di assegnare in maniera diretta un singolo indirizzo IP ad uno specifico *client* DHCP.
- La durata dei permessi, o la quantità di tempo durante la quale un indirizzo IP può essere utilizzato prima che venga richiesto un rinnovamento del permesso.

Un *client* DHCP quando accetta un'offerta di permesso riceve:

- Un indirizzo IP valido per il *subnet* al quale si sta connettendo.
- Le opzioni DHCP richieste, che sono parametri aggiuntivi che il *server* DHCP è impostato per fornire ai suoi *client*. Alcuni esempio di opzioni DHCP sono il *Router* (*gateway* di *default*), i *server* DNS e il nome di dominio DNS.

Licensing

Il licensing del DHCP viene gestito tramite *Device* e *User CAL*.

- **Device CAL:** il licensing tramite *Device CAL* richiede che ciascun dispositivo che riceva un'offerta dal DHCP abbia una licenza CAL. Questo permette di avere un singolo CAL per più utenti, a patto che questi utilizzino lo stesso dispositivo.
- **User CAL** il licensing tramite *User CAL* richiede che ciascun utente che riceva un offerta dal DHCP abbia una licenza CAL. Questo permette ad un singolo utente di utilizzare più dispositivi.

[9]

Descrizione del Musician

Il Musician del DHCP utilizza le API DHCP di Microsoft per enumerare tutti i clienti presenti sul *server* DHCP che sono connessi al dominio di Active Directory, o tramite le interfacce di rete. Il Musician prima di tutto recupera la lista dei *server* DHCP che riesce a trovare, poi recupera tutti i *subnet* presenti nei *server* DHCP e infine ottiene la lista di tutti i *client* DHCP presenti in ciascun *subnet*.

Output del programma

Il Musician del DHCP restituisce in *output* due file:

- *DHCP-Report.csv*: questo file contiene tutte le informazioni recuperate dal Musician, quindi elenca per ciascun *client*:
 - L'indirizzo IP del *server*;
 - L'ID dell'indirizzo del *subnet*;
 - L'indirizzo IP del *client*;
 - L'Hostname del *client*;
 - Il Mac Address del *client*.
- *DHCP.log*: Il log del Musician del DHCP memorizza informazioni sull'esecuzione del programma, incluso il verificarsi di eventuali errori.

Permessi Utente e diritti di accesso richiesti

Per riuscire a recuperare tutte le informazioni necessarie l'utente che esegue il programma deve far parte del gruppo amministratori del DHCP del dominio di Active Directory, oltre ad avere diritti da amministratore sulla macchina dove viene eseguito il programma.

4.5 Microsoft Domain Name System

Descrizione del prodotto

DNS è un sistema di nominazione gerarchico e decentralizzato per computer, servizi e altre risorse collegate ad Internet o ad una rete privata. Questo sistema associa diverse

informazioni a dei nomi di dominio assegnati a ciascuna delle entità partecipanti. Il suo compito principale tuttavia è quello di tradurre dei nomi più facilmente memorizzabili nei loro corrispondenti indirizzi IP numerici necessario per la localizzazione e l'identificazione di servizi e dispositivi con i loro protocolli di rete. Il DNS è un componente essenziale delle funzionalità di Internet in quanto rende disponibile un servizio di directory distribuito a livello mondiale, ed è in uso dal 1985.

Il DNS delega la responsabilità di assegnare i nomi di dominio e mappare quei nomi alle risorse Internet designando dei *server* di nomi autoritari per ciascun dominio. Gli amministratori di rete possono delegare l'autorità di sotto-domini del loro namespace allocato ad altri *server* di nomi. Questo meccanismo fornisce un servizio distribuito e resistente agli errori ed è stato ideato per evitare un singolo *database* centrale.

Il DNS specifica inoltre le funzionalità tecniche del servizio di *database* che ne forma il nucleo. Definisce anche il protocollo DNS, una dettagliata descrizione della struttura dei dati e degli scambi di dati utilizzati dal DNS, come parte della *Internet Suite Protocol*. Storicamente altri servizi di *directory* che hanno preceduto il DNS non erano scalabili a livelli globali in quanto erano basati su file di testo, prevalentemente il risolutore HOSTS.TXT.

Internet mantiene due *namespace* principali, la gerarchia dei nomi di dominio e lo spazio degli indirizzi IP. Il DNS mantiene la gerarchia dei nomi di dominio e fornisce dei servizi di traduzione tra questa e lo spazio degli indirizzi. I name *server* di Internet a un protocollo di comunicazione implementano il DNS. Un name *server* DNS è un *server* che memorizza dei record DNS per un dominio e si occupa di rispondere alle domande che gli vengono poste interrogando il suo *database*. [7]

Licensing

Il licensing dei servizi DNS può essere gestito tramite *Device* e *User CAL*.

- **Device CAL:** il licensing tramite *Device CAL* richiede che tutti i dispositivi identificati che ricevono un indirizzo IP dal DNS abbiano una licenza CAL. Questo permette di avere una singola licenza CAL per più utenti, ma questi utenti devono utilizzare il medesimo dispositivo.
- **User CAL:** il licensing tramite *User CAL* richiede che ciascun utente identificato che acceda ai servizi forniti dal DNS abbia una licenza CAL. Questo permette ad un singolo utente di accedere ai servizi DNS tramite più dispositivi con un'unica licenza.

Descrizione del Musician

Il Musician del DNS utilizza le WMI di Microsoft per recuperare tutti i record DNS presenti sul *server* e compilare una lista di tutte le coppie Host-IP memorizzate nel DNS.

Il Musician per prima cosa recupera una lista di tutte le zone di forward-lookup presenti nel *server* DNS. A questo punto recupera tutti i record presenti in queste zone e filtra quelle di tipologia A,AAAA e CNAME, che sono poi utilizzate per scoprire le coppie Host-IP facendole una query diretta al DNS stesso.

Output del programma

Il Musician del DNS restituisce in *output* due file:

- *DNS-Report.csv*: Questo file contiene tutte le informazioni recuperate dal Musician, in particolare fornisce un elenco di tutte le coppie HostName-IP Address recuperate;
- *DNS.log*: Il log del Musician DNS memorizza informazioni sull'esecuzione del programma, incluso il verificarsi di eventuali errori.

Permessi Utente e diritti di accesso richiesti

Per essere in grado di recuperare tutte le informazioni necessarie l'utente che esegue il Musician del DNS deve avere diritti da amministratore sul *server* dove il DNS è installato.

