

POLITECNICO DI TORINO

**Corso di Laurea Magistrale
in Ingegneria Gestionale**

Tesi di Laurea Magistrale

**Realizzazione di un progetto IT di business intelligence
tramite piattaforma Qlik Sense.**

Caso di studio nel settore finanziario



Relatore

prof. Tania Cerquitelli

Candidato

Maria Lipira

12 Dicembre 2019

Sommario

1. Introduzione	1
1.1. L'obiettivo	1
1.2. Il contesto	2
2. La Business Intelligence e Qlik Sense	4
2.1. La Business Intelligence e l'analisi dei dati	4
2.2. QlikSense come strumento di BI.....	7
2.2.1. Schema di funzionamento del software.....	10
2.2.2. Oggetti chiave dell'interfaccia.....	11
2.2.3. Mashup - introduzione.....	13
3. Il progetto.....	14
3.1. Raccolta dei requisiti	14
3.2. Caratteristiche del caso di studio.....	15
3.3. Creazione della proposta: mockup dell'interfaccia.....	17
3.4. Gestione del progetto	25
3.5. Analisi e sviluppo.....	27
3.5.1. Progettazione del data model.....	28
3.5.2. ETL1 – Estrazione dati	30
3.5.2.1. ETL1 – Bloomberg	32
3.5.2.2. ETL1 – CAPIQ	38
3.5.2.3. ETL1 – FactSet	39
3.5.2.4. ETL1 – Research Tree	39
3.5.2.5. ETL1 – Thomson Reuters.....	41
3.5.2.6. ETL1 – Google Analytics	41
3.5.2.7. ETL1 – Sales Force.....	42
3.5.2.8. ETL1 – xml	42
3.5.2.9. ETL1 - Dotmailer.....	43
3.5.3. ETL2 – Costruzione del data model	45
3.5.4. Presentation – Oggetti QlikSense	54
3.5.5. Presentation – Mashup.....	65
3.6. Cartelle di progetto e schedulazione task di aggiornamento.....	67
3.7. Section Access.....	69
3.8. UAT e passaggio in produzione	70
4. I risultati	72
4.1. Activity Overview	73
4.2. Data Analysis	74
4.3. Trend Analysis	75
4.4. Geo Analysis	76
4.5. Readers List.....	77
5. Conclusioni	78
Ringraziamenti	79
Acronimi e Significati	80
Bibliografia.....	81
Sitografia	82

1. Introduzione

Il presente elaborato è il risultato di un progetto svolto in ambito aziendale in cui è stata creata una applicazione di analisi dei dati per un cliente del settore finance.

Lo scopo di tale tesi è quello di descrivere il processo che ha portato al prodotto finale, con particolare attenzione ai passaggi tecnici più importanti, alle logiche adoperate per l'implementazione e alle funzionalità dell'applicazione finale. Il processo che sarà illustrato non è relativo soltanto alla parte di puro sviluppo dell'applicativo, ma abbraccia anche tutte le tematiche di contorno, necessarie per la corretta creazione dello stesso. È infatti fondamentale considerare il contesto del progetto e quali sono le motivazioni funzionali che hanno spinto a produrre il prodotto e/o parte di esso in un certo modo, rispondendo alle esigenze del cliente, con il quale vi è stato un continuo confronto in tutte le fasi del progetto, atto ad approfondire e migliorare l'output finale, sulla base dei controlli e delle constatazioni effettuati durante tutto il periodo. Il progetto verrà presentato anche sotto l'aspetto prettamente gestionale, mostrando quali sono i tempi medi di analisi, sviluppo e gestione, sulla base delle stime effettuate. Questo approccio permetterà di dare una panoramica a 360 gradi di cosa vuol dire svolgere un progetto di Business Intelligence.

Inoltre verranno date indicazioni circa la tecnologia utilizzata e del perché essa è stata scelta. Verranno spiegati i concetti base di tale tecnologia, come essa funziona e qual è il giusto modus operandi che permette di massimizzare le performance del prodotto finale e la manutenibilità dello stesso nel tempo.

1.1. L'obiettivo

L'obiettivo di questa tesi di laurea è quello di dimostrare come gli strumenti di Business Intelligence, e in particolar modo Qlik Sense, possono essere efficaci per analizzare i dati delle aziende e ricavarne valore aggiunto per queste ultime.

Al giorno d'oggi infatti, risulta indispensabile per le aziende riuscire ad elaborare l'enorme mole di dati a loro disposizione, al fine di prendere decisioni atte a migliorare e aumentare l'efficienza aziendale, creandosi così un vantaggio competitivo anche nel medio – lungo periodo.

Qlik Sense è uno strumento di BI tramite il quale è possibile elaborare i dati provenienti da diverse sorgenti al fine di effettuare analisi su differenti livelli di complessità; inoltre possono essere oggetto di analisi i dati provenienti da qualsiasi settore aziendale, come finance, e-commerce, logistica, sales, ecc.

Utilizzando come caso di studio il progetto su cui ho lavorato negli scorsi mesi, sarà possibile comprendere:

- Che cos'è Qlik Sense e come funziona
- Come è strutturato un progetto di analisi dei dati con Qlik Sense, commissionato dal cliente in outsourcing ad un'azienda di consulenza
- Quali sono i passaggi necessari al fine di creare un output finale che soddisfi i requisiti del cliente

1.2. Il contesto

Il contesto in cui è stato creato il progetto è quello di un'azienda di consulenza informatica (per la quale lavoro) a cui è stato commissionato un progetto in ambito finance per una società di consulenza per gli investimenti.

Tale società cliente (che per motivi di privacy chiameremo col nome di fantasia "Finance") lavora con società quotate in borsa o che desiderano quotarsi, offrendo servizi di consulenza in materia di raccolta fondi, IPO, due diligence, rapporti strategici e studi di fattibilità.

Finance gestisce per i suoi clienti anche la pubblicazione dei report aziendali che permettono di dare informazioni circa la situazione strategica, economica e patrimoniale attuale e futura a dei potenziali investitori, che tramite tali informazioni stabiliscono se investire o meno in tali società. Questi potenziali investitori, visualizzano tali documenti su delle piattaforme in cui, in alcuni casi, per poter accedere è necessario essere iscritti. In questo modo, Finance riesce ad avere dei dati utili su tali potenziali investitori, come il nome della società, la località, la tipologia di investitore, il documento che ha attratto l'interesse, l'azienda di interesse ecc.

Finance con i dati dei potenziali investitori (che chiameremo genericamente "lettori" o "Readers") raccolti dal suo sito e dalle piattaforme in cui pubblica i documenti, vorrebbe poter dare ai suoi clienti un'applicazione che aggrega tali informazioni, fornendo delle analisi utili e significative, che permettano di monitorare l'efficacia dei documenti pubblicati e il presunto interesse dei Readers.

Lo sviluppo di tale applicazione, che è stato commissionato all'azienda per la quale lavoro, ha quindi lo scopo di soddisfare tale requisito primario; non essendo da subito chiaro per Finance il modo in cui tali dati debbano essere visualizzati, ci hanno chiesto di presentare un mockup dell'applicazione, da cui partire per poter scegliere le analisi e la modalità di fruizione dell'App.

Dopo aver definito insieme a Finance quale output fornire e in che modalità, si passa alla stima delle tempistiche di realizzazione del progetto, definite sulla base del tempo necessario e della disponibilità delle risorse. Una volta definiti tali termini e condizioni è stato possibile iniziare con la parte operativa del progetto, che consiste nelle seguenti macro attività:

- Analisi delle strutture dati messe a disposizione,
- Creazione degli estrattori dei dati
- Messa a punto di elaborazioni necessarie atte ad aggregare e collegare i dati provenienti da fonti diverse
- Gestione delle eccezioni nei dati e delle storicizzazioni
- Creazione del data model
- Messa a punto dell'interfaccia utente.

Sulla parte operativa, il mio contributo è stato totale: del progetto finale mi sono occupata, supportata dal team, della realizzazione del mockup, dell'analisi delle richieste e dei dati, dell'estrazione di questi ultimi dalle sorgenti, dell'elaborazione del data model e creazione degli oggetti di analisi da inserire nell'interfaccia web.

Non è stata di mia competenza la parte relativa alla creazione a livello di Front-end dell'interfaccia web su cui inserire le analisi e gli oggetti QlikSense da me creati. Inoltre la parte prettamente relativa alla gestione del progetto è stata gestita dai project managers; su tale parte ho solo fornito dei contributi di analisi del progetto che sono serviti per ottimizzare tali attività di gestione.

2. La Business Intelligence e Qlik Sense

Al fine di comprendere al meglio come è strutturato il progetto in esame, in questo capitolo verranno analizzati i concetti di Business Intelligence e analisi dei dati, con riferimento all'applicabilità di questi ultimi tramite appositi strumenti di BI. Infine verrà analizzata la tecnologia QlikSense utilizzata per la creazione dell'applicativo, spiegandone la struttura, il funzionamento, le tecnologie accessorie che utilizza e le caratteristiche principali.

2.1. La Business Intelligence e l'analisi dei dati

La diffusione e sviluppo delle tecnologie informatiche su scala mondiale hanno notevolmente accorciato le distanze tra i mercati internazionali. In questo ambiente fortemente competitivo, l'azienda deve rispondere rendendo più duttile possibile la sua struttura aziendale, in modo da spingere al massimo la gamma e la varietà di prodotti e servizi forniti. I sistemi aziendali diventano sempre più complessi, risultando così fondamentale l'obiettivo di adattarsi in termini di efficienza e tempestività decisionale.[1]

La Business Intelligence (BI) è nata per rispondere alle esigenze delle aziende, fornendo a queste ultime gli strumenti per poter prendere decisioni atte a migliorare il livello di efficienza organizzativa.

Howard Dresner, analista del gruppo Gartner, nei primi anni '90 definisce la Business Intelligence come:

- L'insieme dei processi aziendali atti ad analizzare dati e a raccogliere informazioni strategiche;
- La tecnologia utilizzata per questi processi;
- Le informazioni ottenute come risultato di questi processi.

Lo scopo della raccolta dei dati da parte delle aziende è quello di trarre informazioni, valutazioni e stime atte a consentire ai manager e responsabili aziendali di prendere decisioni per migliorare le strategie e i processi di business, in modo da trarne vantaggi competitivi al fine del raggiungimento degli obiettivi prefissati; la Business Intelligence è il prodotto che nasce da questa esigenza, costituendo lo strumento principale per orientare le scelte del *management* verso tali obiettivi [10].

La Business Intelligence è indispensabile per trasformare l'enorme e crescente massa dei dati presente all'interno delle aziende in patrimonio informativo facilmente analizzabile, in un quadro che sia il più svincolato possibile da eventi e variabili casuali in modo da rendere il supporto decisionale oggettivo e scientifico tramite un sistema rivolto a stimare, simulare e sviluppare scenari differenti [4].

Negli ultimi anni il divario tra dati immagazzinati ma non analizzati è cresciuto in modo aggressivo, rendendo presenti all'interno delle aziende due grandi problematiche, spesso non facilmente gestibili:

- La storicizzazione dei dati, finalizzata alla creazione di memoria storica aziendale da cui poter attingere

- La necessità di un intermediario che sappia attingere ai dati richiesti dall'utente; ciò provoca spesso un allungamento dei tempi di risposta e può dare luogo ad inefficienti estrazioni/interpretazioni dei dati.

La BI risolve entrambi questi problemi fornendo direttamente all'utente finale gli strumenti necessari per ottenere risposte in modo elastico e tempestivo [5].

I principali modelli che caratterizzano la BI sono [3]:

- *Data Modeling* (modellazione dati): incentrato sulle dipendenze logiche tra i dati, in cui i dettagli dei valori delle singole osservazioni sono ignorati a favore delle strutture dati d'interesse che vengono riunite al fine di eliminare le ridondanze e collegare le strutture dati con delle relazioni.
- *MIS -Management Information Systems* (sistemi di gestione dell'informazione): costituiti dalla rete di tutti i canali di comunicazione usati all'interno di un'organizzazione. Possono essere definiti come sistemi che raccolgono e processano informazioni fornendole ai manager di tutti i livelli, i quali le usano per prendere decisioni, pianificare, implementare programmi e controlli.
- *DSS - Decision Support Systems* (sistemi di supporto alle decisioni): sistemi informativi computerizzati che supportano i processi decisionali.
Sono sistemi informatici rivolti a facilitare e supportare le esigenze decisionali di alto livello, fornendo facile accesso alle informazioni interne ed esterne importanti per raggiungere gli obiettivi strategici dell'azienda. La peculiarità sta nelle visualizzazioni grafiche e nelle facili interfacce utente, che offrono sia i dati aggregati che al massimo livello di dettaglio.

Sono state negli anni sviluppate numerose tecnologie per l'analisi e l'accesso ai dati, che comprendono strumenti di query, reportistica, analisi statistiche e previsioni.

Uno degli obiettivi chiave di un sistema di Business Intelligence è il coerente utilizzo delle informazioni all'interno dell'organizzazione, in quanto esso può fungere da unico punto di accesso per diversi dati aziendali.

Lo schema di processo delle tecnologie di BI segue il seguente standard:

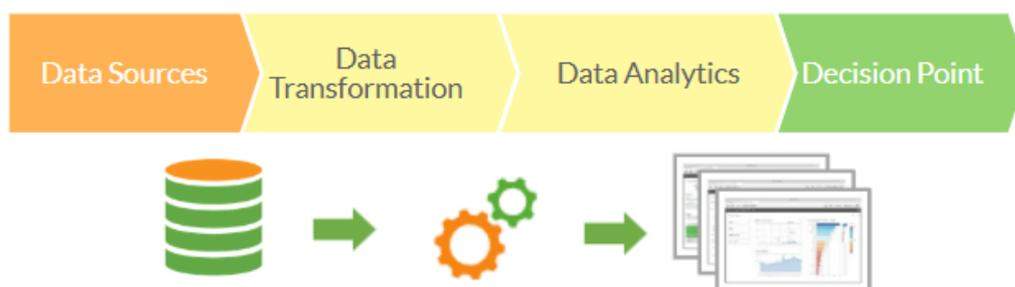


Figura 1. Schema del processo utilizzato dalle tecnologie BI

Per supportare questo processo, l'adozione della BI può anche comportare lo sviluppo di servizi di infrastruttura nell'organizzazione, come governance dei dati e gestione dei metadati.

Le iniziative di Business Intelligence si concentrano sulle informazioni necessarie per supportare il processo decisionale su diversi livelli all'interno dell'organizzazione [2]:

- Livello esecutivo: supporta decisioni strategiche
- Livello di gestione: supporta decisioni tattiche
- Livello di processo: supporta le decisioni operative.

La BI segue l'evoluzione delle dinamiche competitive attuali; le soluzioni fornite da essa sono per lo più date da progetti eseguiti da società di consulenza specializzate. Infatti, le aziende al giorno d'oggi hanno preso coscienza del fatto che servirsi dell'*outsourcing* per alcune funzioni aziendali è la strada migliore per cercare di garantire una forma di vantaggio economico e competitivo. Queste considerazioni valgono ovviamente anche per il mondo della BI.

Attualmente sono poche le aziende che possono sostenere internamente i costi legati alla realizzazione di un progetto di BI. Infatti una scelta di "*insourcing*" di questo tipo richiederebbe investimenti elevati in termini di formazione e costi fissi iniziali con la consapevolezza di ottenere risultati solo in un'ottica di medio – lungo termine. D'altra parte, il ricorso ad una consulenza BI è un passo delicato per la necessità di individuare la società di consulenza che meglio si adatti all'esigenze dell'azienda, in quanto la scelta di un partner che si riveli errato (per incapacità tecniche ecc.) comporta dei costi di uscita elevati.

Uno dei più grossi vantaggi della BI in *outsourcing* consiste nel fatto che il cliente usufruisce di prestazioni migliori rispetto a quelle che riuscirebbe ad ottenere internamente, poiché l'azienda consulente è specialista nel settore e può sviluppare in esso economie di scala, di scopo e di esperienza. Inoltre quando verranno superate le difficoltà di coordinamento iniziali tra i partner, il fornitore sarà in grado di adeguare sempre più i propri servizi alle richieste del cliente [6].

In tale contesto, un ruolo chiave è dato dalla figura del Business Analyst, che funge da collegamento principale tra il cliente e il fornitore nella realizzazione di un progetto di BI. Egli spesso fa parte del team del fornitore stesso, partecipando dunque anche alle attività di implementazione del progetto tra cui [2]:

- Modellizzazione dei dati
- Progettazione di presentations (ad esempio, dashboards)
- Creazione di query ad per l'estrazione dei dati.

Le attività chiave che il Business Analyst deve svolgere relativamente al progetto sono [2]:

- Analisi requisiti e valutazione di potenziali soluzioni
- Comprensione dei dati, estrazione e tecniche analitiche predittive, sviluppo di visualizzazioni
- Analisi dei dati dei sistemi di origine da utilizzare per gli scopi analitici richiesti
- Modellazione e definizione dei dati di origine e di destinazione
- Creazione di modelli di struttura di dati logici.

Il seguente schema mostra quali sono gli steps standard di un progetto di Business Intelligence:



Figura 2. Steps di sviluppo di una soluzione di BI

Partendo dunque da una o più necessità di business (BUSINESS PROBLEM), viene fatta la raccolta e l'analisi dei requisiti, al fine di verificare quali sono le funzionalità che il sistema deve avere per soddisfarli. Dopo viene effettuata la progettazione della soluzione, al fine di schematizzare le attività di sviluppo necessarie, le sorgenti e gli output previsti. In questa fase può essere utile creare un mockup dell'interfaccia da far preventivamente vedere al cliente, in modo da smarcare subito eventuali modifiche sulle funzionalità da creare. Successivamente si procede con la fase di sviluppo dell'applicazione. Al termine dagli sviluppi sarà necessario testare l'applicativo, sia da parte del fornitore ma soprattutto da parte del cliente che userà l'applicazione, in modo da far emergere eventuali bug o anomalie nelle funzionalità.

Una volta che l'applicativo è stato testato, ed eventuali modifiche sono state apportate, segue la fase di rilascio dell'applicativo a tutti gli user autorizzati all'accesso. In questo modo gli users avranno lo strumento che soddisfa le loro necessità (BUSINESS SOLUTION).

Una limitazione aziendale comune con l'introduzione di piattaforme di Business Intelligence è il sottoutilizzo della risorsa informativa stessa in quanto gli users che non hanno familiarità con lo strumento potrebbero concentrarsi semplicemente sulle funzionalità principali di visualizzazione della soluzione, senza esplorare le altre tipologie di servizi spesso resi disponibili da tali piattaforme, come la self-service BI, che possono essere utilizzate dagli stessi in autonomia.

2.2. QlikSense come strumento di BI

QlikSense è un software di Business Intelligence della società QlikTech International, con l'obiettivo di aiutare le aziende ad ottenere valore dai dati, offrendo uno strumento che permetta di lavorare in modo intelligente tramite soluzioni end-to-end all'avanguardia che si adattino alle specifiche esigenze.

La QlikTech è un'azienda leader mondiale nelle tecniche di analisi *in-memory* che propone una nuova classe di soluzioni analitiche facili da utilizzare, veloci e flessibili, che supportano gli individui nel miglioramento delle prestazioni delle organizzazioni nelle quali operano e nel governo dell'innovazione e del cambiamento. I software Qlik rendono i dati associativi, creando relazioni visuali di facile comprensione anche quando si gestiscono set di dati ampi e complessi.

La peculiarità di Qlik sta nell'innovativa tecnologia di analisi *in-memory* attraverso la quale tutti i dati vengono caricati dinamicamente nella memoria e le metriche vengono definite man mano che l'utente esegue le selezioni. Le soluzioni analitiche così ottenute sono veloci, flessibili e facili da utilizzare; l'utilizzo di questa nuova tecnologia che elabora dinamicamente i dati in memoria permette dunque di evitare la definizione nei dati, di diverse gerarchie dimensionali, come accade nei tradizionali cubi OLAP.

La tecnologia associativa gestisce i dati in una maniera simile ai processi umani di elaborazione dei problemi matematici, tramite un criterio per associazione graduale delle informazioni. Infatti, mentre i sistemi tradizionali di ricerca delle informazioni richiedono spesso un approccio dall'alto verso il basso, QlikSense permette di iniziare con qualsiasi parte di dati, indipendentemente dalla sua posizione nella struttura dei dati.

Uno dei punti forti di QlikSense è che esso va oltre l'analisi dei dati a livello tradizionale, in cui tutto ciò che veniva visualizzato dall'utente era frutto elaborazioni sui dati precedentemente stabilite; QlikSense fornisce agli utenti strumenti per fare analisi significative self-service, esplorando i dati senza necessità di alcuna idea preconcepita. Inoltre tale strumento è stato creato in modo che, in base al set di dati che osserva, suggerisce all'utente delle tipologie di analisi, grazie a grafici e oggetti automaticamente impostati dal software [14].

Dati diversi, provenienti da diverse sorgenti, possono essere messi a disposizione in un'unica piattaforma dove è possibile analizzarli, condividerli e utilizzarli per prendere decisioni. E' fondamentale l'utilizzo dei dati trasformati al fine di creare vantaggio competitivo nelle aziende, rendendo le scoperte più importanti disponibili a tutti gli utenti dell'azienda attraverso [15]:

- Democratizzazione dei dati: tutti i dati e in qualsiasi combinazione sono accessibili a tutti gli utenti dell'azienda.
- Indicizzazione associativa e Intelligenza aumentata: Qlik combina la potenza dell'apprendimento automatico con l'intuizione umana, mettendo in evidenza nuove intuizioni tutte da approfondire, accelerando le scoperte e aumentando l'alfabetizzazione dei dati e l'affidabilità delle stesse.
- Analytics integrate: le analisi diventano parte di tutti i processi decisionali, entrando nei flussi dei processi aziendali quotidiani, dai dispositivi edge fino alle app principali.

Questo software ha molte aree di applicazione, e consente l'acquisizione di una visione generale unificata e coerente dei dati esistenti su database differenti. QlikSense consente un rapido *time-to-value*, grazie al fatto che non necessita di una definizione preconstituita di dimensione, in quanto ogni dato è disponibile come dimensione o misura in modo flessibile, facilmente modificabile in base alle specifiche esigenze.

Il modello su cui si basa la struttura dati (chiamata anche nuvola) di QlikSense, è quello relazionale, nel quale i record sono suddivisi in modo da ridurre la ridondanza, mentre i campi chiave sono utilizzati per riunire nuovamente i *record* al momento del loro utilizzo effettivo.

Nel momento in cui avviene il caricamento dei dati da un'origine dati, QlikSense crea un modello dati *in-memory* che consente l'accesso a milioni di dati con velocità superiori e minor spazio, in quanto ottimizza le informazioni nel momento in cui le carica, rimuovendo i valori ridondanti dei diversi campi e correlando automaticamente le tabelle.

Grazie alla particolare struttura del modello dati *in-memory*, QlikSense esegue calcoli a velocità elevata, fornendo a tutti gli utenti funzionalità di analisi multidimensionale. Di conseguenza gli utenti possono ottenere risposte alle *query* ed eseguire calcoli in pochissimo

tempo, nonché elaborare grandi *dataset*, motivo per il quale le aziende possono soddisfare in modo efficiente ed efficace le esigenze di un gran numero di utenti. Uno dei grandi pregi di questo software è la sua flessibilità: oltre a fornire potenti applicazioni analitiche, esso consente di eseguire analisi flessibili ad hoc e di creare semplici report stampabili. Questo rende possibile estendere QlikSense a tutti gli utenti, dagli analisti qualificati che creano report dettagliati ai dirigenti che hanno bisogno di una panoramica generale sulle informazioni aziendali. Visto che i dati non necessitano di una pre-aggregazione, gli utenti possono dinamicamente analizzare l'intero volume di dati, fino al livello della singola transazione.

La visione integrata delle informazioni può avvenire tramite una moltitudine di strumenti come cruscotti, analisi ad hoc e report, analizzando i dati provenienti da una o più sorgenti di dati combinate (ERP, Excel, XML, API, ODBC, ecc.).

L'interfaccia utente è caratterizzata da una rappresentazione visiva intuitiva e all'avanguardia, supportata da strumenti come cruscotti, grafici e tabelle navigabili che permettono agli utenti di poter accedere velocemente alle informazioni di cui necessitano; inoltre è possibile la condivisione dei risultati con altri utenti tramite apposite funzioni di condivisione, report, stampe e la completa integrazione con Office.

La piattaforma Qlik si integra agevolmente con l'infrastruttura esistente di un'organizzazione rendendo quindi estremamente facile e veloce raggiungere l'operatività. Le analisi effettuate sono rese accessibili attraverso tecnologie web interattive, flessibili e veloci all'interfaccia utente di Qlik. La piattaforma BI di Qlik offre un'ampia gamma di opzioni di distribuzione, dai client Windows fino a Java. Dipendenti, fornitori e clienti possono connettersi al server Qlik utilizzando un browser Web standard senza dover rinunciare all'uso dell'interfaccia grafica di analisi di Qlik [7].

Riassumendo, con Qlik è possibile:

- Creare un'applicazione flessibile, personalizzata in base alle esigenze
- Collegare i dati in modo relazionale e avere a disposizione uno schema di essi (nuvola dati)
- Realizzare presentazioni basate sui dati, grafici e tabelle dinamici tramite strumento di storytelling
- Eseguire analisi statistiche, previsionali e di storico
- Creare applicazioni che effettuano analisi dei dati provenienti da una moltitudine di sorgenti

Qlik costituisce dunque una robusta piattaforma per la *business analysis*, fornendo soluzioni su una moltitudine di aree di business come marketing, supplychain, produzione, finanza, medicina, meccanica e risorse umane.

2.2.1. Schema di funzionamento del software

Per la creazione di un'applicazione in QlikSense, il primo passaggio consiste nello stabilire quali dati si desidera caricare; questi, di solito provengono da un file, un foglio di calcolo o un database esistente. È possibile caricare i dati da un unico file o database di origine, oppure caricare e integrare i dati da sorgenti diverse contemporaneamente. QlikSense, può utilizzare dati provenienti da una moltitudine di sorgenti, inclusi database generici, connessioni tramite API, protocolli SFTP/FTP, file web, SAP, Google Analytics ed altri, tramite appositi web connectors.

Dopo aver determinato da dove reperire le informazioni occorre creare uno script di caricamento per copiare i dati dall'origine al database associativo di QlikSense. La creazione di uno script di caricamento risulta semplificata dall'uso di procedure guidate che generano le istruzioni dello script per i tipi di sorgenti supportati.

Durante il processo di caricamento QlikSense esamina ciascuna istruzione dello script e la elabora in ordine sequenziale. All'interno dello script di caricamento può essere inserita l'istruzione "store" che permette di creare una copia dei dati caricati ed eventualmente elaborati dallo script, che vengono salvati in dei file come formato qvd. Questi dati possono poi essere ricaricati da un'altra applicazione Qlik, al fine di effettuare ulteriori elaborazioni o di utilizzare tali dati per creare il data model che esprime il collegamento (relazione) tra le diverse tabelle su cui si appoggerà l'applicazione di presentation.

Solitamente viene utilizzato uno standard per la costruzione di un progetto Qlik, che prevede degli steps definiti, che possono però essere adattati in base alla circostanza del progetto.

1. ETL1 – Estrazione: primo livello di applicazioni QlikSense che estraggono i dati direttamente dalla sorgente e li storicizzano in file qvd (chiamati di livello "Staging"). A questo livello di estrazione è preferibile non effettuare nessuna trasformazione dei dati.
2. ETL2 – Trasformazione: secondo livello di applicazioni QlikSense che caricano i dati salvati nei qvd di Staging e li elaborano tramite trasformazioni e normalizzazioni al fine di rendere i dati fruibili nei termini di interesse del progetto (ad esempio creando misure derivate, aggregando tabelle, dati ecc.). I dati così elaborati vengono salvati in file qvd di secondo livello (chiamati "Associative")
3. Presentation: caricamento delle tabelle del livello Associative e creazione del modello relazionale tra le tabelle (chiamata nuvola dati). In questo caso i dati non vengono più storicizzati ma restano all'interno dell'applicazione; per poterli utilizzare è necessario inserire degli elementi nell'interfaccia che consentano di raggruppare i dati in modo ordinato e specifico per le esigenze specifiche.

Di seguito uno schema classico di strutturazione di un progetto QlikSense:

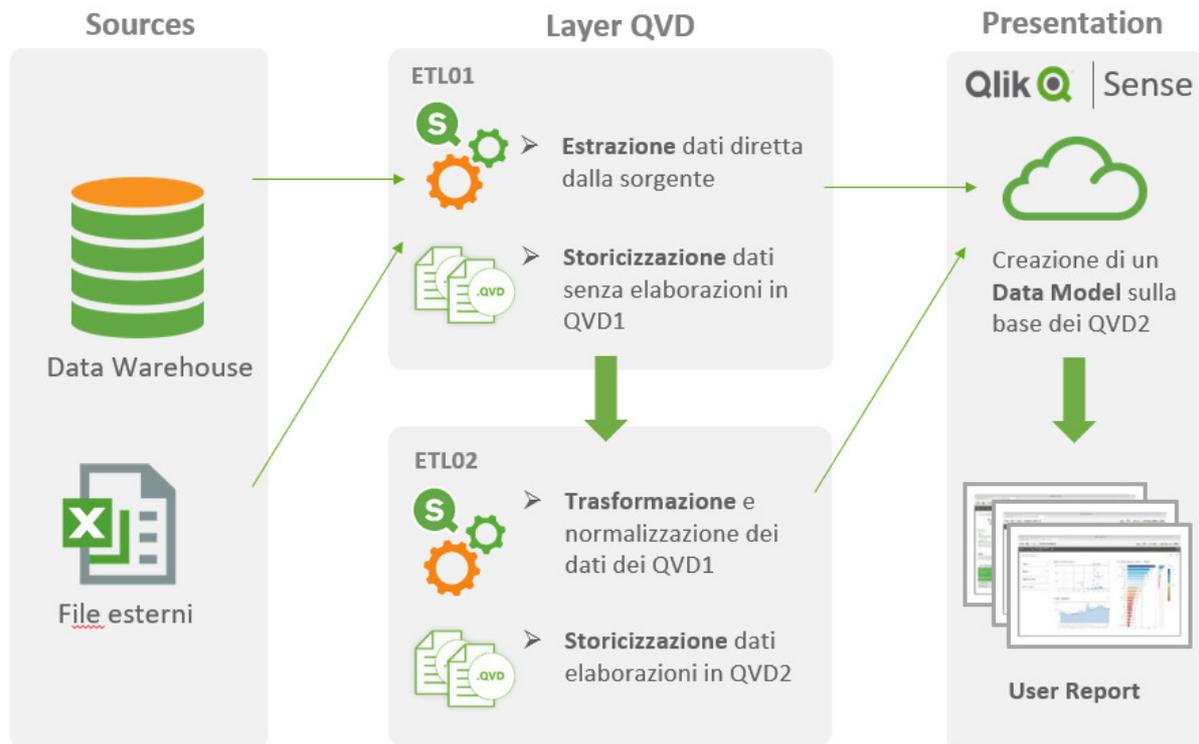


Figura 3. Schema del processo di creazione reports Qlik Sense

2.2.2. Oggetti chiave dell'interfaccia

L'analisi, in Qlik, viene eseguita su fogli navigabili. Ogni foglio può contenere più oggetti (KPI, grafici, tabelle, ecc.) per analizzare il modello di dati sottostanti. Tutti i fogli sono collegati tra loro, per cui le selezioni effettuate su un foglio hanno effetto su tutti gli altri oggetti di tutti gli altri fogli [16].