4.6 Microsoft Office

Descrizione del prodotto

Microsoft Office è una suite di applicazioni, *server* e servizi Office sviluppati da Microsoft. Il pacchetto Office include diversi programmi, presentati qui brevemente:

- **Microsoft Word**: un programma per la realizzazione di documenti di testo;
- **Microsoft Excel**: un programma per la realizzazione di fogli elettronici;
- **Microsoft PowerPoint**: un programma per la realizzazione di presentazioni multimediali;
- **Microsoft OneNote**: un programma per gestire un blocco note;
- **Microsoft Outlook**: un *client* per la posta elettronica;
- **Microsoft Access**: un programma per creare, amministrare e visualizzare *database*;
- **Microsoft Publisher**: un programma che offre la possibilità di creare volantini, siti web, lettere, biglietti attestati e vari altri tipi di progetti di pubblicazione.

[23]

Licensing

Ci sono diverse versioni di Office che offrono diversi servizi e funzionalità.

- **Office Standard**: Office Standard offre un set completo di strumenti per migliorare la produttività individuale, incluse una semplice gestione di e-mail, documenti, fogli di calcolo e presentazioni professionali.
- **Office Professional Plus**: Office Professional Plus offre un set completo di strumenti per migliorare la produttività personale e di gruppo. Oltre alle funzionalità della versione standard offre un'integrazione migliorata con altri soluzioni *server* di Microsoft come SharePoint, Exchange e Skype for Business.

- **Office 365 ProPlus:** Office 365 ProPlus offre i medesimi servizi di Office Professional Plus, ma installati localmente. È offerto come un servizio a sottoscrizione e ciascuna licenza permette un'installazione contemporanea dei programmi su di un massimo di cinque dispositivi.

[24]

Descrizione del Musician

Il Musician di Office legge le chiavi di registro per recuperare le informazioni relative alla versione di Microsoft Office correntemente installata sul computer.

Output del programma

Il Musician di Office restituisce in *output* due file:

- *Office-Report.csv*: questo file contiene tutte le informazioni recuperate dal Musician sulle versioni di Office correntemente installate:
 - Display Name: Il nome dell'installazione di Office;
 - Version: La versione dell'installazione di Office.
- *Office.log*: Il log del Musician di Microsoft Office memorizza informazioni sull'esecuzione del programma, incluso il verificarsi di eventuali errori.

Permessi Utente e diritti di accesso richiesti

Per riuscire a recuperare le informazioni richieste l'utente che esegue il Musician di Office deve avere diritti da amministratore sulla macchina dove viene lanciato il programma.

4.7 Microsoft Project

Descrizione del prodotto

Microsoft Project è un prodotto *software* dedicato al Project Management. È stato ideato per assistere un project manager nello sviluppo di un piano di progetto, l'assegnazione di risorse a specifici task, il tracciamento dei progressi, la gestione del budget e l'analisi dei workload. Project è parte della famiglia di Microsoft Office ma non è mai stato incluso in nessuna delle suite di Office. [25]

Licensing

Microsoft Project offre un licensing flessibile per soluzioni di Project Portfolio Management (PPM) online e in-loco, permettendo di mantenere e raggiungere efficacemente priorità strategiche.

Project Cloud-based Solutions

Le soluzioni cloud di Project hanno un licensing su base utente. Possono essere acquistate come piani standalone o come servizi aggiuntivi a piani di Office 365.

- **Project Online Premium:** permette alle organizzazioni di iniziare a dare priorità a investimenti di Project Portfolio e fornire il valore di *business* atteso;
- **Project Online Professional:** permette di mantenere i diversi progetti, risorse e team organizzati. Offre la possibilità di pianificare facilmente ed efficacemente progetti, tracciarne lo stato e collaborare con altri indipendentemente dalla locazione;
- **Project Online Essentials:** fornisce le capacità necessarie ai membri di un team per gestire *task*, inserire schede di attività e collaborare con i colleghi.

Project Solution For On-premises

Le soluzioni Project in-locò sono disponibili tramite Microsoft Volume Licensing. Il licensing di Project Server 2016 si basa su un modello Server/CAL. Le soluzioni PPM richiedono una licenza Project Server 2016 per ciascuna istanza attiva del *software*, ed è richiesta una licenza CAL per ciascun utente o dispositivo che accede ad un *server* di Project.

- **Project Server 2016:** aiuta le organizzazioni a dare priorità agli investimenti di Project Portfolio e fornire il valore di *business* atteso;
- **Project Server CAL 2016:** è richiesta per ciascun utente o dispositivo che si connetta ad un Project Server 2016;
- **Project Professional 2016:** permette di iniziare facilmente a collaborare ad un progetto. Include una licenza Project Server CAL 2016;
- **Project Standard 2016:** permette di mantenere i progetti organizzati.

Nel caso vengano acquistate meno di cinque copie di Project Professional 2016 o Project Standard 2016 queste possono essere acquistate da rivenditori.

Descrizione del Musician

Il Musician di Project legge le chiavi di registro per recuperare le informazioni relative alla versione di Microsoft Project correntemente installata sul computer.

Output del programma

Il Musician di Project restituisce in *output* due file:

- *Project-Report.csv*: questo file contiene tutte le informazioni recuperate dal Musician sulla versione di Project correntemente installata sul computer:
 - Display Name: Il nome dell'installazione di Microsoft Project;
 - Version: La versione dell'installazione di Microsoft Project.
- *Office.log*: Il log del Musician di Microsoft Project memorizza informazioni sull'esecuzione del programma, incluso il verificarsi di eventuali errori.

Permessi Utente e diritti di accesso richiesti

Per riuscire a recuperare tutte le informazioni richieste l'utente che esegue il Musician di Project deve avere diritti da amministratore sulla macchina dove viene lanciato il programma.

4.8 Microsoft Visio

Descrizione del prodotto

Microsoft Visio, conosciuto precedentemente come Microsoft Office Visio, è un applicazione che permette di realizzare diagrammi e grafiche vettoriali facente parte della famiglia Microsoft Office. Come Microsoft Project tuttavia non fa parte delle normali distribuzioni del pacchetto Office. [29]

Licensing

Microsoft Visio è distribuito sotto diverse licenze.