Di seguito l'elenco dei principali oggetti QlikSense

- **Casella di elenco:** oggetto che presenta i dati presi da una singola colonna di una tabella in modalità "distinct". Vengono visualizzati solo valori univoci, non valori duplicati.
- **Grafici e cruscotti:** oggetti che contengono una o più espressioni che vengono ricalcolate ogni volta che si esegue una o più selezioni. I risultati sono visualizzati come grafici a barre, lineari e a dispersione. Tutti i grafici sono completamente interattivi, quindi è possibile effettuare selezioni o query direttamente puntando il mouse e facendo clic o "selezionando" l'area di interesse (come avviene nel caso di report geografici). Esistono anche i grafici combinati, che presentano nella stessa

visualizzazione sia barre ciascuna delle quali fa riferimento ad un asse diverso, che permette ad esempio il confronto tra valori assoluti e percentuali.

- *Tabelle*: oggetti interattivi le cui selezioni possono essere effettuate direttamente all'interno delle tabelle stesse o in elenchi a discesa selezionabili sulle etichette delle dimensioni. Grazie al modello associativo, è possibile visualizzare in un'unica tabella dati di campi provenienti da tabelle differenti nel data model, ma legate tramite un campo chiave.
- *Oggetto di testo*: oggetto che permette di inserire testi, misure, immagini e collegamenti a pagine web.
- *KPI*: oggetto che permette, previo inserimento della misura o dell'espressione da calcolare, di visualizzare KPI ed eventuali variazioni percentuali degli stessi. È anche possibile aggiungere un collegamento ad un foglio.
- *Mappa Geografica*: oggetto che permette di combinare all'interno di una cartina geografica dinamica, dati geo spaziali e misure, come il numero di lettori per città. I livelli geografici di visualizzazione possono essere per aree, punti, linee, densità, grafici o sfondo. Più livelli possono essere sovrapposti.
- *Mappa ad Albero*: permette la visualizzazione dei dati suddivisi gerarchicamente, mostrando più valori all'interno di uno stesso spazio.
- *Box Plot*: oggetto utilizzato per il confronto di intervalli e distribuzioni su cluster di dati.
- *Misuratore*: oggetto che permette la visualizzazione della posizione di una singola misura, rispetto ad un range impostato che ne determina il valore minimo e quello massimo.

2.2.3. Mashup - introduzione

Un mashup è un'applicazione web che utilizza il contenuto di più di una fonte per creare un'interfaccia grafica per gli utenti.

Grazie alle API è possibile integrarsi facilmente con gli oggetti Qlik Sense per produrre interfacce web arricchite di oggetti Qlik Sense che altrimenti potevano essere consultate solo all'interno delle app di presentation. In questo modo invece è possibile riutilizzare le visualizzazioni di Qlik Sense, comprese le estensioni personalizzate, nonché utilizzare i dati e i calcoli di Qlik Sense [19].

Tramite l'utilizzo di strumenti di filtro e l'interazione con i grafici, è possibile applicare filtri in modo che i dati vengano filtrati allo stesso modo di come accade all'interno delle app Qlik.

Il funzionamento generico è il seguente:

- Utilizzando il metodo `qlik.openApp` si apre il motore associativo da WebSocket a Qlik.
- Gli oggetti usano lo stesso WebSocket, ovvero la stessa sessione, il che significa che sono connessi.
- Gli oggetti Qlik Sense funzionano esattamente come nel client Qlik Sense.

Di seguito uno schema che descrive come viene connesso il mashup ai dati Qlik Sense:

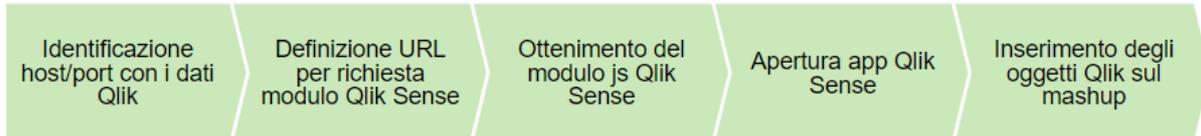


Figura 4. Steps di connessione tra apps e mashup Qlik Sense

Esistono due modi per integrare un oggetto Qlik Sense all'interno del mashup:

- Incorporare un foglio o una visualizzazione in un mashup dall'hub.
- Utilizzare il *single configurator* sull'hub di sviluppo. (Questa funzionalità non è disponibile nelle edizioni cloud di Qlik Sense.)

Il *single configurator* è uno strumento di Qlik Sense che fornisce un modo semplice per creare pagine di mashup senza dover scrivere alcun codice. Restituisce un oggetto Qlik Sense, identificato in un URL. L'oggetto Qlik Sense è in genere una visualizzazione di un'app, ad esempio un grafico a barre.

Il configuratore singolo crea un URL che restituisce una pagina HTML completa che contiene qualcosa di generico, di cui una visualizzazione incorporata è un esempio. Questo URL può essere incorporato in una pagina Web includendolo ad esempio in un `iframe` [19].

3. Il progetto

L'azienda Finance oggetto del progetto di studio, è una società di ricerca e consulenza per gli investimenti, che fornisce ai suoi clienti diversi servizi relativi al rapporto con gli investitori e consulenza strategica. Gli ambiti di cui si occupano principalmente sono:

- Due diligence commerciale / diligenza del venditore
- Note di raccolta fondi, pre-IPO e IPO
- Valutazioni aziendali
- Rapporti strategici, rapporti settoriali e piani aziendali
- Studi di fattibilità, nonché rapporti finanziari e strategici ad hoc

La loro visione aziendale è quella di adottare per i propri clienti un approccio flessibile, che venga in contro alle loro esigenze, offrendo una ricerca approfondita e finanziariamente rigorosa volta a migliorare la visibilità e aumentare la comprensione da parte degli investitori di tutte le società, indipendentemente dalle quotazioni in Borsa.

In tale contesto, la necessità di Finance da cui è scaturita l'ideazione del progetto, è quella appunto di fornire ai suoi clienti uno strumento per il controllo strumentale dei potenziali investitori (Readers), incluse informazioni dettagliate sui documenti che leggono sulla società cliente. I dati che tratterà il progetto non sono pertanto da intendersi prettamente relativi all'ambiente finance inteso come dati di Borsa, ma bensì delle analisi sui Readers dei loro clienti.

3.1. Raccolta dei requisiti

L'attività preliminare da effettuare all'avvio di un progetto di ambito IT è la raccolta dei requisiti utente. Tale attività è di fondamentale importanza per la buona riuscita del progetto stesso. Alla base di tale processo vi è un attento ascolto di quali sono gli obiettivi che il progetto deve raggiungere; tali obiettivi vengono espressi tramite dei requisiti determinati dall'utente. A tali requisiti corrispondono delle funzionalità che il software dovrà possedere per soddisfarli. Talvolta tali requisiti non sono esplicitamente dichiarati dall'utente, ma bensì sono da sottintendersi, per varie motivazioni; in alcune occasioni infatti il cliente non sa esprimere un certo requisito, oppure non sa di averlo: tali requisiti *latenti* necessitano di essere scoperti da parte dei consulenti che prendono in carico il progetto.

Nel nostro caso, dopo alcune riunioni e un'attenta elaborazione delle informazioni raccolte, sono stati individuati i seguenti requisiti:

- L'applicazione deve essere in grado di fornire ai clienti informazioni di dettaglio sui Readers dei loro documenti
- L'applicazione deve clusterizzare i Readers in base a delle caratteristiche comuni
- L'applicazione deve poter permettere un'analisi delle letture nel tempo

- L'applicazione deve poter permettere un'analisi delle letture per singolo documento
- L'applicazione deve essere di facile utilizzo, anche da parte di clienti poco esperti
- L'applicazione deve avere un layout accattivante
- L'applicazione non deve contenere analisi di complessa comprensione
- L'applicazione deve essere di facile accesso per i clienti
- L'applicazione deve evidenziare quali sono i "best" nei dati
- L'applicazione deve poter mantenere l'anonimato del singolo Readers
- Deve essere possibile monitorare la qualità dei dati oggetto di analisi (*)

(*) tale requisito è emerso successivamente alla raccolta dei requisiti, durante l'analisi dei dati descritta nel paragrafo successivo.

3.2. Caratteristiche del caso di studio

Alla luce di tali requisiti, è stato possibile iniziare a pensare in termini di funzionalità e caratteristiche che l'applicazione deve avere. Per fare ciò è però stato necessario avere evidenza di quali sono i dati messi a disposizione, quali attributi è possibile usare e la mole di dati che dovrà essere accolta dall'applicazione. È emerso che le sorgenti messe a disposizione da cui estrarre i dati sono le seguenti:

1) Piattaforme web dove sono pubblicati i documenti dei clienti di Finance

- Piattaforme partners di Finance
- Sito dell'azienda Finance

Una volta recuperata la struttura tracciati dei flussi che è possibile recepire dalle varie piattaforme, è stato necessario individuare quali sono gli attributi utili per l'analisi in questione, possibilmente utilizzati da più piattaforme. Erano infatti presenti, per ciascuna sorgente, attributi diversi, presenti spesso in una sola piattaforma, ma non presenti nelle altre. Tali campi, anche se potevano risultare utili per le nostre analisi, non potendo essere ricevuti da gran parte delle piattaforme, hanno perso di importanza in quanto avrebbero portato a dei risultati parziali.

A valle dell'analisi degli attributi presenti in ogni piattaforma, sono stati selezionati i campi comuni (o quasi) a tutte le piattaforme, che potevano essere utili per la progettazione della nostra app. In generale, per ogni lettore, le informazioni che sono state selezionate sono le seguenti:

- Reader's Name: non estraibile da tutte le piattaforme. Alcune piattaforme possiedono dei nomi utente falsi, mentre in altre il Reader è anonimo
- Reader's Institution: nome della società per la quale lavora il Reader
- Investor Type -> tipologia di investitore (riferito al Reader)

- Reader's Location: città e Stato del Readers
- Reader's email: email del Readers
- Read date: data e ora di lettura del documento
- Document ID: ID del documento letto

Tali informazioni anche se non sempre presenti in tutte le piattaforme, sono emerse come quelli più utili per il nostro scopo.

2) Anagrafica Readers

L'azienda Finance avrebbe messo a disposizione un database in cui erano presenti, per ciascun Reader, dei dati anagrafici relativi all'azienda di appartenenza. Tali dati, come i precedenti sono stati analizzati al fine di individuare quelli più utili per il progetto:

- Reader's Name: Nome del Reader
- Reader's Email: Email del Reader
- Reader's Institution: nome della società per la quale lavora il Reader
- Investor Type: tipologia di investitore
- Reader's Location: città e Stato del Reader
- Read date: data e ora di lettura del documento

È importante notare che la tipologia di dato individuato è la stessa delle piattaforme. Questo perché in fase di riunione è emerso per Finance la necessità di capire quanto i dati provenienti dalle piattaforme siano affidabili e congrue con quelli a loro disposizione (presenti in anagrafica). Per poter effettuare tale confronto è necessario prendere le stesse tipologie di dato da entrambe le strutture e metterle a confronto. Tale punto emerso può essere considerato come un nuovo requisito, da cui è emersa una nuova esigenza per Finance: quella di avere uno strumento complementare di data quality, utilizzabile solo a livello operativo interno, con cui verificare la bontà dei dati provenienti dalle piattaforme.

3) Dotmailer

L'azienda in questione utilizza un software di email marketing, tramite cui invia delle mail agli investitori, annunciando la pubblicazione di nuovi documenti delle aziende clienti. All'interno di tale software sono quindi presenti i riferimenti email di tali investitori (i nostri possibili Readers), le mail loro inviate, le aperture di tale mail e i click effettuati sugli URL che rimandano ai documenti in questione.

Tra i dati raccolti da tale software sono stati evidenziati come utili per le nostre analisi i seguenti attributi, relativi al reader:

- Reader email: indirizzo email del lettore
- Date open email: Data di apertura della mail

- Date click: Data di click dell'URL che porta al documento (non presente per tutte le mail aperte)
- URL: serve per individuare il documento presente nella mail, tramite un determinato codice chiamato CMS

Ogni volta che un Reader apre una mail e clicca sull'URL suggerito, viene aperto il documento, dunque tale indicazione viene interpretata da Finance come equivalente alla lettura di tale documento da parte del Reader.

4) Documenti (xml)

I documenti sono postati sul web tramite appositi xml. Questi ultimi contengono tutte le informazioni necessarie per identificare i dettagli del documento di appartenenza. In particolare è possibile ricavare i seguenti dati, utili per le nostre analisi:

- ID del documento
- Titolo del documento
- Data di pubblicazione del documento
- CMS del documento: un altro identificativo documento che viene utilizzato da Dotmailer e Google Analytics per identificare il documento
- Nome dell'azienda oggetto del documento
- Settore dell'azienda oggetto del documento
- URL dove è pubblicato il documento

3.3. Creazione della proposta: mockup dell'interfaccia

A valle dell'analisi delle sorgenti dati e degli attributi disponibili, è stato possibile creare un mockup dell'interfaccia web, da visualizzare con Finance al fine di stabilire la modalità di visualizzazione e fruibilità dell'app, cercando da subito di ricoprire i requisiti precedentemente raccolti.

La costruzione del mockup è stata progettata sulla base dei seguenti fattori, che devono essere considerati contemporaneamente al fine di poter creare uno strumento adeguato:

- Requisiti raccolti: l'interfaccia dovrà soddisfare in primis i requisiti raccolti
- Visualizzazione in interfaccia web: le analisi devono essere adatte per la consultazione su interfaccia web (dunque ad esempio è meglio evitare inserire troppe tabelle, ma è meglio prediligere grafici che diano visibilità immediata delle informazioni)
- Grafici e analisi realizzabili su QlikSense: tale strumento utilizza un set di oggetti grafici limitato, pertanto occorre pensare alle analisi possibili con gli oggetti disponibili
- Tipologia di dati disponibili dalle sorgenti: i dati messi a disposizione dalle sorgenti limitano il range di analisi che è possibile effettuare.

Dopo vari schemi e prove di implementazione, il risultato finale è stata la creazione del seguente mockup dell'interfaccia, composto da 5 sezioni:

- Homepage
- Data Analysis
- Trend Analysis
- Geo Analysis
- Readers Report

La prima figura mostra il mockup della Home Page, in cui vengono identificati una serie di KPI utili per l'utente:

- Report published: numero di documenti pubblicati
- Total reads: numero di letture dei documenti
- Unique readers: conteggio distinto del numero di lettori dei documenti
- Country reached: conteggio distinto delle country da cui provengono i lettori
- New readers: numero di utenti che hanno fatto una sola lettura (e che quindi è possibile considerare come nuovi lettori)

La parte alta del mockup dell'interfaccia mostra, sul primo livello, le altre pagine di analisi in cui l'utente potrà navigare. Il secondo livello mostra invece dei filtri che sarà possibile applicare alle analisi, come l'intervallo di tempo da considerare.