- **Microsoft Visio Standard 2016:** è pensato per gli utenti singoli che sono alla ricerca di un piattaforma potente e completa per realizzare diagrammi che renda disponibile un set di matrici predefinite.
- **Microsoft Visio Professional 2016:** permette a utenti singoli e gruppi di creare e condividere con grande facilità programmi professionali che semplificano la rappresentazione di informazioni complesse. Include tutte le funzionalità della versione Standard, più forme aggiornate, template e stili, supporto avanzato per la collaborazione in team, inclusa la possibilità di lavorare insieme contemporaneamente ad un diagramma, oltre all'abilità di collegare direttamente i diagrammi ad insiemi di dati.
- **Microsoft Visio Pro for Office 365:** questa versione viene resa disponibile tramite la sottoscrizione a Microsoft Office 365 e include tutte le funzionalità di Visio Professional, ma permette a ciascun utente di installare fino a cinque copie di Visio in cinque differenti computer con sistema operativo Windows 10, Windows 8 o Windows 7. Installa automaticamente gli ultimi aggiornamenti fino a quando vale l'abbonamento.

Se vengono comprate meno di cinque copie di Microsoft Visio Pro per Office 365 queste possono essere acquistate da rivenditori.

Una licenza di Visio viene distribuita come parte di SharePoint quando viene acquistata una licenza SharePoint Enterprise CAL, Office 365 Enterprise E3 e con il secondo piano di SharePoint Online. [30]

Descrizione del Musician

Il Musician di Visio legge le chiavi di registro per recuperare le informazioni relative alla versione di Microsoft Visio correntemente installata sul computer.

Output del programma

Il Musician di Project restituisce in *output* due file:

- *Visio-Report.csv*: questo file contiene tutte le informazioni recuperate dal Musician sulla versione di Visio correntemente installata sul computer:
 - Display Name: Il nome dell'installazione di Microsoft Visio;
 - Version: La versione dell'installazione di Microsoft Visio.
- *Visio.log*: Il log del Musician di Microsoft Visio memorizza informazioni sull'esecuzione del programma, incluso il verificarsi di eventuali errori.

Permessi Utente e diritti di accesso richiesti

Per riuscire a recuperare le informazioni richieste l'utente che esegue il Musician di Visio deve avere diritti da amministratore sulla macchina dove viene lanciato il programma.

Capitolo 5

Opera House

In questo capitolo verrà descritto in maggiore dettaglio il prototipo di interfaccia web sviluppato durante lo stage. Verrà fornita una descrizione generale dell'Opera House seguita da delle brevi sezioni sulle singole pagine realizzate.

Introduzione

Il prototipo realizzato di interfaccia web è stato un sito statico in versione desktop. Per il suo sviluppo mi è stato fornito un pdf contenente un mockup delle diverse pagine da realizzare, che conteneva per ciascuna pagina il *layout* e i diversi elementi che la dovevano comporre. Il mockup è stato quindi realizzato come prototipo con codice HTML e CSS

Una volta terminata la costruzione del sito statico un collega ha modificato le pagine HTML inserendoci del codice php, rendendolo più interattivo ed estraendo i dati da inserire nel sito da un *database*.

Passeremo ora ad una descrizione più dettagliata dell'Opera House seguendo il medesimo percorso che seguirebbe un utente qualsiasi, presentando man mano le diverse pagine che andrebbe ad incontrare.

Descrizione

La prima pagina che un Fornitore si trova ad affrontare una volta aperto il sito è quella di Login. Questa mostra il logo ufficiale di Beryllium e due campi di *input* dove inserire il nome utente e la *password*.

Una volta effettuato il login l'utente viene portato ad una homepage, Figura 5.1, che mostra una *overview* di informazioni sui suoi clienti. La parte sinistra dello schermo è occupata da una colonna presente in tutte le pagine del sito. In questa colonna possiamo trovare un'immagine dell'utente correntemente loggato con a fianco il suo nome utente e il link di logout che fa uscire l'utente dal sito riportandolo alla pagina di login. Sotto a questo troviamo una barra di ricerca, che permette al Fornitore di effettuare una ricerca tra i suoi clienti. Sotto ancora ci sono tre pulsanti:

- *Add a new customer*: sposta l'utente su una pagina dove ha la possibilità di inserire un nuovo cliente;

- *Customer list*: mostra una lista completa dei clienti del Fornitore;
- *System preferences*: sposta l'utente su una pagina dove ha la possibilità di impostare le sue preferenze.

Sotto ancora ci sono una lista di informazioni generali di interesse al Cliente seguite dal logo di Beryllium. Sotto quelle è presente il logo dell'azienda e la scritta *beta*, che sta ad indicare che la versione attuale è un prototipo.

Sul lato destro dello schermo invece troviamo un titolo che indica l'azienda di cui fa parte il fornitore e una serie di diagrammi contenenti un'overview dei trend di diverse informazioni di interesse del Fornitore.

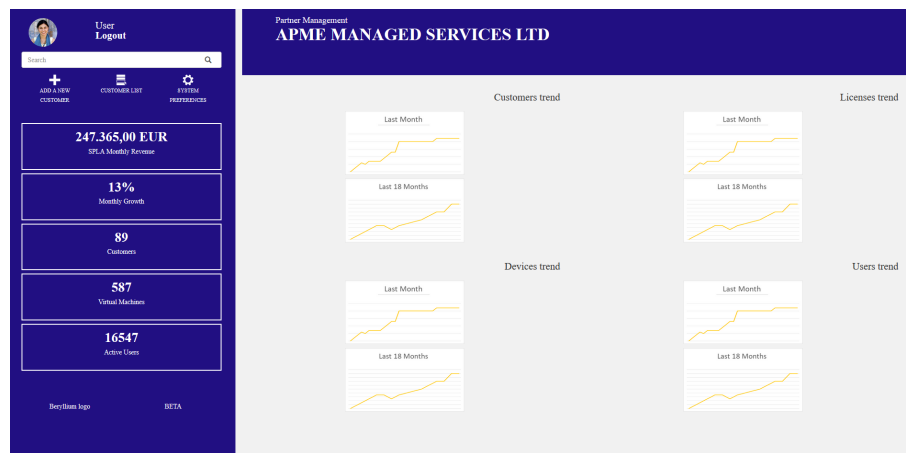


Figura 5.1: Opera House - Homepage.

A questo punto per continuare a navigare è necessario che il Fornitore selezioni uno dei suoi clienti dall'elenco o effettuando una ricerca. Una volta selezionato l'utente viene spostato in una seconda parte del sito specifica per ciascun Cliente composta da diverse pagine. In ciascuna di queste pagine, oltre alla colonna precedentemente descritta, ve ne è una seconda che contiene quattro icone rappresentanti ciascuna delle altre pagine, permettendo quindi di navigare comodamente tra le diverse pagine del sito, e una quinta icona che permette al Fornitore di scaricare un Conductor da installare sui *server* del Cliente. Il titolo in alto a destra inoltre ora si riferisce al Cliente selezionato e non più all'azienda del Fornitore, e sotto di esso sono presenti una serie di informazioni generali sul Cliente.