Nella parte destra del mockup è presente un grafico a barre che consente di vedere l'andamento nel tempo delle letture. Sotto tale grafico sono presenti dei grafici a torta che mostrano le top 5 delle letture, suddivise per differenti caratteristiche del lettore:

- Top Investor type
- Top Institution
- Top location

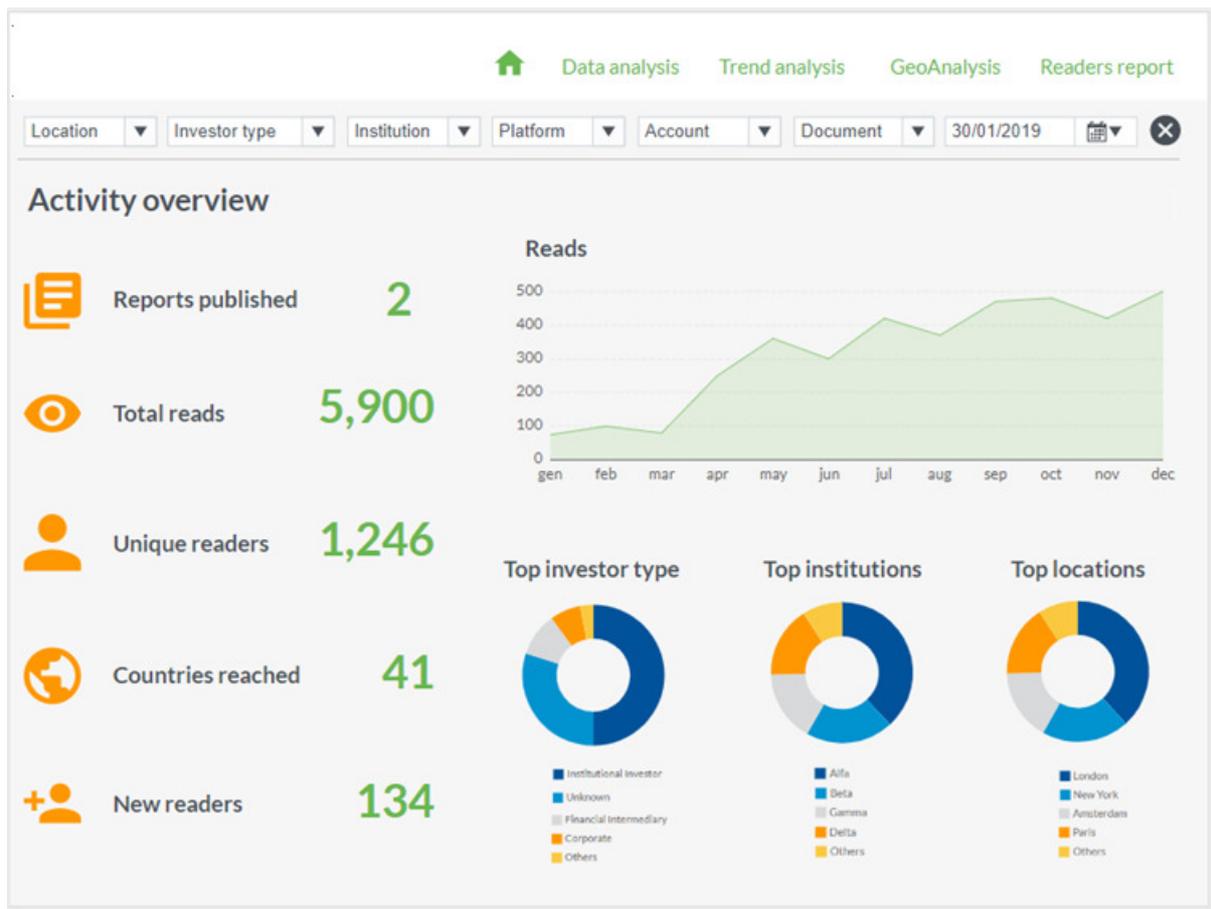


Figura 5. Mockup Home Page

La seconda figura rappresenta il mockup della pagina Data Analysis in cui sono presenti due sezioni: quella di sinistra dedicata al numero dei lettori, suddivisi per caratteristiche comuni, mentre la parte destra è dedicata alla visualizzazione dei documenti più letti, per quelle stesse caratteristiche del lettore.

E' possibile selezionare il numero di caratteristiche e documenti top su cui fare l'analisi, ad esempio la top 5 per investor type permette di visualizzare da un lato la suddivisione percentuale delle tipologie di investitore più comuni dei lettori, mentre sul lato destro è possibile vedere quali sono i documenti più letti da queste tipologie di investitore.

Tali analisi cross vengono fatte sui seguenti attributi dei lettori:

- Investor Type
- Institution
- Platform
- Location

Nella parte inferiore del mockup dell'interfaccia è presente un grafico a barre che indica quali sono stati i documenti più letti, indipendentemente dalle caratteristiche del lettore; una tabella sotto quest'ultimo, indica il numero e la percentuale di letture di questi documenti.



Figure 6-7-8. Mockup pagina Data Analysis

Il mockup della pagina Trend Analysis è così strutturato: nella parte alta è presente l'elenco dei documenti pubblicati, sopra cui è possibile cliccare per visualizzare il documento in questione, e la data di pubblicazione di questi ultimi.

Sotto un grafico di trend mostra l'andamento delle letture nel tempo, in modo analogo alla homepage.

Seguono altri due grafici di trend, con l'andamento delle letture nel tempo, in uno suddivise per piattaforma di appartenenza e nell'altro per documento.

Infine la parte finale della pagina mostra una serie di KPI legati esclusivamente alle letture avvenute tramite il sito di Finance e le mail inviate da quest'ultimi ai potenziali lettori:

- Unique visitors: conteggio delle letture dei documenti dal sito Finance
- Avg visit duration: media della durata delle letture dei documenti (in minuti)
- Research Link Visitors: numero di utenti che accedono al sito Finance tramite link diretto al sito
- E-mail opens: numero di e-mail aperte dai potenziali lettori
- E-mail click throughs: numero di e-mail in cui è stato cliccato l'url al link che rimanda al documento da leggere

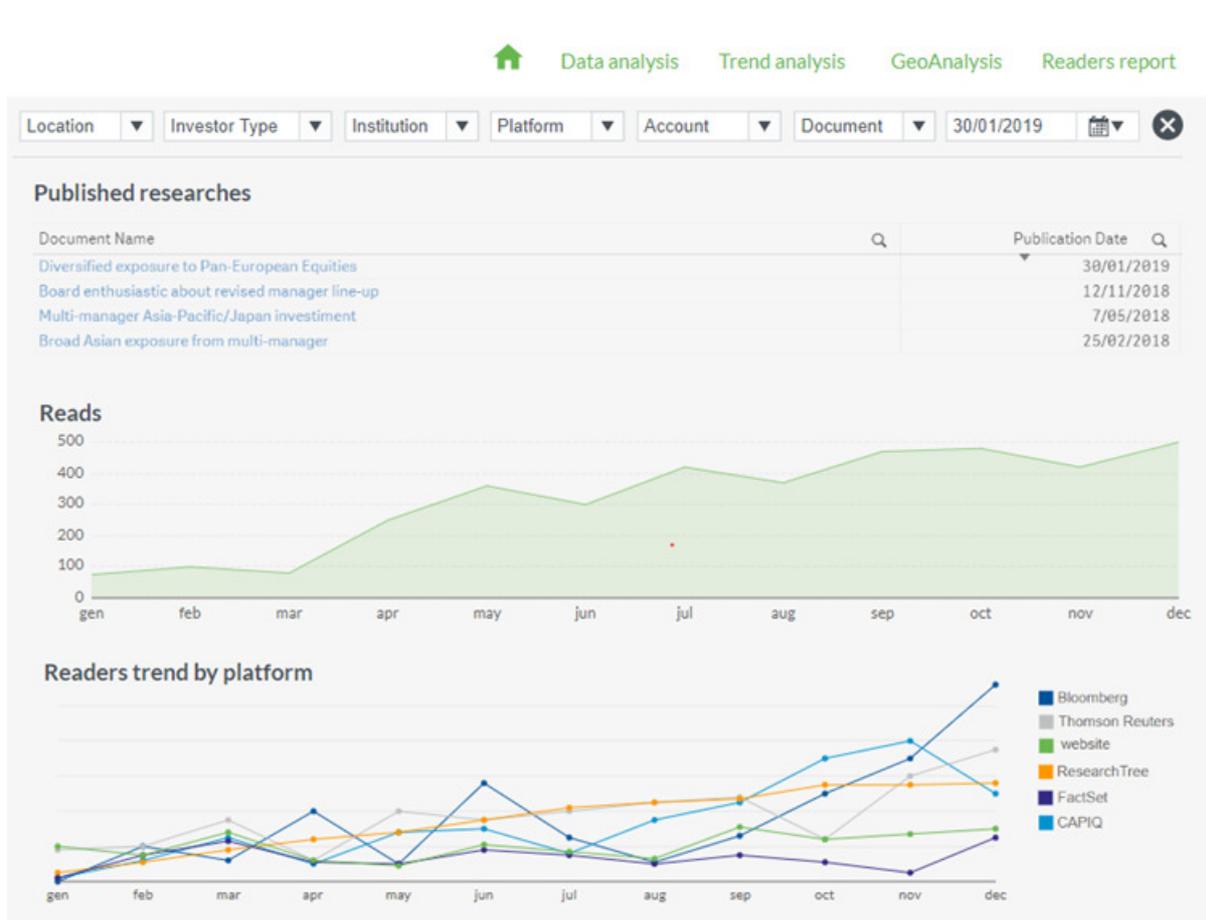


Figura 9. Mockup pagina Trend Analysis

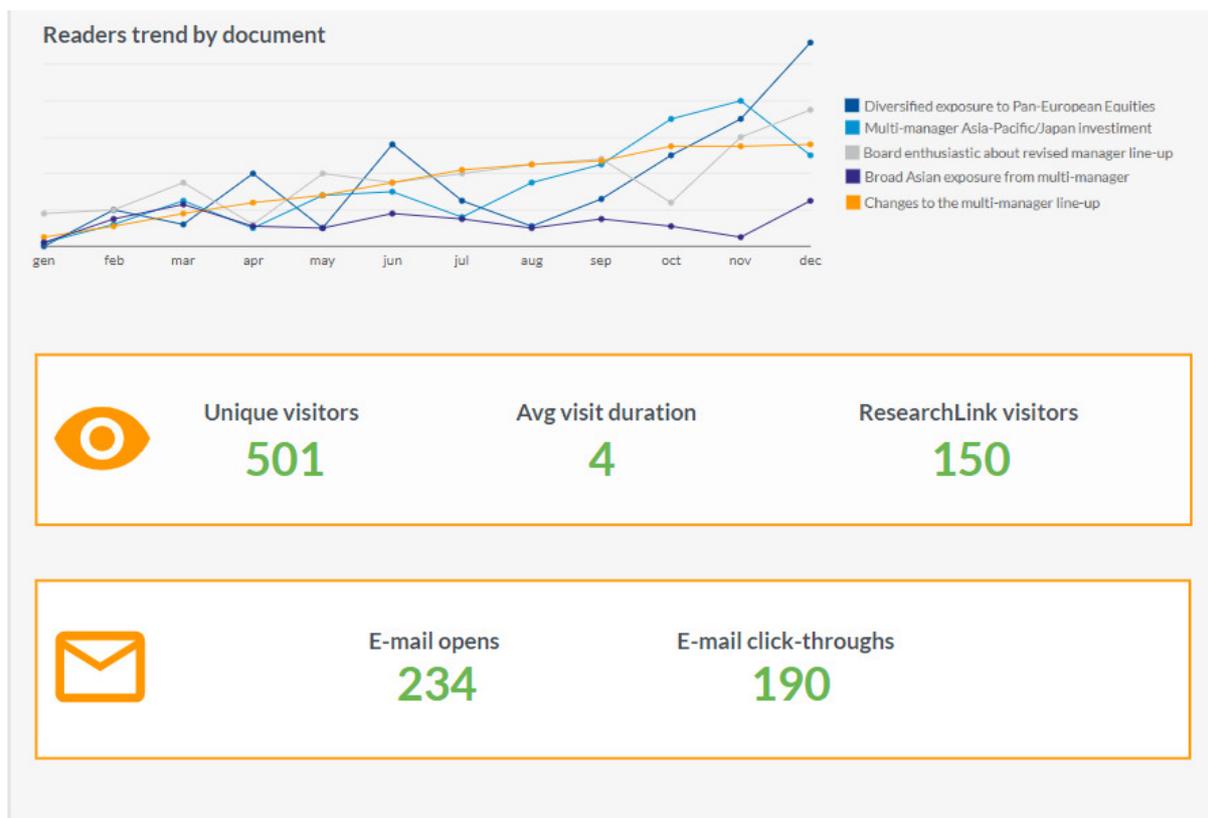


Figura 10. Mockup pagina Trend Analysis

Il mockup della pagina “Geo Analysis” mostra la distribuzione delle letture a livello geografico. Una mappa geografica interattiva mostra tramite una mappa di calore quali sono le aree in cui sono presenti i letti: maggiore è il numero di lettori più tendente al rosso sarà la zona individuata.

Due grafici a barre sotto mostrano come sono distribuiti i lettori tra i vari continenti, suddividendoli per investor type e piattaforma di provenienza dell’informazione di lettura

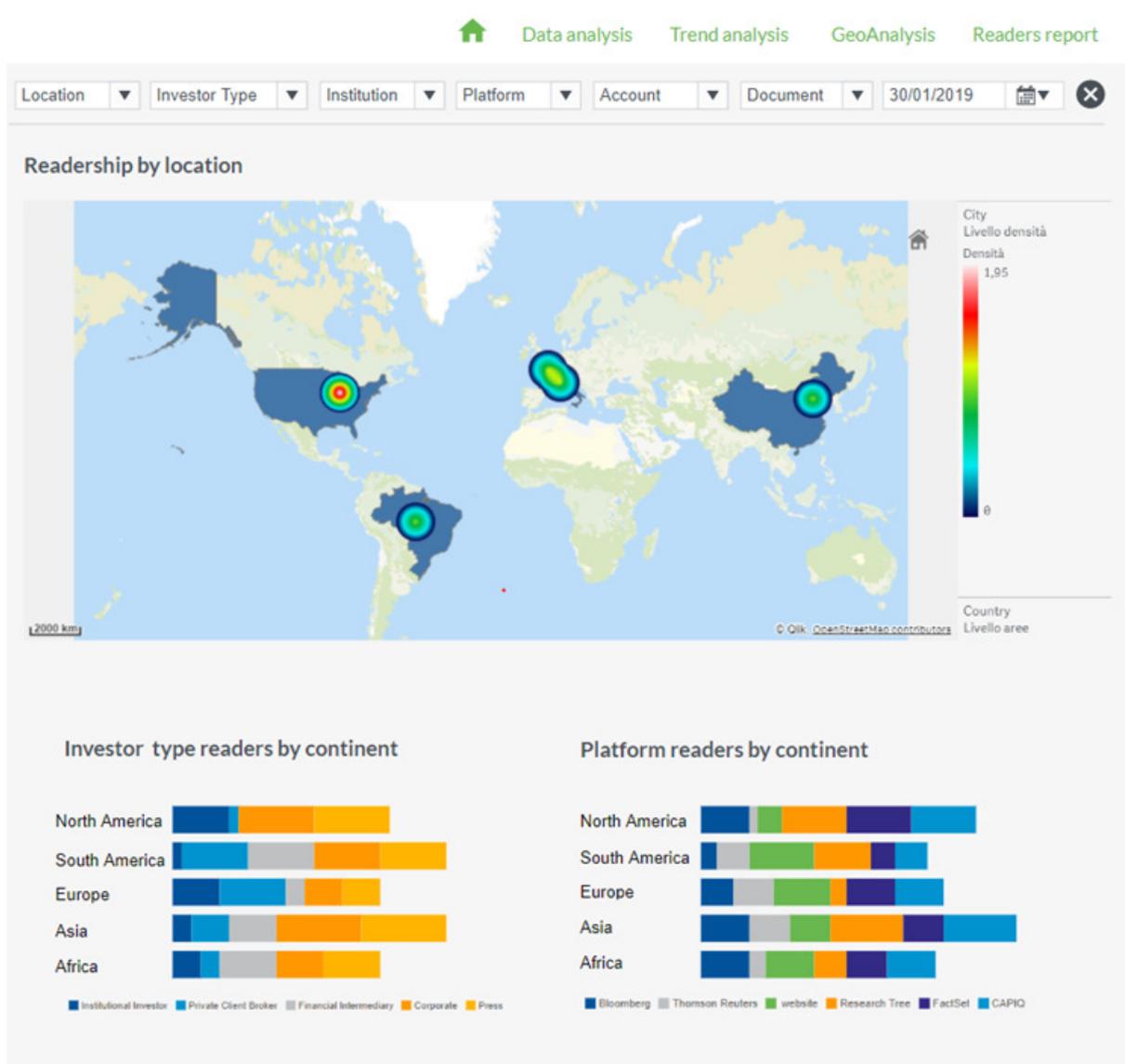


Figura 11. Mockup pagina Geo Analysis

L'ultima pagina del mockup mostra la pagina di Readers Report, in cui per ciascuna azienda di cui fanno parte i lettori, vengono individuati i seguenti fattori

- Tipo di azienda
- Località
- Numero di letture dal sito di Finance
- Numero di click agli url dei documenti nelle mail
- Numero di mail aperte
- Numero di letture tramite le piattaforme partner
- Totale delle letture
- Numero di lettori distinti

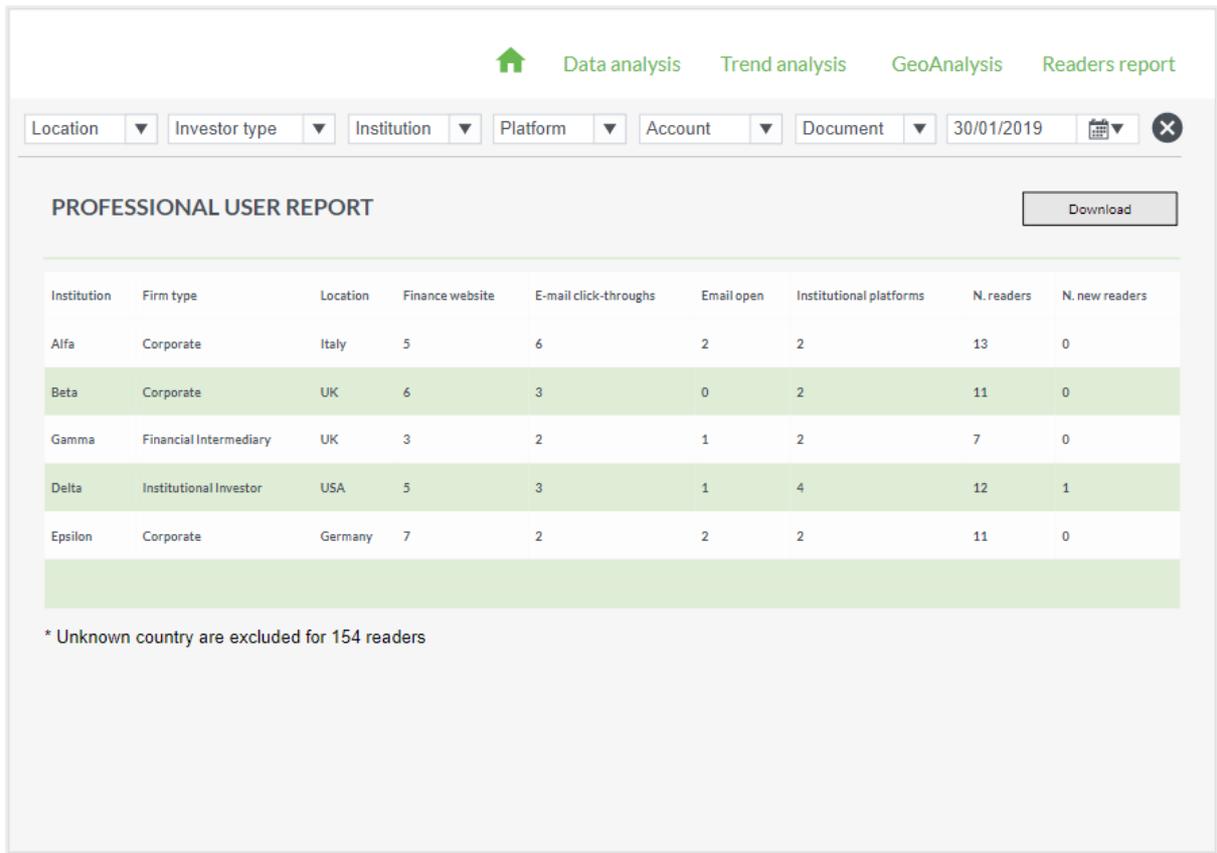


Figura 12. Mockup pagina Readers Report

3.4. Gestione del progetto

La gestione del progetto è un'attività fondamentale per la buona riuscita dello stesso, pertanto è necessario che essa venga fatta sin dalle fasi iniziali, attraverso la valutazione dei tempi, dei costi e delle risorse necessarie per tutto il ciclo di vita del progetto.

Partendo dalle esperienze precedenti e dalle competenze acquisite sul campo, il project manager scompone il progetto in macro attività, strutturandole in modo sequenziale all'interno di un Gantt. Tale modello aiuta a definire le fasi del ciclo di vita del progetto, stabilendo l'approccio interno che si intende adottare e descrivendo il processo che si vuole seguire.

Al fine di pianificare le giornate di progetto, è dunque necessario cercare di suddividere il progetto nelle macroattività, stimando per ciascuna di esse l'effort necessario e stabilendo quali sono le risorse necessarie da allocare e per quanto tempo. Inoltre è utile prevedere sul totale del progetto un certo numero di giornate che saranno utilizzate per attività di coordinamento.

Nel nostro caso il progetto è stato suddiviso nelle seguenti macro attività:

- *Scoping*: attività di analisi volte ad individuare cosa dovrà essere visualizzato sull'interfaccia e come.
- *SW Installation*: attività di installazione del tool e messa a punto dei server su cui verrà inserito il progetto
- *Data modeling*: attività di estrazione e modellazione dei dati
- *Build*: costruzione del modello dati e del presentation (comprensivo di interfaccia web)
- *UAT*: attività di test sui dati da parte degli utenti e contestuale risoluzione dei bug emersi.
- *PM (Project Management)*: attività trasversali di coordinamento del progetto e comunicazione con il cliente.

In figura è rappresentato il gantt di progetto, secondo due tipologie di viste:

- Macro attività e risorse allocate nel tempo
- Macro attività nel tempo

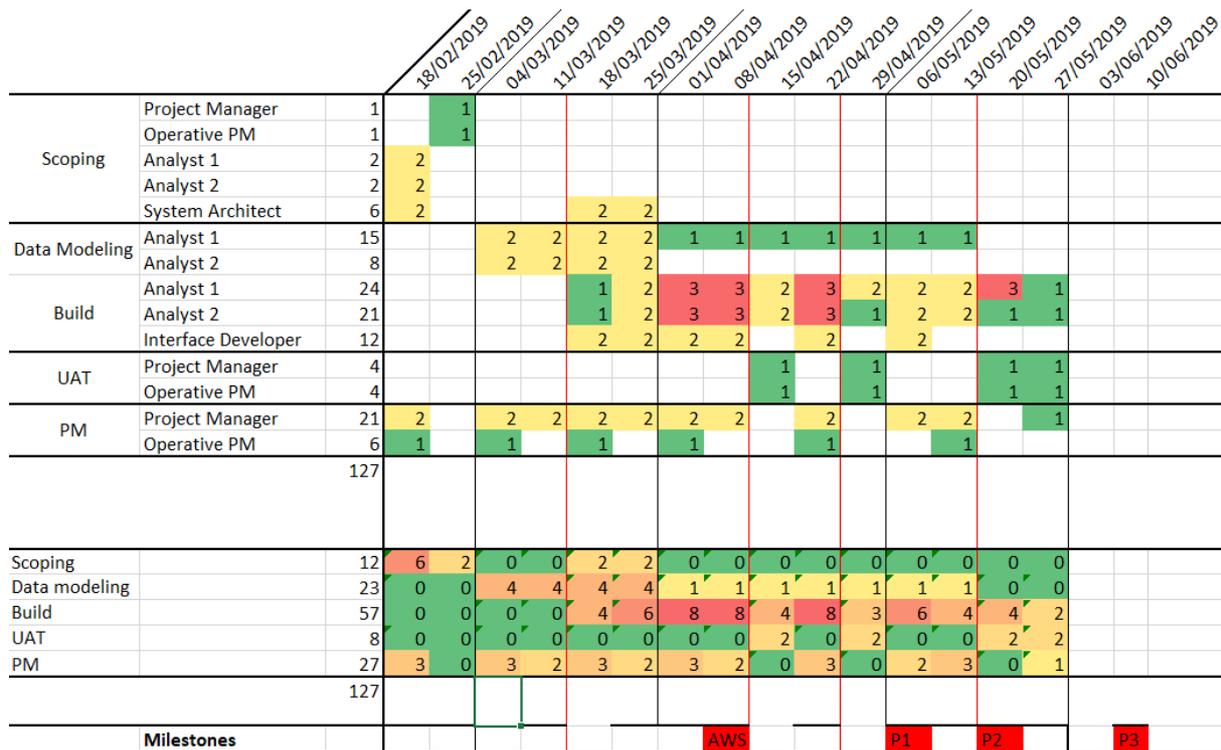


Figura 13. Gantt per macro attività e risorse

Come è facile notare dalla figura, le due viste evidenziano lo stesso effort totale; ciò che cambia è il grado di dettaglio rappresentato. In generale, i numeri presenti all'interno di ogni cella identificano le giornate/uomo settimanali, ad eccezione della colonna dei totali, che per ciascuna riga identifica il numero totale di giornate preventivate per il progetto. Sono presenti tre milestones, che permetteranno di rilasciare parti dello strumento finale, in modo che eventuali anomalie o richieste di modifica vengano prese in carico dal team di sviluppo prima del completamento dello stesso.

Nel primo gantt è possibile vedere quali risorse lavoreranno sul progetto, su quali macroattività e per quanto tempo; questa suddivisione è molto utile a livello operativo interno in quanto permette di sapere per ciascuna risorsa qual è il suo tempo di allocazione settimanale per tutta la durata del progetto, consentendo di monitorare l'effort e i progressi di ciascun individuo.

Le risorse che sono state individuate per l'intero processo sono:

- *Project Manager*: rappresenta il capo progetto, colui che ha il compito di valutare e monitorare costi e tempo di raggiungimento degli obiettivi prefissati sulla base del piano
- *Operative Project Manager*: collabora con il Project Manager sul monitoraggio di costi e tempistiche, ma attua anche attività di coordinamento del team, supportandolo nella risoluzione di issues e problematiche che emergono durante gli sviluppi. Si fa carico della comunicazione con il cliente degli aspetti tecnici e funzionali da chiarire, fungendo da collante comunicativo tra il team tecnico e l'utente finale.
- *Business Analyst*: si occupa della parte di ingegnerizzazione della soluzione, sviluppando il progetto secondo i requisiti emersi in fase di analisi, studiando la soluzione più appropriata alla circostanza, cercando di far fede alle best practices dello

strumento. Qualora emergono problematiche o issues che per essere risolte necessitano della collaborazione del cliente, supporta il project manager nella comunicazione con il cliente, fornendo tutti i dettagli tecnici necessari. In tale progetto questo è stato il mio ruolo.

- *System architect*: si occupa dell'installazione e messa a punto dei server necessari per l'utilizzo di QlikSense, previa analisi del dimensionamento server in grado di supportare il progetto ed eventuali evoluzioni dello stesso, Inoltre provvede all'installazione di tutti i pacchetti software Qlik necessari per il progetto.
- *Interface developer*: analista specializzato nello sviluppo della parte web interface di Qlik, tramite l'utilizzo del mashup. Si occupa dunque di inserire gli oggetti e i grafici creati su QlikSense dagli analyst all'interno di una interfaccia web.

Nel secondo gantt viene semplicemente fornito l'effort settimanale per ogni macro attività; tale tipologia di gantt di "alto livello" è solitamente utilizzata per la condivisione del piano e il monitoraggio dello stesso insieme al cliente, il quale ha interesse non sulle risorse allocate dal fornitore, ma sui i tempi di realizzazione delle macro attività. Il progresso e l'avanzamento viene continuamente monitorato su questo gantt dove quando necessario vengono apportate le opportune modifiche, dovute ad esempio a delle variazioni di requests rispetto al piano originale.

3.5. Analisi e sviluppo

La fase di sviluppo del progetto è senza dubbio molto delicata; al fine di facilitare tali attività è importante che le fasi precedenti vengano eseguite con cura. Nel nostro caso l'azienda Finance ha gradito il mockup proposto, chiedendoci di effettuare solo alcuni cambiamenti non strutturali, per lo più pertinenti alle modalità di visualizzazione dei dati, oltre che alla sostituzione/rimozione di alcuni KPI e grafici. E' stato pertanto possibile procedere con lo sviluppo del progetto.

Come precedentemente anticipato nel paragrafo dedicato a QlikSense, l'implementazione del progetto prevede alcuni steps ben precisi, che devono essere rispettati al fine della realizzazione di un progetto solido ed automatizzato, che possa nel tempo mantenersi da solo senza bisogno di ulteriori interventi, se non dovuti a volontarie modifiche ed evoluzioni dello stesso.

In generale, gli steps previsti per la realizzazione di un'app QlikSense sono:

- Progettazione del data model: studio delle sorgenti dati e dell'output finale, al fine di stabilire quale debba essere la struttura tabellare più adatta per la creazione del modello dati
- Estrazione dei dati (ETL1): processo di estrazione dei dati dalle differenti sorgenti disponibili
- Costruzione del data model (ETL2): attività di trasformazione, aggregazione e normalizzazione dei dati al fine di creare il modello dati adatto alle esigenze progettuali
- Presentantation: costruzione dell'app QlikSense che legge i dati precedentemente elaborati e li utilizza all'interno degli oggetti grafici di analisi

- Mashup: inserimento degli oggetti QlikSense all'interno di un'interfaccia web apposita, che verrà resa accessibile all'utente

3.5.1. Progettazione del data model

La progettazione del modello dati che dovrà essere creato sull'applicazione è un'operazione che viene effettuata tenendo presenti vari fattori, tipici di un progetto QlikSense:

- *La struttura tabellare del modello dati*, che permette di organizzare i dati con una logica simile a quella di un database relazionale, in cui sono presenti delle tabelle di anagrafica ma anche delle tabelle cosiddette “dei fatti” in cui sono presenti le misure e le metriche non derivabili. Ciò permette di ottimizzare lo spazio ed evitare la ridondanza dei dati tra più tabelle.
- *La logica associativa tra i dati*, che consente di mantenere un sistema relazionale tra differenti tabelle, in modo che un filtro applicato su un dato di una tabella si ripercuota sulle altre tabelle che possiedono un collegamento con quest'ultima, sia che sia diretto o anche indiretto.
- *Le tipologie di dati esistenti*, quelli derivabili tramite calcolo a frontend (tramite formule) e quelli da precalcolare in backend, ad esempio per semplificare le logiche o diminuire la pesantezza di calcolo runtime che dovrebbe altrimenti sostenere la piattaforma.