Segue ora una breve descrizione delle singole pagine:

- *Pagina di overview*: Questa è la pagina su cui viene portato il Fornitore nel momento in cui seleziona un Cliente. Su questa pagina si possono trovare una serie di informazioni generali sul Cliente selezionato. Una volta che si è spostati da questa pagina per tornarvi bisogna rileszionare il Cliente. Uno *screenshot* della pagina è visibile in Figura 5.2.

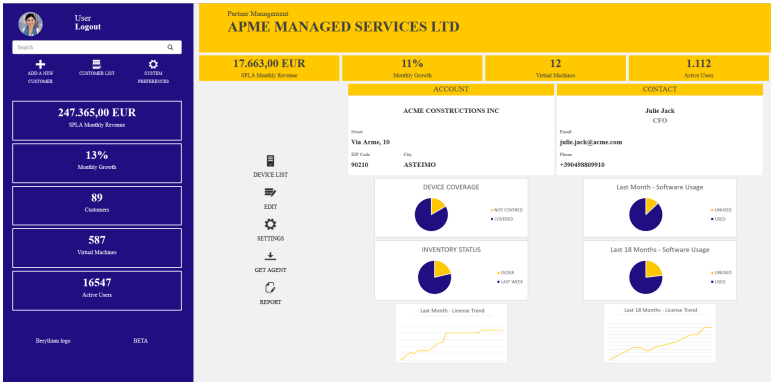


Figura 5.2: Opera House - Overview Page.

- *Pagina dell’inventario:* Su questa pagina è possibile vedere una vista dell’inventario del Cliente selezionato. Uno *screenshot* della pagina è visibile in Figura 5.3.

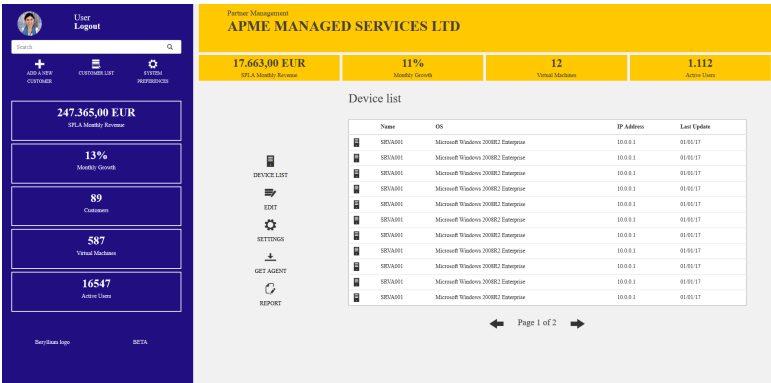


Figura 5.3: Opera House - Inventory Page.

- *Pagina di report SKU:* Su questa pagina è possibile vedere una vista degli SKU del Cliente selezionato. Uno *screenshot* della pagina è visibile in Figura 5.4.

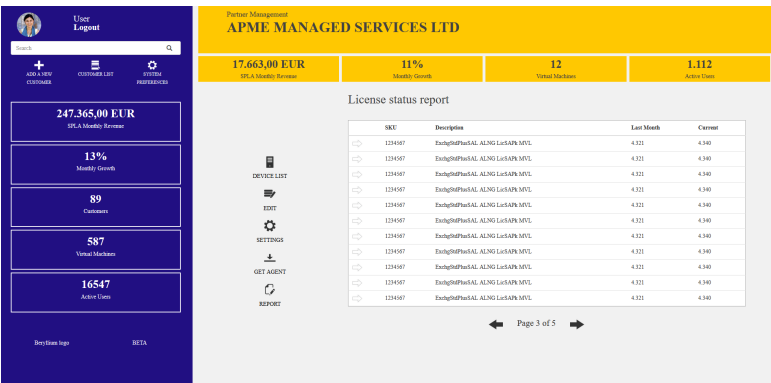


Figura 5.4: Opera House - Report Page.

- *Pagina di edit del Cliente:* Su questa pagina è possibile modificare le informazioni riguardanti al Cliente e viene anche offerta la possibilità di eliminare il Cliente attualmente selezionato. Uno *screenshot* della pagina è visibile in Figura 5.5.

The screenshot shows the 'Edit Customer' page in the Opera House system. The sidebar on the left contains a search bar, a 'User Logout' button, and a list of navigation icons: 'ADD A NEW CUSTOMER', 'CUSTOMER LIST', 'SYSTEM MANAGEMENT', '247,365,00 EUR (DFA Monthly Revenue)', '13% Monthly Growth', '89 Customers', '587 Virtual Machines', and '16547 Active Users'. The main content area has a yellow header with 'Partner Management' and 'APME MANAGED SERVICES LTD'. Below the header is a summary table with four columns: '17,663,00 EUR (DFA Monthly Revenue)', '11% Monthly Growth', '12 Virtual Machines', and '1,112 Active Users'. The 'Edit Customer' form includes sections for 'Account' (Account name, Street, Zip Code, City), 'Contact' (Contact, Role, Email, Phone), and 'Agreement' (Sales Representative, Agreement Start Date, Agreement End Date). At the bottom are 'Save', 'Cancel', and 'Delete Customer' buttons.

Figura 5.5: Opera House - Edit Page.

- *Settings:* Questa pagina permette di modificare le impostazioni riguardanti delle funzionalità di *warning* offerte dall'Opera House. Per esempio qui è possibile indicare quanti giorni prima della scadenza del contratto inviare una mail automatica al Cliente e al Fornitore per avvisarli. Uno *screenshot* della pagina è visibile in Figura 5.6.

The screenshot shows the 'Customer Settings' page in the Opera House system. The sidebar is identical to the previous screenshot. The main content area has a yellow header with 'Partner Management' and 'APME MANAGED SERVICES LTD'. Below the header is a summary table with four columns: '17,663,00 EUR (DFA Monthly Revenue)', '11% Monthly Growth', '12 Virtual Machines', and '1,112 Active Users'. The 'Customer Settings' form includes a 'Warnings' section with four fields: 'Overusage threshold', 'Underusage threshold', 'Agreement warning', and 'Agreement critical'. At the bottom are 'Save', 'Cancel', and 'Delete Customer' buttons.

Figura 5.6: Opera House - Settings Page.

Capitolo 6

Valutazione retrospettiva

Questo capitolo mostra il bilancio complessivo dell'esperienza di stage, valutando il raggiungimento degli obiettivi posti dal Piano di Lavoro e presentando le conoscenze acquisite durante questo periodo.

6.1 Raggiungimento degli obiettivi

Gli obiettivi determinati durante la stesura del Piano di Lavoro, suddivisi per tipologia, possono essere riassunti come segue:

- **Minimi**

- *min01*: Comprensione del mondo SAM, SPLA e dei prodotti Microsoft di interesse;
- *min02*: Progettazione, sviluppo, test e documentazione di almeno due Musician.

- **Massimi**

- *max01*: Progettazione, sviluppo, test e documentazione di sei Musician;
- *max02*: Miglioramento ed estensione delle funzionalità dei Musician per Exchange e Lync già esistenti;
- *max03*: Sviluppo di un prototipo dell'Opera House

- **Formativi**

- *for01*: capacità di creare un ambiente microsoft con Exchange, SharePoint, Active Directory, SQL, DNS, DHCP e RDS;
- *for02*: capacità di utilizzare sistemi di versionamento in contesti aziendali;
- *for03*: capacità di progettare componenti modulari che dovranno interagire con sistemi di produzione;
- *for04*: capacità di sviluppare parti di progetti di grandi dimensioni;
- *for05*: capacità di utilizzare sistemi infrastrutturali VMware.

L'attività di stage è iniziata con un periodo di formazione come previsto. Durante

questa prima parte sono stato introdotto al mondo Microsoft e al SAM. In questo primo periodo ho anche potuto iniziare a creare un ambiente di lavoro. Per fare questo ho dovuto imparare a sfruttare i sistemi di infrastruttura virtuale aziendali e impostare una macchina server Windows dove poter installare i vari programmi che sarei poi dovuto andare ad analizzare. Al termine di questo primo periodo stati soddisfatti i requisiti *min01*, *for05* e una prima parte di *for01*.

Dopo questa prima parte di formazione sono passato a lavorare al progetto vero e proprio. Ho iniziato svolgendo un'analisi dei requisiti generale che mi ha permesso di comprendere meglio le funzionalità dell'intero prodotto e sono poi passato ad iniziare lo sviluppo dei Musician. Con lo sviluppo dei Musician ho iniziato a soddisfare gli altri requisiti. In particolare ho dovuto far uso del sistema di versioning aziendale, soddisfacendo l'obiettivo *for02*, e ho dovuto integrare quello che stavo facendo con il resto dell'applicativo, completando *for03* e *for04*.

Una volta completato lo sviluppo di due Musician, e avendo quindi soddisfatto *min02*, ho dovuto interrompere l'attività di analisi del terzo per passare allo sviluppo dell'Opera House. Per necessità aziendali infatti era richiesto avere un prototipo dell'interfaccia web in tempi brevi al fine di poterla mostrare durante una presentazione generale del progetto alla Microsoft. Ho quindi sviluppato il prototipo a partire da un mockup fornitomi soddisfacendo il requisito *max03*.

Una volta terminato lo sviluppo del prototipo sono tornato allo sviluppo dei Musician. Ho quindi ripreso il lavoro su quello che avevo interrotto in fase di analisi e, dopo averlo completato ne ho sviluppati altri tre, completando dunque l'obiettivo *max01*.

Dopo aver sviluppato i sei Musician mi sono dedicato all'estensione delle funzionalità dei Musician già esistenti di Exchange e Lync. Mentre per il primo ho potuto ampliarne le funzionalità senza intoppi per il secondo ciò non è stato possibile. Per poterne continuare lo sviluppo era infatti necessario procurarsi il *software* server di Lync, piuttosto costoso, che l'azienda non aveva disponibile. Vista l'impossibilità di lavorare al Musician di Lync ho quindi sviluppato un Musician extra rispetto ai requisiti facilitato dal fatto che si basava sullo stesso funzionamento di due che avevo già programmato.

Durante i periodi di analisi e sviluppo ho potuto continuare ad approfondire la conoscenza dei diversi programmi che andavo ad utilizzare, ho installato un nuovo server e ho potuto completare l'ultimo requisito formativo restante, il *for01*.

Al termine dello stage ho dunque completato tutti gli obiettivi minimi, tutti quelli formativi e due di quelli massimi. L'unico rimanente è stato il *max02*, il cui completamento è stato impossibilitato dalla non disponibilità del server Lync. Al suo posto tuttavia è stato sviluppato un Musician ulteriore rispetto ai requisiti massimi posti.

6.2 Conoscenze acquisite

Lo sviluppo del progetto di stage ha richiesto lo studio e l'apprendimento di diverse tecnologie precedentemente sconosciute. L'ampia gamma di prodotti Microsoft che dovevano essere analizzati dai Musician mi ha permesso di ottenere una visione molto estesa, seppure abbastanza superficiale, del mondo dei *software* e del licensing Microsoft. Ho inoltre appreso molto sulla gestione di un ambiente Windows con Active Directory, in particolare dal lato server. Ho anche avuto la possibilità di imparare un nuovo linguaggio di programmazione, e anche di apprezzare le potenzialità e l'utilità di PowerShell, che si è rivelato uno strumento sorprendentemente utile e quasi indispensabile per il lavoro svolto. Le principali conoscenze acquisite vengono qui brevemente descritte:

- **SAM e licensing**: durante l'intera durata del progetto ho dovuto costantemente interagire con il SAM e le diverse regole di licensing dei prodotti Microsoft. L'azienda mi ha anche offerto l'accesso ai fini di studio all'ISO/IEC 19770-1:2012. Questo mi ha permesso di ottenere una visione generale di questi concetti e soprattutto di comprendere quanto sia ampio e profondo questo argomento.
- **C#**: lo sviluppo dei Musician ha richiesto l'utilizzo di C# come linguaggio principale. Per questo ho potuto impararne diversi aspetti. Oltre alle funzionalità di base ho anche imparato come interfacciarmi con la WMI (Windows Management Instrumentation) e con Application Program Interface (API)^[g] esterne, in particolare ho dovuto apprendere e sfruttare il concetto di Marshalling per gestire codice *unmanaged* all'interno di un ambiente .NET.
- **PowerShell**: sebbene lo sviluppo finale dei Musician dovesse essere svolto in C#, PowerShell si è rivelato uno strumento estremamente utile soprattutto nelle fasi preliminari dello sviluppo di ciascun Musician. Esso infatti permette un accesso estremamente semplice ad una grandissima quantità di funzionalità per interfacciarsi con i prodotti Microsoft, e mi ha permesso di testare molte soluzioni prima di andare a scrivere del codice più complicato. È stato inoltre molto utile per effettuare il test dei diversi Musician, in quanto ha fornito uno strumento alternativo con cui confrontare i risultati.
- **Visual Studio**: Visual Studio è stato l'ambiente di sviluppo utilizzato per il progetto. Le diverse funzionalità offerte dall'IDE si sono rivelate particolarmente utili lavorando in un ambiente Microsoft, ed il comodo sistema di versionamento ha reso semplice lavorare con i colleghi.
- **Windows Server 2008 R2/2016**: l'apprendimento delle funzionalità di base della gestione dei server Microsoft è stato essenziale. Per poter sviluppare i Musician infatti ho spesso dovuto installare e impostare i prodotti Microsoft che andavano ad analizzare, molti dei quali erano rivolti ad un ambiente server. Durante lo sviluppo ho potuto quindi installare due versioni diverse dei server Microsoft, che ho poi dovuto imparare a gestire autonomamente.
- **Active Directory**: imparare il funzionamento di Active Directory si è rivelato fondamentale per l'utilizzo delle diverse tecnologie. Tutte quante infatti si integravano con il dominio aziendale, e ho quindi dovuto apprendere come gestirlo.
- **SharePoint**: l'apprendimento di SharePoint è stato necessario per lo sviluppo del corrispondente Musician. Ho infatti dovuto autonomamente installare SharePoint sui miei server, un'operazione che si è rivelata più ardua del previsto, e poi creare i diversi siti da analizzare. Per ottenere l'accesso alle informazioni che dovevo recuperare e determinare i requisiti di accesso necessari ad ottenerle ho anche potuto approfondire particolarmente l'aspetto dei permessi di SharePoint, che è sorprendentemente stratificato.
- **DNS**: per sviluppare il corrispondente Musician ho dovuto installare e configurare su uno dei miei server il ruolo di DNS. Per poter controllare il funzionamento del mio programma ho anche dovuto provvedere a popolare le diverse zone del DNS con vari tipi di record, il che ha richiesto uno studio abbastanza approfondito del funzionamento di questa tecnologia.
- **DHCP**: come per il DNS ho dovuto installare anche il ruolo di DHCP sul medesimo server. Dopo averlo impostato ho avuto delle difficoltà ad effettuare il

lease di un indirizzo IP visto il protocollo con cui vengono assegnati e il fatto che il server era inserito all'interno della rete aziendale, che disponeva già di un DHCP. Fortunatamente questo problema è stato risolto con l'aiuto di un collega.

- **Exchange:** l'apprendimento di Exchange non è stato profondo come quello per le tecnologie finora elencate, tuttavia è stato sicuramente significativo. Ho dovuto infatti imparare ad interfacciarmi con un servizio Exchange già installato per poterne modificare le impostazioni e recuperare le informazioni che mi servivano.
- **Bootstrap:** per lo sviluppo dell'Opera House ho dovuto imparare alcune funzionalità basilari di Bootstrap, che hanno notevolmente semplificato la realizzazione del sito grazie ai diversi template.

6.3 Conclusioni

L'attività di stage mi ha dato la possibilità di entrare in contatto per la prima volta con il mondo del lavoro. È stata un'esperienza estremamente istruttiva che mi ha permesso sia di mettere in pratica una buona parte della teoria precedentemente appresa all'università, sia di imparare una sorprendente quantità di nuove tecnologie e strumenti di lavoro.

L'ambiente in cui sono stato inserito è stato molto positivo, i miei colleghi si sono sempre dimostrati disponibili e la strumentazione datami per lavorare si è sempre rivelata adatta allo svolgimento dei miei compiti.

Durante lo stage ho potuto imparare a lavorare sia come parte di un team, sia in autonomia. In particolare ho dovuto imparare ad adattarmi alle tempistiche aziendali, inclusa l'aggiunta di una scadenza non preventivata sul lavoro che stavo svolgendo, che ha portato ad una modifica delle tempistiche di sviluppo.

Lavorando otto ore al giorno ho inoltre potuto avere un assaggio della vita lavorativa, di quelle che sono le tempistiche del lavoro a tempo pieno e di come avvicinarmi ad un ambiente di lavoro.

Glossario

Active Directory è un insieme di servizi di rete, meglio noti come *directory service*, adottati dai sistemi operativi Microsoft e gestiti da un *domain controller*. Esso si fonda sui concetti di dominio e di *directory*, ovvero la modalità con cui vengono assegnate agli utenti tutte le risorse della rete attraverso concetti quali ad esempio account utente, account computer, cartelle condivise e stampanti, secondo l'assegnazione da parte dell'amministratore di sistema di *Group Policy* ovvero criteri di gruppo [1]. 6, 11, 15, 16, 23, 25, 29, 32, 43–45, 47

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 ad alto livello semplificando così il lavoro di programmazione [2]. 51

Asset si riferisce ad una qualsiasi risorsa tangibile o intangibile che possa essere posseduta o controllata per produrre valore [3]. 4, 47

CLR è il nome scelto da Microsoft per la macchina virtuale e le librerie standard della piattaforma .NET. Il CLR è l'ambiente di esecuzione del Common Intermediate Language, il linguaggio intermedio in cui i *compilatori* della piattaforma .NET traducono i linguaggi ad alto livello supportati dalla piattaforma [5]. 51

Configuration Item si riferisce ad una unità di configurazione (elemento dell'infrastruttura IT) che può essere gestita individualmente (tipo: computer, router, server, software) [6]. 1, 47

CSS è un linguaggio utilizzato per definire la formattazione di documento HTML, XHTML, XML, ad esempio le pagine web dei siti [4]. 51

DHCP è un protocollo di rete di livello applicativo che permette ai dispositivi o terminali di una certa rete locale di ricevere automaticamente ad ogni richiesta di accesso ad una rete IP la configurazione IP necessaria per stabilire una connessione e operare su una rete più ampia basata su Internet Protocol [8]. 51

DNS è un sistema utilizzato per la risoluzione dei nodi della rete in indirizzi IP e viceversa. Il servizio è realizzato tramite un database distribuito costituito dai server DNS. Il DNS è diviso in domini (com, org, it) a ciascuno dei quali corrisponde un *nameserver* che conserva un *database* con le informazioni dei

domini di cui è responsabile. Questo dataserve si rivolge ai nodi successivi quando deve trovare informazioni che appartengono ad altri domini [7]. 51

Exchange è un software di *networking* che permette la collaborazione online tra utenti di un'organizzazione. Le funzionalità principali che offre sono la gestione dell'email centralizzata, l'amministrazione di calendari e agende che possono essere condivise tra utenti della rete e la pianificazione condivisa di attività di *business* [21]. 11, 15, 16, 25–27, 34, 43, 44, 46, 48