Dopo varie prove e ragionamenti, il modello associativo iniziale individuato è così composto:

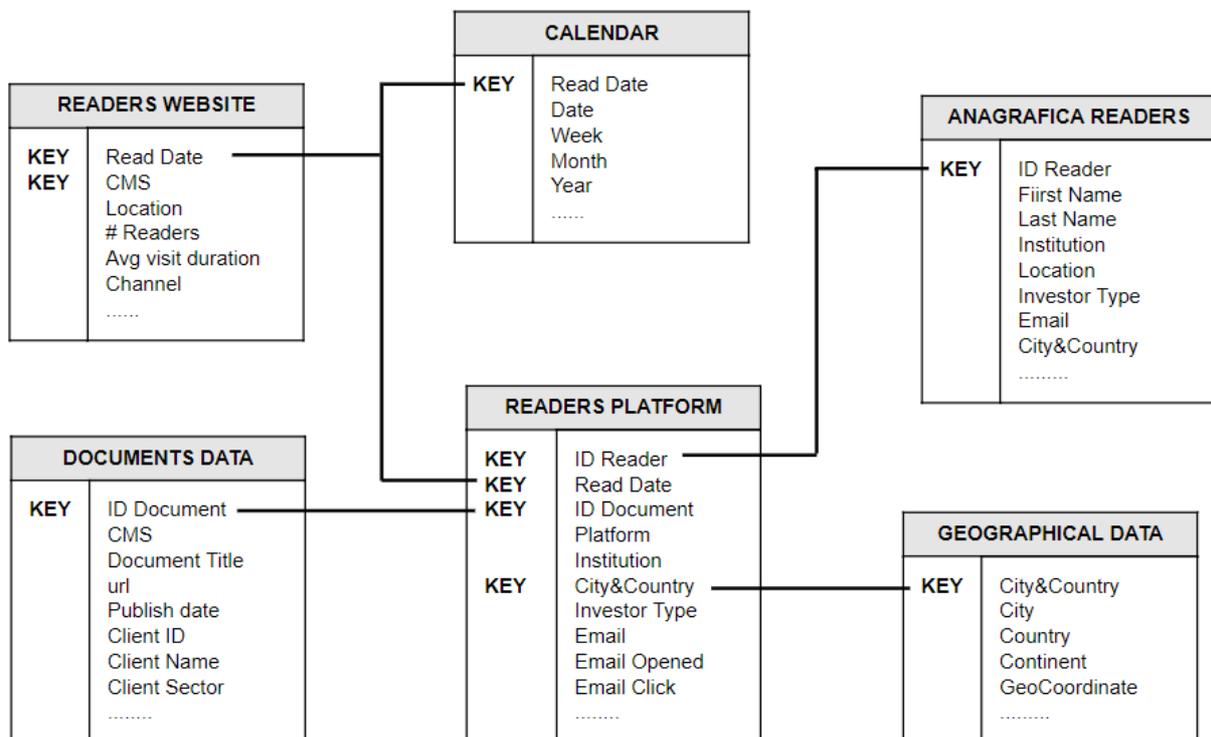


Figura 14. Prototipo del Data Model da sviluppare

Tale schema di modello è stato semplicemente abbozzato con carta e penna più volte, fino ad ottenere una struttura soddisfacente che rispetti i requisiti sopra descritti.

E' stata individuata come soluzione una struttura relazionale a stella in cui l' entità centrale è costituita dalla tabella dei fatti "READERS PLATFORM" contenente le informazioni base relative a ciascuna lettura avvenuta nelle varie piattaforme. Per ciascuna lettura (che corrisponde ad una riga della tabella) saranno presenti i seguenti attributi:

- ID Reader (campo chiave): nome e cognome del Reader (quando presente)
- Read Date (campo chiave): data di lettura
- ID Document (campo chiave): ID identificativo del documento letto
- Platform: nome piattaforma da cui proviene la lettura
- Institution: nome dell'azienda a cui appartiene il Reader
- City & Country (campo chiave): Identificativo luogo dell'azienda
- Investor Type: tipologia di investitore (riferito al Reader)
- Email: email del Reader
- Email opened: dato compilato solo se siamo in presenza di una mail aperta
- Email click: dato compilato solo se siamo in presenza di una mail cliccata

In questa tabella sono presenti 4 campi chiave che permettono di collegare i dati con altri attributi, soprattutto di anagrafica, inseriti in altre tabelle. In questo modo è possibile avere una collegamento tra i dati, rendendo questi ultimi più compatti e limitandone la ridondanza.

Il campo chiave "ID Reader" collega il Reader con la sua anagrafica (tabella ANAGRAFICA READERS) proveniente da Dotmailer, in cui per ciascun lettore censito (non anonimo o falso) abbiamo dati come Institution, Location, Investor Type, Email ecc. Da notare che questi dati sono anche presenti nella fact table, ma l'origine è diversa; mentre i dati della fact table provengono dalle singole piattaforme da cui è arrivata la lettura, i dati della tabella di anagrafica provengono da Dotmailer. Come precedentemente detto, Finance ha richiesto di poter monitorare la bontà dei dati che riceviamo dalle piattaforme (quindi se questi sono concordi a parità di Reader con i dati di Dotmailer), dunque occorre mantenerli entrambi sul data model per poter creare un confronto.

Il campo chiave "Read Date" collega la data di lettura con due tabelle: READERS WEBSITE e CALENDAR. La Tabella CALENDAR, permette di poter associare a ciascuna data di dettagli del calendario di riferimento, come il mese, il giorno, la settimana e l'anno di appartenenza della data. E' comodo per quando si desiderano fare delle aggregazioni temporali nelle selezioni, dunque è utile utilizzarlo in tutti i progetti in cui vengono visualizzati dati con riferimenti nel tempo. La tabella READERS WEBSITE contiene i dati delle letture provenienti dal sito di Finance, estratti da Google Analytics; analizzando la struttura che potevamo ottenere da quest'ultimo infatti, ci si è resi conto che i dati non sono estratti per singola lettura ma sono aggregati per numero di letture che hanno stesse caratteristiche, in quanto il lettore non viene censito come univoco, ma risulta anonimo. Il risultato pertanto è, a parità di data di lettura, CMS (che identifica l'ID del documento) e Location, delle misure come il numero delle letture, il tempo medio di visita dei documenti e il canale tramite cui viene raggiunto il sito. Tali misure permettono di calcolare dei KPI

appositi per le letture sul sito Finance, che sono stati inseriti all'interno del mockup. Essendo il livello di aggregazione diverso rispetto a quello della Fact Table, in prima battuta si è pensato di lasciare tale dato in una tabella separata da quest'ultima, collegando i dati per data di lettura.

Il campo chiave "ID Document" permette di collegare la Fact Table con l'anagrafica del documento, grazie alla quale per ciascun documento è possibile individuare attributi come il titolo del documento, l'azienda di appartenenza, il settore dell'azienda, l'URL di collegamento al documento ecc.

Infine il campo chiave "City & Country" permette di collegare la fact table con i dati utili per l'analisi geografica che sarà creata tramite l'utilizzo delle mappe Qlik. La tabella GEOGRAPHICAL DATA è infatti un'anagrafica delle città, per ciascuna delle quali sono stati individuati dati come le coordinate geografiche del punto, la country, il continente di appartenenza ecc. Queste informazioni sono reperibili da appositi file interni QlikSense, che agevolano il corretto utilizzo dell'oggetto mappa, tramite l'implementazione di questi ultimi sul data model.

3.5.2. ETL1 – Estrazione dati

L'estrazione dei dati dalla sorgente comporta uno studio preliminare delle connessioni necessarie, al fine di rendere possibile tali attività. Ciascuna connessione ha bisogno di adeguati parametri di accesso che variano in base al tipo di connessione e alle autorizzazioni disponibili.

Il progetto in esame necessita delle seguenti tipologie di connessione:

- FTP
- SFTP
- API
- File Web
- Path a cartelle di progetto
- ODBC

Per ciascuna sorgente occorre stabilire le modalità di estrazione e storicizzazione dei dati, ragionando in termini di caricamento full dalla sorgente oppure incrementale, quando possibile, al fine di estrarre di volta in volta solo i dati nuovi e storicizzandoli insieme allo storico già esistente.

Nei seguenti paragrafi analizzeremo la suddivisione degli ETL, le modalità di estrazione e la logica di storicizzazione implementata.

Al fine di gestire al meglio la manutenzione degli ETL, è stato stabilito di creare 9 ETL, uno per ciascuna sorgente; tale scelta permette di controllare in modo separato il processo di estrazione di ogni singolo ETL, evitando che eventuali problemi di estrazione su una sorgente si ripercuotano anche sulle altre estrazioni. Infatti, anche se possibile estrarre più sorgenti dati da un solo ETL, tale scelta implica che un errore nel caricamento da una sorgente, provoca il

blocco dell'intero processo di caricamento, anche da parte di altre sorgenti che non avrebbero avuto problemi di estrazione. Questo è dovuto al fatto che in ciascuna app QlikSense (sia ETL che Presentation), una volta andato in esecuzione lo script, se è presente un errore, il processo si blocca e il caricamento va in fallimento. Dividere invece gli ETL in più sorgenti comporta che se l'ETL di una sorgente va in errore, gli altri ETL possono essere eseguiti indipendentemente. A livello di manutenibilità tale soluzione ci è apparsa la migliore, considerando soprattutto la quantità di connessioni differenti da implementare.

Di seguito uno schema degli ETL di estrazione dati che sono stati creati, con le principali informazioni sulle modalità di estrazione e storicizzazione dei dati

Nome ETL - Source	Descrizione estrattore	Tipo di connessione	formato files	Tipo di estrazione	Modalità di storicizzazione	Range di storicizzazione
ELT1 – Bloomberg	letture da piattaforma Bloomberg	SFTP	txt	Incrementale - nuovi files	qvd per Anno/mese	3 anni + 1 Rolling
ETL1 – CAPIQ	letture da piattaforma CAPIQ	Share directory	xls	Incrementale - nuovi files	qvd per Anno/mese	3 anni + 1 Rolling
ETL1 – FactSet	letture da piattaforma FactSet	SFTP	csv	Incrementale - nuovi files	qvd per Anno/mese	3 anni + 1 Rolling
ETL1 – Research Tree	letture da piattaforma Research Tree	API	N/A	Incrementale - Day Back 1	qvd per Anno/mese	3 anni + 1 Rolling
ETL1 – Thomson Reuters	letture da piattaforma Thomson Reuters	FTP	csv	Incrementale - nuovi files	qvd per Anno/mese	3 anni + 1 Rolling
ETL1 – Google Analytics	letture sito Finance da Google Analytics	Qlik Web connectors	N/A	Incrementale - Day Back 6	qvd per Anno/mese	3 anni + 1 Rolling
ETL1 – Dotmailer	email aperte e letture tramite link da Dotmailer	API	N/A	Full	unico qvd per tabella	Full
ETL1 – Sales Force	dati Readers da anagrafica	ODBC - Query	N/A	Full	unico qvd per tabella	Full
ETL1 – xml	dati sui documenti pubblicati	Share directory	xml	Incrementale - nuovi files	unico qvd per tabella	Full

3.5.2.1. ETL1 – Bloomberg

Questo ETL permette di estrarre i dati delle letture provenienti dalla piattaforma bloomberg tramite protocollo di rete SFTP.

SFTP indica il trasferimento di file sicuro tramite Secure Shell (o SSH), da qui il nome SSH File Transfer Protocol (SFTP). I due dispositivi (server e client) comunicano tramite un unico canale, quello di SSH, in cui vengono inviati i comandi di gestione dei file remoti e dove avviene il trasferimento dei file. L'unica porta necessaria per SFTP è di default la 22, la stessa di SSH [11].

Per poter estrarre i dati da file accessibili tramite SFTP, è necessario dapprima creare la connessione tramite i Web Connectors di Qlik.

Di seguito uno screenshot dell'interfaccia WebConnectors da cui è possibile creare l'url di connessione SFTP:

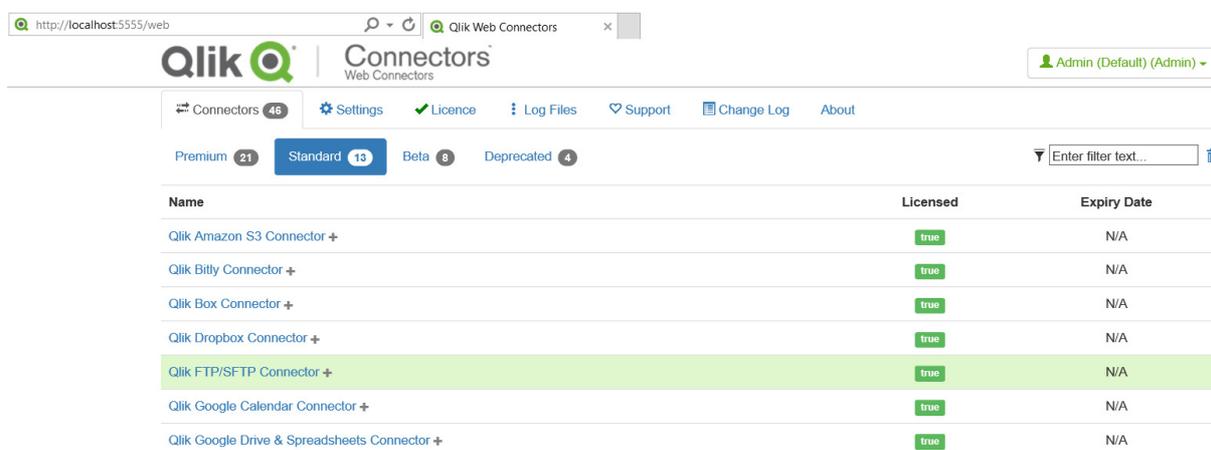


Figura 15. Pagina dei Qlik Web Connectors Standard

Come si evince dall'immagine, i Web Connector sono utilizzati per poter accedere anche ad altre tipologie di risorse Web. Infatti essi consentono di connettersi a diverse origini dati e saranno utilizzati nei paragrafi seguenti anche per recuperare i dati da Google Analytics.

Il connettore FTP/SFTP di Qlik consente di scaricare file o caricarli in FTP/SFTP direttamente dalle applicazioni o dalla directory locale di Qlik Sense. È possibile inoltre scaricare più file da una posizione e disporre di una tabella di file scaricati a scopo di verifica all'interno dell'applicazione.

Le caratteristiche del connettore Qlik FTP / SFTP sono le seguenti [12]:

- Autenticazione con nome utente e password o chiavi pubbliche / private (SFTP).
- Elenco di file e cartelle in una determinata posizione sul server.
- Download di un file denominato dal server FTP / SFTP in una directory locale.
- Download dei contenuti di un file con nome dal tuo server FTP / SFTP direttamente su QlikView o Qlik Sense.
- Caricamento di un file sul tuo server FTP / SFTP.

Il connettore QlikFTP / SFTP ha le seguenti tabelle disponibili[12]:

- FTP/SFTPListFilesAndFolders: elenca i file e le cartelle nella directory specificata del server FTP/SFTP.
- SFTPGetRawFile - Scarica un file con nome dal tuo server SFTP e restituisce direttamente i risultati. È possibile utilizzare l'URL generato come origine del file Web.
- Download FTP/SFTP - Scarica un file con nome, o più file se si utilizza un carattere jolly '*', dal proprio server FTP/SFTP in una cartella locale. La tabella restituita mostra il numero di file scaricati.
- FTP/SFTPDownloadAndList: scarica un file con nome, o più file se si utilizza un carattere jolly '*', dal proprio server FTP/SFTP in una cartella locale. La tabella restituita elenca ogni file scaricato.
- FTP/SFTPUploadFile: carica un file con nome sul server FTP/SFTP da una cartella locale.
- LatestLog: recupera i registri di connessione più recenti che possono essere utili ai fini del debug.

Nel nostro caso, oltre all'estrazione effettiva dei file, è stato necessario studiare un sistema che permettesse di scaricare di volta in volta solo i file nuovi per poter minimizzare i tempi di caricamento.

A tal fine, risulta utile poter estrarre l'elenco dei file presenti nella directory tramite la tabella SFTPListFilesAndFolders, in modo da poter preventivamente capire quali sono i files nuovi da estrarre ad ogni caricamento che viene eseguito.