Framework Architettura logica di supporto (spesso un'implementazione logica di un particolare *design pattern*) su cui un *software* può essere progettato e realizzato, spesso facilitandone lo sviluppo da parte del team di sviluppatori [12]. 12, 13, 48

Help Desk è una risorsa destinata a fornire supporto, all'utente o al cliente, relativamente a prodotti o servizi informatici ed elettronici. Lo scopo dell'Help desk è di risolvere problemi o fornire indicazioni su prodotti come computer, apparecchiature elettroniche o *software*. Le aziende di solito forniscono il supporto Help desk per i loro clienti attraverso vari canali, quali numeri verdi, siti web, e-mail e messaggistica istantanea [13]. 5, 48

HTML è un linguaggio di *markup* nato per la formattazione e impaginazione di documenti ipertestuali disponibili nel web. È utilizzato prevalentemente per il disaccoppiamento della struttura logica di una pagina web e la sua rappresentazione, gestita tramite stili CSS [14]. 51

IT Life Cycle Management è l'insieme di processi, procedure, competenze e strumenti utilizzati nella gestione dei *Configuration Item* che compongono il sistema informativo in tutte le fasi del loro ciclo di vita [16]. 1, 2, 48

IT Operations Management è un processo ITIL che monitora e controlla i servizi e l'infrastruttura IT dell'azienda [17]. 1, 48

ITIL è un insieme di linee guida ispirate alla pratica nella gestione dei servizi IT e consiste in una serie di pubblicazioni che forniscono indicazioni su *best practice* nell'erogazione di servizi IT di qualità e sui processi e mezzi necessari a supportarli da parte di un'azienda [15]. 51

ITSM è una disciplina che si occupa della gestione di sistemi di Information Technology (IT) su larga scala, filosoficamente concentrata sulla prospettiva del cliente e del contributo dell'IT al *business* [18]. 51

Licensing Per licensing si intende l'attività di commercializzazione delle licenze secondo la quale il licenziante (*licensor*) accorda al licenziatario (*licensee*) il diritto di utilizzare un brevetto, un marchio, un certo know-how o un elemento fondamentale per la produzione e/o la commercializzazione di un prodotto o servizio [19]. 1, 21, 25, 28, 29, 32, 33, 35, 36, 45, 48

Lync era il nome originale del programma adesso chiamato Skype for Business, un client di messaggistica istantanea, *Voice Over IP* e videoconferenza. Skype for Business è la versione commerciale di Skype rivolta all'ambiente aziendale [20]. 15, 43, 44, 48

Mockup si riferisce all'attività di riprodurre un oggetto o modello in scala ridotta o maggiorata. In generale un mockup deve dare un'idea di come sarà o dovrà essere l'originale e viene utilizzato per ottenere un *feedback*. 12, 39, 49

RDS è un componente di Microsoft Windows che permette all'utente di prendere controllo di un computer o una macchina virtuale tramite una connessione di rete, permettendogli dunque di utilizzarlo da remoto [31]. 51

SAM si riferisce ad una pratica di *business* che comprende la gestione e l'ottimizzazione dell'acquisto, distribuzione, mantenimento, utilizzo e smaltimento di prodotti *software*. Il Software Asset Management è l'intera infrastruttura e l'insieme di processi necessari per l'efficacia della gestione, controllo e protezione degli asset software attraverso tutti gli stadi del loro ciclo di vita [35]. 51

SharePoint è una piattaforma *software* CMS (Content Management System) che permette di creare e distribuire siti web con scopo di *business* (Intranet) che possono poi successivamente venire resi disponibili anche online come normali siti web. Il focus principale di SharePoint è la condivisione di informazioni e/o documenti. È possibile creare liste, *repository* di documenti, calendari sincronizzati tramite Outlook e molte altre cose. SharePoint è anche completamente integrato con il pacchetto Microsoft Office [27]. 11, 15, 16, 23, 29, 30, 34, 37, 43, 45, 49

SKU è un tipo specifico di prodotto o servizio, inclusi tutti gli attributi a lui associati che lo distinguono da altri tipi. Il termine SKU può anche riferirsi ad un identificatore unico o ad un codice che si riferisce ad un particolare SKU [37]. 51

SOAP è un protocollo per lo scambio di messaggi tra componenti *software* in rete che si basa sul paradigma della programmazione orientata agli oggetti. Utilizza una sintassi XML per la struttura dei messaggi e si poggia sul protocollo HTTP o SMTP per la trasmissione dei messaggi [34]. 51

SPLA si riferisce a una tipologia di licenze mirate alle organizzazioni che desiderano offrire servizi software ai loro clienti. La differenza tra una licenza SPLA e una normale è che mentre la seconda può essere utilizzata solamente da chi la ha acquistata la prima permette al compratore di renderla disponibile a terzi. Questo permette ad alcune organizzazioni di offrire servizi basati su software Microsoft gestendo autonomamente le licenze, in maniera trasparente agli utilizzatori finali [33]. 51

SQL è un linguaggio standardizzato per *database* basati sul modello relazionale [38]. 51

Stage-IT è un'iniziativa che mira a mettere in contatto imprese e studenti. Offre un punto di contatto comune dove gli studenti possono scegliere tra diverse offerte di stage delle varie aziende e le aziende possono entrare in contatto con gli studenti in maniera diretta [36]. 9, 49

Ticket indica una richiesta di assistenza, tracciata e presa in carico da un servizio di assistenza tecnica. Il ticket permette di tracciare tutto il processo di gestione dell'incidente fino alla sua risoluzione. 4, 5, 49

Web Service in informatica un Web Service, o servizio web, è un sistema software progettato per supportare l'interoperabilità tra diversi elaborati su di una medesima rete in un contesto distribuito. Questo permette di offrire diversi tipi di servizi tramite il web [39]. 10, 50

XHTML è un linguaggio di marcatura che associa alcune proprietà dell'XML con le caratteristiche dell'HTML. Una file XHTML è una pagina HTML scritta in conformità con lo standard XML [10]. 51

XML è un metalinguaggio per la definizione di linguaggi di *markup*, ovvero un linguaggio marcatore basato su un meccanismo sintattico che consente di definire e controllare il significato degli elementi contenuti in un documento o in un testo [11]. 51