Uno volta creati gli url che permettono di effettuare la connessione SFTP, basterà inserire l'url generato dal web connector in una connessione di tipo file Web Qlik. Inserita tale connessione all'interno dello script, sarà possibile leggere i file desiderati. Di seguito il processo che è stato strutturato all'interno dello script per la lettura dei dati:

1° estrazione

- 1) Creazione di una connessione che permetta di accedere alla cartella di progetto dove sono stati inseriti i file di storico forniti da Finance
- 2) Estrazione dei dati provenienti dal file di storico e storicizzazione di quest'ultimo in qvd suddivisi per anno-mese di lettura dei documenti da parte dei Readers
- 3) Creazione di un qvd di supporto in cui per ciascun file letto viene indicato se il file è stato estratto; in questo modo, per i caricamenti successivi sarà sufficiente creare un controllo automatizzato che permette di verificare ogni file se è stato estratto precedentemente; il file in questione verrà estratto solo se non risulta tra quelli già estratti in precedenza. Essendo la prima estrazione proveniente da un file di storico fornitoci da Finance, e quindi non presente tra quelli estraibili tramite SFTP, tale qvd di supporto sarà inizialmente vuoto, si andrà popolando man mano che verranno estratti i dati dall'SFTP

Estrazioni successive (con modalità incrementale)

- 1) Estrazione dell'elenco dei file presenti nella directory che rispettano un determinato requisito in termini di naming convention e formato (nel nostro caso il file doveva chiamarsi con il seguente standard rpt2EDS_mmyyyy.txt dove mmyyyy indicano il mese e anno)
- 2) Verifica per ciascun file se era stato precedentemente caricato, grazie al confronto con il qvd precedentemente generato, contenente il nome dei file già caricati:
 - Se il file risulta essere già stato caricato -> viene scartato
 - Se il file risulta non essere mai stato caricato -> viene caricato
- 3) Per ciascun file estratto, i dati vengono storicizzati in file qvd suddivisi per anno-mese di Read Date. Qualora esista già il file per l'anno-mese di destinazione (ad esempio perché erano stati caricati dati per lo stesso anno-mese da altri files), i dati vengono concatenati a quelli già esistenti e storicizzati in unico file per quell'anno-mese.
- 4) Per ciascun file estratto, viene creata una nuova riga nel qvd di supporto che permetterà di sapere, nelle estrazioni successive, che quel file è già stato caricato

L'implementazione di tali logiche è stata effettuata cercando di rendere i processi dello script più generici possibile, rendendo lo stesso codice riutilizzabile anche in altri script, con la possibilità di adattarlo ma mantenendo la struttura portante invariata.

A tal proposito è stato molto utile l'utilizzo di cicli for e subroutine. Queste ultime permettono, all'interno della stessa applicazione, di richiamare in punti diversi una porzione di codice, che viene dunque scritta una volta sola e utilizzata all'interno delle script ogni volta che viene chiamata, tramite l'istruzione `call`. In questo modo, ad esempio, la parte di codice che permette la storicizzazione dei dati è stata scritta una sola volta, ed essa viene richiamata sia nella parte di codice relativa all' estrazione full che in quella relativa all' estrazione incrementale. Inoltre le estrazioni full e incrementale sono state anche esse inserite ciascuna all'interno di una propria subroutine; in questo modo è possibile spegnere/accendere con facilità la porzione di codice desiderata, potendo solo commendare/decommntare la chiamata alla subroutine di di interesse. Questo migliora la manutenibilità nel tempo dello script.

Di seguito viene riportata a titolo dimostrativo la porzione di script creata per la parte di caricamento incrementale:

```
Sub Incremental_Load;
BLOOMBERG_FILE_LIST_temp:
LOAD
    Name,
    Path,
    text(Mid(Name, 9, 6)) as [File Date],
    'Bloomberg' as FileOrigin
FROM [lib://Bloomberg list] (qvx)
where num((Mid(Name, 11, 4)))>=2016;
```

```

If QvdCreateTime('$ (v_Path_Configuration)Execution_Bloomberg.qvd') <> Null()
Then
    Left Join (BLOOMBERG_FILE_LIST_temp)
    Load
        "File Date",
        'Yes' as Flag_reload
    from $(v_Path_Configuration)Execution_Bloomberg.qvd (qvd);

    NoConcatenate
    BLOOMBERG_FILE_LIST:
    load *
    resident BLOOMBERG_FILE_LIST_temp where Flag_reload <> 'Yes';
    drop table BLOOMBERG_FILE_LIST_temp;

Else
    BLOOMBERG_FILE_LIST:
    load *,
    null() as Flag_reload
    resident BLOOMBERG_FILE_LIST_temp;
    drop table BLOOMBERG_FILE_LIST_temp;

end if;

st_Bloomberg:
LOAD * INLINE [Platform,Story ID,Customer #,Customer Name,User Country,User
City,Read Date];

let vsNumRowsFull=0;

//----- ciclo caricamento FTP per ogni file presente nella
cartella, non già caricato precedentemente -----
for t=0 to NoOfRows('BLOOMBERG_FILE_LIST')-1

    Let vsFileName=peek('Name',t,'BLOOMBERG_FILE_LIST');
    let vQwcConnectionBloomberg = 'lib://Bloomberg Connection';

    concatenate (st_Bloomberg)
    Load
        'Bloomberg' as Platform,
        "Story ID",
        date([Read Date],'YYYYMM') as [YearMonth Store],
        "Customer #",

```

```

        "Customer Name",
        "User Country",
        "User City",
        "Read Date"
FROM [$(vQwcConnectionBloomberg)]
(URL IS
[http://localhost:5555/data?connectorID=FileTransferConnector&table=SFTPGet
RawFile&host=sftp.bloomberg.com&port=xxxxxx(vsFileName)&appID=], txt, utf8,
embedded labels, delimiter is '|', msq)

where not IsNum("Story ID") and "Story ID"<>'UNKNOWN';
    let vsFileDate=peek('File Date',t,'BLOOMBERG_FILE_LIST');

    letvsNumRows=Num(NoOfRows('st_Bloomberg'))-num('$ (vsNumRowsFull) ');

    let vsNumRowsFull=Num(NoOfRows('st_Bloomberg'));

Metadata:
Load
'BLOOMBERG' as Table,
Now() as ExecDate,
'$ (vsFileDate)' as [File Date],
'$ (vsNumRows)' as Rows
AutoGenerate 1;

Next t;

If $(vsNumRowsFull)<>0 then
    If QvdCreateTime('$ (v_Path_Configuration)Execution_Bloomberg.qvd')<>
Null() Then
        Concatenate(Metadata)
        Metadata:
        Load * From '$ (v_Path_Configuration)Execution_Bloomberg.qvd'
(qvd);

    End if;

Store Metadata into '$ (v_Path_Configuration)Execution_Bloomberg.qvd';
Drop Table Metadata;

call store;

```

```
end if;

drop tables st_Bloomberg, BLOOMBERG_FILE_LIST;

end Sub
```

Lo script di storicizzazione incrementale, richiamato dalla funzione call con attributo “store” è il seguente:

```
Sub Store;

Period:
load distinct
[YearMonth Store]
resident st_Bloomberg;

    for v = 0 to NoOfRows('Period')-1

        Let vsPeriod = peek('YearMonth Store',v,'Period');

        NoConcatenate
        st_Bloomberg_YearMonth:
        load *
        resident st_Bloomberg
        where [YearMonth Store]='$(vsPeriod)';

        store st_Bloomberg_YearMonth into
        $(v_Path_QVD_DwStaging)Bloomberg/st_BLOOMBERG_$(vsPeriod).qvd;
        drop table st_Bloomberg_YearMonth;

    Next v;

Drop table Period;

End Sub;
```

3.5.2.2. ETL1 – CAPIQ

L' ETL1 – CAPIQ è stato creato per poter estrarre i dati delle letture provenienti dalla piattaforma CAPIQ. In questo caso i dati da estrarre sono contenuti in dei file xls che vengono depositati in una share directory sul server Qlik Sense. Basterà dunque in questo caso creare una connessione che consenta di accedere al percorso dove sono presenti tali file.

Di seguito il processo che è stato strutturato all'interno dello script per la lettura di tali dati:

1° estrazione

- 1) Estrazione dell'elenco dei file presenti nella directory che rispettano un determinato requisito in termini di naming convention e formato xls.
- 2) Estrazione dei dati provenienti da ciascun file e storicizzazione di questi ultimi in qvd suddivisi per anno-mese di lettura dei documenti da parte dei Readers
- 3) Creazione di un qvd di supporto in cui per ciascun file letto viene indicato se il file è stato estratto; in questo modo, per i caricamenti successivi sarà sufficiente creare un controllo automatizzato che permette di verificare ogni file se è stato estratto precedentemente; il file in questione verrà estratto solo se non risulta tra quelli già estratti in precedenza

Estrazioni successive (con modalità incrementale)

- 5) Estrazione dell'elenco dei file presenti all'interno della share directory
- 6) Verifica per ciascun file se era stato precedentemente caricato, grazie al confronto con il qvd precedentemente generato, contenente il nome dei file già caricati.
 - Se il file risulta essere già stato caricato -> viene scartato
 - Se il file risulta non essere mai stato caricato -> viene caricato
- 7) Per ciascun file estratto, i dati vengono storicizzati in file qvd suddivisi per anno-mese di Read Date. Qualora esista già il file per l'anno-mese di destinazione (ad esempio perché erano stati caricati dati per lo stesso anno-mese da altri files), i dati vengono concatenati a quelli già esistenti e storicizzati in unico file per quell'anno-mese.
- 8) Per ciascun file estratto, viene creata una nuova riga nel qvd di supporto che permetterà di sapere, nelle estrazioni successive, che quel file è già stato caricato

E' utile notare che la logica adoperata è del tutto analoga a quella utilizzata per estrarre i dati da Bloomberg; infatti anche se la tipologia di connessione è differente, l'utilizzo di una logica di caricamento e storicizzazione del dato simile, permette una più semplice manutenibilità degli stessi ETL nel tempo, in quanto eventuali modifiche possono essere gestite in modo simile anche negli altri ETL.

3.5.2.3. ETL1 – FactSet

L' ETL1 – FactSet permette di estrarre i dati delle letture provenienti dalla piattaforma FactSet tramite protocollo di rete SFTP. Esso funziona in modo analogo all' ETL1 – Bloomberg descritto nel paragrafo 3.5.2.1, per tale motivo si rimanda a tale paragrafo la spiegazione del processo di estrazione e storicizzazione dei dati, sia a livello full che incrementale.

3.5.2.4. ETL1 – Research Tree

L' ETL1 – Research Tree permette di estrarre i dati delle letture provenienti dalla piattaforma Research Tree tramite l'utilizzo di apposite API.

L'API utilizzata necessita di 5 parametri per l'estrazione dei dati, che devono essere inseriti all'interno dell'url dell'API stessa:

- TokenID: identificativo
- Startdate: data di inizio lettura dei documenti da cui si vuol iniziare ad estrarre i dati
- Enddate: data di fine lettura dei documenti per l'estrazione dei dati
- Startrow: n° riga da cui si vogliono iniziare a leggere i dati
- Pagelength: n° di righe da leggere

Essendo necessario per noi dover creare un processo automatizzato, che possa ogni giorno caricare solo il range temporale di interesse ai fini di aggiornare lo storico già esistente in modo incrementale, si è deciso di rendere variabili i parametri da inserire. La stringa di connessione all'API in questo modo diventa:

```
WITH CONNECTION (URL "https://research-tree-  
api.com/api/DownloadHistoryByProvider?token=$(vsToken)&startdate=$(vStartDate)  
&enddate=$(vEndDate)&startrow=$(vsRowStartPage)&pagelength=$(vSkipPage) "  
)
```

Con le seguenti 3 variabili settate in modo statico:

```
LET vStartDate = date((Today() - 1), 'YYYY-MM-DD');  
LET vEndDate = Date(Today(), 'YYYY-MM-DD');  
LET vsToken = '1e232eb1-602b-472a-82f8-xxxxxxxxxx';
```

In questo modo ogni giorno vengono estratti i dati del giorno precedente e del giorno in corso, senza dover modificare lo script di caricamento.

Anche in questo caso, nella prima estrazione è stato necessario effettuare un recupero dello storico, pertanto in tale contesto la variabile vStartDate è stata settata al 2016-01-01, per permetterci di storicizzare lo storico di 3 anni + 1 rolling.

Per quanto riguarda i parametri Startrow e Pagelength è stato necessario studiare un processo che possa in automatico settarne correttamente il valore per ciascuna estrazione.

Queste due variabili sono strettamente legate tra loro; la variabile Pagelength limita il numero di righe che possono essere lette per ciascuna chiamata all'API, mentre Startrow permette di iniziare a leggere i dati non necessariamente dalla prima riga, ma da qualunque riga della

tabella. In questo modo, se ad esempio il numero massimo di righe che è possibile leggere per ciascuna chiamata è 100, e la tabella da leggere ha 200 righe, basterà fare 2 chiamate, una settando Startrow=0 e Pagelength=100, mentre l'altra con Startrow=100 e Pagelength=100. Al primo giro verranno dunque lette le prime cento righe, al secondo giro le altre 100.

È possibile sapere qual è il valore massimo di Pagelength da poter settare grazie ad un'altra API che per Token, Startdate e Enddate ci restituisce il numero massimo di righe da leggere.

In particolare la connessione:

```
WITH CONNECTION (URL "https://research-tree-api.com/api/DownloadHistoryByProvider?token=$(vsToken)&startdate=$(vStartDate)&enddate=$(vEndDate)");
```

restituisce degli attributi relativi alla grandezza della tabella che si desidera leggere, come il numero di righe che essa contiene e il campo "CurrentPageLength" che indica qual è il numero massimo di righe che è possibile estrarre ad ogni chiamata.

E' stato pertanto necessario creare un processo ciclico che calcoli per ciascuna chiamata qual è il numero di riga da cui iniziare a leggere, in modo da riuscire ad estrarre in più chiamate il set di dati completo.

Di seguito vengono riassunte le logiche di caricamento e storicizzazione dei dati:

1° estrazione full di storico

- 1) Settaggio della variabili vStartDate al 1° gennaio 2019, vEndDate al giorno attuale e vsToken
- 2) Estrazione dell' attributo CurrentPageLength e settaggio della variabile vSkipPage uguale al valore assunto da tale attributo
- 3) Esecuzione di un ciclo Do while che estrae per ogni chiamata un numero di righe uguale alla variabile vSkipPage, con primo settaggio della variabile vsRowStartPage=0, che aumenta il suo valore di vSkipPage ad ogni chiamata. Il ciclo si conclude appena sono state estratte tutte le righe della tabella.
- 4) Storicizzazione dei dati estratti in qvd suddivisi per anno-mese di lettura dei documenti da parte dei Readers

Estrazioni successive (modalità incrementale con DayBack=1)

- 1) Settaggio della variabili vStartDate al giorno precedente quello di estrazione, vEndDate al giorno attuale e vsToken
- 2) Estrazione dell' attributo CurrentPageLength e settaggio della variabile vSkipPage uguale al valore assunto da tale attributo
- 3) Esecuzione di un ciclo Do while che estrae per ogni chiamata un numero di righe uguale alla variabile vSkipPage, con primo settaggio della variabile vsRowStartPage=0, che aumenta il suo valore di vSkipPage ad ogni chiamata. Il ciclo si conclude appena sono state estratte tutte le righe della tabella.
- 4) Caricamento dei qvd di storico degli anno-mese relativi agli ultimi 2 giorni. Questi dati vengono concatenati a quelli estratti dalle API, ma vengono prima eliminati dallo storico tutti i record con data di lettura superiore agli ultimi 2 giorni. In questo modo ad ogni estrazione si rinnova lo storico degli ultimi 2 giorni. Da qui la dicitura DayBack=1, che si differenzia dal normale incrementale che aggiunge solo i record nuovi; in questo caso i record degli ultimi 2 giorni dalla data di caricamento vengono

ricaricati totalmente, previa eliminazione di quelli che erano già stati salvati precedentemente.

3.5.2.5. ETL1 – Thomson Reuters

L' ETL1 – Thomson Reuters permette di estrarre i dati delle letture provenienti dalla piattaforma Thomson Reuters tramite protocollo di rete FTP. Esso funziona in modo analogo all' ETL1 – Bloomberg descritto nel paragrafo 3.5.2.1, anche se in quel caso si trattava di protocollo SFTP; il processo di utilizzo dei web connectors e di sviluppo dello script è però analogo, dunque si rimanda a tale paragrafo la spiegazione del processo di estrazione e storicizzazione dei dati, sia a livello full che incrementale

3.5.2.6. ETL1 – Google Analytics

L'ETL1 – Google Analytics è stato costruito al fine di estrarre i dati delle letture provenienti dal sito di Finance. Tali dati vengono estratti da Google Analytics, tramite appositi Web Connectors.