Acronimi e abbreviazioni

API Application Program Interface. 45, 47

CLR Common Language Runtime. 14, 47

CSS Cascading Style Sheets. 14, 39, 47, 48

DHCP Dynamic Host Configuration Protocol. 11, 15, 16, 31, 32, 43, 45–47

DNS Domain Name System. 11, 15, 16, 31–34, 43, 45, 47

HTML HyperText Markup Language. 13, 14, 39, 47, 48, 50

ITIL Information Technology Infrastructure Library. 2, 4, 9, 48

ITSM IT Service Management. 4, 48

RDS Remote Desktop Services. 11, 15, 16, 28, 29, 43, 49

SAM Software Asset Management. 1, 3, 5, 9, 11, 15, 16, 43–45, 49

SKU Stock Keeping Unit. 10, 18, 20, 49

SOAP Simple Object Access Protocol. 10, 49

SPLA Services Provider License Agreement. 11, 15, 16, 43, 49

SQL Structured Query Language. 11, 15, 16, 23, 43, 49

XHTML eXtensible HyperText Markup Language. 14, 47, 50

XML eXtensible Markup Language. 14, 47, 49, 50

Bibliografia

- [1] *Active Directory*. URL: https://it.wikipedia.org/wiki/Active_Directory (cit. a p. 47).
- [2] *API*. URL: https://it.wikipedia.org/wiki/Application_programming_interface (cit. a p. 47).
- [3] *Asset*. URL: <https://en.wikipedia.org/wiki/Asset> (cit. a p. 47).
- [4] *Cascading Style Sheets*. URL: <https://it.wikipedia.org/wiki/CSS> (cit. a p. 47).
- [5] *Common Language Runtime*. URL: https://it.wikipedia.org/wiki/Common_Language_Runtime (cit. a p. 47).
- [6] *Configuration Item*. URL: http://www.knowledgetransfer.net/dictionary/ITIL/en/Configuration_Item.htm (cit. a p. 47).
- [7] *Domain Name System*. URL: https://it.wikipedia.org/wiki/Domain_Name_System (cit. alle pp. 33, 48).
- [8] *Dynamic Host Configuration Protocol*. URL: https://it.wikipedia.org/wiki/Dynamic_Host_Configuration_Protocol (cit. alle pp. 31, 47).
- [9] *Dynamic Host Configuration Protocol licensing*. URL: <https://blogs.technet.microsoft.com/volume-licensing/2014/03/10/licensing-how-to-when-do-i-need-a-client-access-license-cal/> (cit. a p. 32).
- [10] *eXtensible HyperText Markup Language*. URL: <https://it.wikipedia.org/wiki/XHTML> (cit. a p. 50).
- [11] *eXtensible Markup Language*. URL: <https://it.wikipedia.org/wiki/XML> (cit. a p. 50).
- [12] *Framework*. URL: <https://it.wikipedia.org/wiki/Framework> (cit. a p. 48).
- [13] *Help Desk*. URL: https://it.wikipedia.org/wiki/Help_desk (cit. a p. 48).
- [14] *HyperText Markup Language*. URL: <https://it.wikipedia.org/wiki/HTML> (cit. a p. 48).
- [15] *Information Technology Infrastructure Library*. URL: <https://it.wikipedia.org/wiki/ITIL> (cit. a p. 48).
- [16] *IT Life Cycle Management*. URL: <http://www.netcom.it/content/it/societa/> (cit. a p. 48).

- [17] *IT Operations Management*. URL: https://wiki.en.it-processmaps.com/index.php/IT_Operations_Management (cit. a p. 48).
- [18] *IT Service Management*. URL: <http://www.itil-italia.com/itsm.htm> (cit. a p. 48).
- [19] *Licensing*. URL: <http://www.glossariomarketing.it/significato/licensing/> (cit. a p. 48).
- [20] *Lync*. URL: https://en.wikipedia.org/wiki/Skype_for_Business (cit. a p. 48).
- [21] *Microsoft Exchange*. URL: https://it.wikipedia.org/wiki/Microsoft_Exchange (cit. alle pp. 25, 48).
- [22] *Microsoft Exchange licensing*. URL: <https://products.office.com/it-it/exchange/microsoft-exchange-server-licensing-licensing-overview> (cit. a p. 26).
- [23] *Microsoft Office*. URL: https://en.wikipedia.org/wiki/Microsoft_Office (cit. a p. 34).
- [24] *Microsoft Office licensing*. URL: <https://www.microsoft.com/it-it/Licensing/product-licensing/office.aspx> (cit. a p. 35).
- [25] *Microsoft Project*. URL: https://en.wikipedia.org/wiki/Microsoft_Project (cit. a p. 35).
- [26] *Microsoft Project licensing*. URL: <https://products.office.com/en/project/microsoft-project-licensing>.
- [27] *Microsoft SharePoint*. URL: https://it.wikipedia.org/wiki/Microsoft_SharePoint (cit. alle pp. 29, 49).
- [28] *Microsoft SharePoint licensing*. URL: <https://products.office.com/en-us/sharepoint/sharepoint-licensing-overview> (cit. a p. 30).
- [29] *Microsoft Visio*. URL: https://en.wikipedia.org/wiki/Microsoft_Visio (cit. a p. 37).
- [30] *Microsoft Visio licensing*. URL: <https://products.office.com/it-it/visio/microsoft-visio-volume-licensing-visio-for-multiple-users> (cit. a p. 37).
- [31] *Remote Desktop Services*. URL: https://en.wikipedia.org/wiki/Remote_Desktop_Services (cit. alle pp. 28, 49).
- [32] *Remote Desktop Services licensing*. URL: [https://technet.microsoft.com/en-us/library/cc753650\(v=ws.11\).aspx](https://technet.microsoft.com/en-us/library/cc753650(v=ws.11).aspx) (cit. a p. 29).
- [33] *Services Provider License Agreement*. URL: <https://www.microsoft.com/en-us/licensing/licensing-programs/spla-program.aspx> (cit. a p. 49).
- [34] *Simple Object Access Protocol*. URL: <https://en.wikipedia.org/wiki/SOAP> (cit. a p. 49).
- [35] *Software Asset Management*. URL: https://en.wikipedia.org/wiki/Software_asset_management (cit. a p. 49).

- [36] *Stage-IT*. URL: <http://www.progettogiovani.pd.it/stage-it-2017/> (cit. a p. 49).
- [37] *Stock Keeping Unit*. URL: https://en.wikipedia.org/wiki/Stock_keeping_unit (cit. a p. 49).
- [38] *Structured Query Language*. URL: <https://en.wikipedia.org/wiki/SQL> (cit. a p. 49).
- [39] *Web Service*. URL: https://it.wikipedia.org/wiki/Web_service (cit. a p. 50).