Google Analytics è un servizio di Google servizio a cui si appoggia Finance per monitorare il suo sito web. Tale strumento elabora delle analisi relative agli accessi degli utenti. In questo modo è possibile monitorare il numero di sessioni, di utenti effettivi e altri indicatori come il tempo medio di visita di una certa pagina, mantenendo l'anonimato degli utenti ma fornendo dettagli su alcune caratteristiche di questi ultimi, come l'area geografica di appartenenza [13].

Il web Connector Qlik di Google Analytics permette, una volta settati i parametri di accesso, di estrarre i dati da una tabella customizzata in base alle proprie esigenze. E' infatti possibile estrarre i dati scegliendo le dimensioni e misure che deve avere la tabella, tra quelle messe a disposizione dal servizio.

Anche in questo caso il Web connector genera un url da inserire all'interno dello script per consentire l'estrazione dei dati. Nell' url di connessione è richiesto l'inserimento del TimePeriod, cioè l'intervallo di inizio e fine estrazione dei dati, espresso in DDMMYYYY.

Al fine di estrarre di volta in volta solo i dati nuovi, il processo di caricamento è stato gestito in maniera incrementale, con un DayBack=6. Tale DayBack è stato inserito per avere maggiore sicurezza circa il fatto che eventuali rielaborazioni ai dati più recenti da parte del servizio, seppur improbabili, non vengano perse.

Inoltre è risultato necessario effettuare un primo doppio caricamento full dello storico; il sito Finance era stato cambiato di recente, pertanto la maggior parte delle informazioni di storico, provenienti dal vecchio sito web, non erano presenti su Google Analytics. Pertanto ci è stato consegnato un file csv da caricare contenente le informazioni del vecchio sito, che abbiamo concatenato a quelle nuove estratte da Google Analytics.

Di seguito i processi relativi alle varie estrazioni dello script.

1° estrazione full dello storico del vecchio sitoweb

- 1) Creazione connessione alla cartella di progetto contenente il file di storico
- 2) Estrazione e storicizzazione dei dati in qvd suddivisi per anno-mese di lettura dei documenti da parte dei Readers

1° estrazione full dello storico dell' attuale sitoweb da Google Analytics

- 1) Settaggio delle variabili di data inizio e fine estrazione (full a partire dal primo giorno di presenza del nuovo sitoweb).
- 2) Estrazione e storicizzazione dei dati in qvd suddivisi per anno-mese di lettura dei documenti da parte dei Readers

Estrazioni successive (modalità incrementale con DayBack=6)

- 1) Settaggio della variabili vStartDate=Today()-6 e vEndDate al giorno attuale
- 2) Estrazione dei dati e caricamento dei qvd di storico degli anno-mese relativi agli ultimi 7 giorni. Questi dati vengono concatenati a quelli dell'ultima estrazione, ma vengono prima eliminati dallo storico tutti i record con data di lettura superiore agli ultimi 7 giorni. In questo modo ad ogni estrazione si rinnova lo storico degli ultimi 7 giorni. Storicizzazione dei dati in qvd suddivisi per anno-mese di lettura dei documenti da parte dei Readers.

3.5.2.7. ETL1 – Sales Force

L'ETL1 Sales Force permette di estrarre i dati di anagrafica dei readers presenti su Sales Force tramite apposito connettore, in cui vengono inseriti i parametri di accesso necessari.

Le tabelle da cui estrarre i dati sono due, una relativa ai dati del singolo reader (tabella contact) e l'altra relativa alle aziende di appartenenza dei readers (tabella account) ogni reader appartiene ad una sola azienda, dunque le due tabelle connesse logicamente tra di loro tramite un ID.

Trattandosi di tabelle di anagrafica, le righe di queste possono essere aggiornate continuamente, dunque la scelta migliore è quella di ricaricarle integralmente ogni volta che viene fatto un reload, e storicizzarle in un unico qvd per tabella (in quanto non presente nessuna data di lettura), che sovrascrive quello dell'estrazione precedente.

3.5.2.8. ETL1 – xml

L'ETL1 – XML è stato creato per estrarre le informazioni di dettaglio relative ai documenti oggetto di lettura da parte dei Readers. Tali informazioni vengono estratte da degli xml, che contengono il codice sorgente di visualizzazione di tali documenti nelle pagine web. Una volta stabiliti quali parametri estrarre dagli xml al fine di recepire le informazioni utili alla nostra applicazione, è stato sufficiente estrarre le informazioni e storicizzarle in apposite tabelle, seguendo la suddivisione strutturale degli xml stessi.

Gli xml sono depositati in un'apposita share directory sul server QlikSense, raggiungibile previa creazione della connessione al path; a seguito dell'estrazione i dati vengono pertanto storicizzati in delle tabelle uniche, senza alcuna suddivisione per data. Ad ogni estrazione è stato deciso di ricaricare tutti i dati e sovrascriverli a quelli vecchi, in modo da lasciare la possibilità di effettuare modifiche e aggiornamenti ad xml già esistenti.

3.5.2.9. ETL1 - Dotmailer

L'ELT1 – Dotmailer è stato creato al fine di estrarre ed elaborare i dati degli users a cui Finance invia le e-mail per promuovere i documenti pubblicati. Ciascuna e-mail, relativa alla pubblicazione di un determinato documento, viene inviata ad un gruppo di user; tale e-mail è chiamata campagna, in quanto è ricevuta da più users. Per estrarre i dati relativi alle campagne abbiamo utilizzato delle API con architettura REST. In questo modo, una volta creata la connessione REST previo inserimento dei parametri necessari di accesso, è stato possibile estrarre i dati, utilizzando i 3 seguenti metodi GET:

- **GET ALL CAMPAIGN:** contiene l'elenco delle campagne inviate, con il CMS associato, che permette di ricondurre al codice del documento

`https://r1-api.dotmailer.com/v2/campaigns?select=1000&skip=$(numSkip)`

- **GET CAMPAIGN OPEN:** contiene per ciascuna campagna l'elenco degli users che hanno aperto l'e-mail

`https://r1-api.dotmailer.com/v2/campaigns/$(Ext)/opens?select=1000&skip=$(numSkip)`

- **GET CAMPAIGN CLICK:** contiene per ciascuna campagna l'elenco degli users che hanno cliccato all'interno dell'e-mail sull'url che rimanda al documento pubblicato

`https://r1-api.dotmailer.com/v2/campaigns/$(Ext)/clicks?select=1000&skip=$(numSkip)`

Come nel caso dell'ETL1 – Research Tree, anche qui è stato necessario variabilizzare alcuni parametri, in particolare:

- **\$(Ext)** -> variabile contenente il codice della campagna. I metodi "Get Campaign Open" e "Get Campaign Click" permettono di estrarre i dati relativi alle mail aperte e cliccate per singola campagna. Il codice della campagna deve essere pertanto specificato all'interno dell'url.
- **\$(numSkip)** -> variabile che permette di stabilire da quale numero di riga leggere i dati. Analogamente a quanto accaduto per l'ETL1 Research Tree, non è possibile estrarre l'interno set di valori in una unica estrazione. In questo caso il numero massimo di righe estraibili per ciascuna chiamata è 1000, dunque è stato necessario impostare un ciclo che estragga i dati 1000 righe per volta.

Di seguito i passaggi logici adoperati all'interno dello script:

- Estrazione “Get All Campaign”: estrazione di tutte le campagne attive e i codici CMS associati. Storicizzazione della tabella in sovrascrittura a quella dei caricamenti precedenti.
- Estrazione “Get Campaign Open”: creazione di due cicli annidati, che permettono di estrarre i dati di apertura delle mail per ciascuna campagna attiva estratta dal metodo “Get All Campaign”, e di estrarre i dati 1000 righe alla volta. Storicizzazione della tabella in sovrascrittura a quella dei caricamenti precedenti. Di seguito lo script contenente il doppio ciclo di caricamento annidato:

```
LET NumRows=NoOfRows('GET_ALL_CAMPAIGNS');
FOR i=1 to $(NumRows)
    LET Ext=FieldValue('id_campaign',$i);
    let a=1000;
    let numSkip=1;
    Do while a=1000
        RestConnectorMasterTable:
        SQL SELECT
            "contactId",
            "email",
            "mailClient",
            "dateOpened"
        FROM JSON (wrap on) "root"
        WITH CONNECTION (
            URL"https://r1-
            api.dotmailer.com/v2/campaigns/$(Ext)/opens?select=1000&skip=$(numSkip)");
        Let a=NoOfRows('RestConnectorMasterTable');
        let numSkip= numSkip+1000;
        DROP TABLE RestConnectorMasterTable;
    Loop
next;
```

- Estrazione “Get Campaign Click”: creazione di due cicli annidati, che permettono di estrarre i dati di click dell'url nella mail per ciascuna campagna attiva estratta dal metodo “Get All Campaign”, e di estrarre i dati 1000 righe alla volta. La logica dello script è analoga a quella utilizzata per l'estrazione dei dati dal metodo “Get Campaign Open”. Storicizzazione della tabella in sovrascrittura a quella dei caricamenti precedenti.

È opportuno notare che il processo di estrazione non prevede in questo caso dei caricamenti e storicizzazioni incrementali, ma ogni volta prevedere l'estrazione e storicizzazione dell'intero

set di dati. Questo è dovuto al fatto che, per ciascun metodo, non è possibile settare sull'url un range temporale per cui estrarre i dati, dunque è necessario estrarre tutto il set ad ogni chiamata. Il ciclo viene fatto infatti su parametri come il codice della campagna, che è indipendente da un range temporale, ed è in continuo aggiornamento, dunque necessità di una costante estrazione (ad esempio la mail di una campagna può essere aperta anche a distanza di anni da parte degli users).

3.5.3. ETL2 – Costruzione del data model

L'ETL2 è stato creato al fine di poter elaborare i dati estratti e storicizzati dai precedenti ETL, tramite la costruzione e storicizzazione di tabelle contenenti i dati trasformati e normalizzati secondo le logiche necessarie per poter ottenere l'output desiderato sull'interfaccia utente.

Le nuove tabelle così elaborate, vengono poi storicizzate in qvd di secondo livello, a cui attingerà l'app di presentation per la costruzione del data model. A differenza degli ETL1, in cui le estrazioni e le storicizzazioni sono incrementali, l'ETL2 effettua un'estrazione ed elaborazione totale dei qvd di primo livello, pertanto la storicizzazione dei dati elaborati sarà anch'essa "full": per ogni caricamento i qvd di secondo livello vengono sovrascritti a quelli pre esistenti dal caricamento precedente. Questa tecnica è consigliata in quanto eventuali modifiche di logiche hanno effetto in automatico su tutti i dati, riducendo eventuali problemi dovuti a differenti trasformazioni dei dati nel tempo: i dati in questo modo vengono sempre trasformati con le stesse logiche, indipendentemente dell'età di estrazione del dato stesso.

Lo scopo dell'ETL2 è quello di creare delle tabelle che possono essere legate tra loro in modo relazionale, tramite campi chiave; queste tabelle verranno poi caricate sull'app di presentation, permettendo così la creazione del data model. Al fine di rendere il data model correttamente funzionante, nella creazione di tali tabelle è necessario tenere a mente le seguenti linee guida:

- Cercare di collegare ciascuna tabella ad un'altra tramite un solo campo chiave; è sconsigliabile pertanto la presenza di chiavi sintetiche. Quando due o più tabelle di dati hanno due o più campi in comune, questo indica una relazione di chiavi composite. Qlik Sense gestisce questa condizione mediante la creazione automatica di chiavi sintetiche. Queste chiavi sono campi anonimi che rappresentano tutte le combinazioni ricorrenti della chiave composta [17].

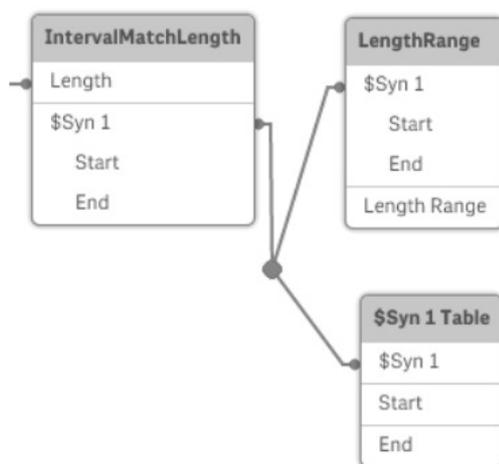


Figura 16. Esempio di chiave sintetica tra tabelle [17]

Qualora risulti necessario collegare due tabelle in base a più attributi, è consigliabile creare un unico campo chiave composto, contenente all'interno tutti gli attributi necessari.

- Non utilizzare relazioni circolari tra tabelle: se in una struttura di dati sono presenti riferimenti circolari, le tabelle vengono associate in modo che esistano più percorsi di associazioni tra due campi. È consigliabile evitare per quanto possibile questo tipo di struttura di dati, perché potrebbe comportare ambiguità di interpretazione degli stessi. Se Qlik Sense rileva strutture di dati circolari durante l'esecuzione dello script di caricamento, viene visualizzata una finestra di dialogo di avviso e una o più tabelle vengono impostate come logicamente disconnesse. Nel sistema di visualizzazione modello dati le tabelle logicamente disconnesse sono indicate da collegamenti rossi punteggiati verso altre tabelle [18], come in figura:

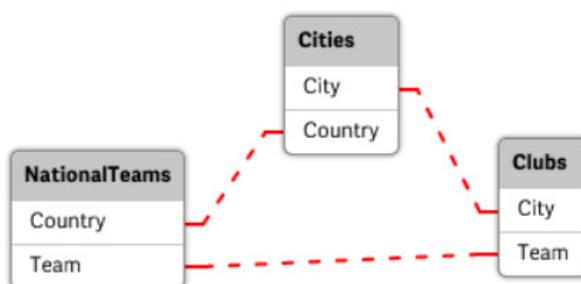


Figura 17. Esempio riferimento circolare tra tabelle [18]

Nel caso in cui vengano generati riferimenti circolari, occorre modificare lo script di caricamento dei dati assegnando un nome univoco a uno dei campi con nomi identici [18].

Partendo dall'idea di data model precedentemente individuato (rif. paragrafo 3.5.1), si procede con il caricamento e la trasformazione dei dati, suddivisi in tabelle. Lo script di caricamento, è stato suddiviso in 10 tab, ciascuno dedicato a delle specifiche trasformazioni:

- *Main*: tab presente in tutte le applicazioni Qlik, dove sono settate le variabili di default di sistema, come il formato data e ora, i divisori e gli abbreviatori numerici, le descrizioni dei giorni e mesi. Il settaggio che è stato utilizzato è quello inglese.

```
SET CollationLocale='en-UK'
```

- *Config*: tab in cui è stato caricato un file excel di configurazione contenente tutti i path di progetto dove sono presenti i dati da caricare e storicizzare. Ciascun path è associato ad una variabile; in questo modo ogni volta che si desidera utilizzare un path sarà sufficiente scrivere il nome della variabile associata. Questo metodo è comodo in quanto centralizza eventuali modifiche ai path all'interno del file, senza dover fare modifiche multiple sullo script, in quanto il nome della variabile può restare invariato.
- *Xml*: tab che esegue il caricamento e le trasformazioni relative ai dati dei documenti storicizzati nei qvd1 dall' ETL1 – xml. Vengono dapprima caricati i qvd di primo livello relativi ai documenti; successivamente tali tabelle, che hanno in comune l'attributo relativo al nome del file, vengono messe in join tra loro, al fine di tenere tutte le informazioni in un'unica tabella:

- File Name: nome del report
- Document ID xml: codice del documento.
- CMS: codice identificativo del documento, ogni documento ne possiede solo uno
- Doc Publication Date: data di pubblicazione del documento
- Document Title: titolo del documento
- URL: url di collegamento al documento pubblicato sul web
- Account Name: nome dell'azienda oggetto del documento
- Account ID: codice identificativo dell'azienda oggetto del documento
- Account Sector Code: codice identificativo del settore dell'azienda
- Account Sector Name: descrizione del settore dell'azienda

Creazione di una tabella temporanea di mapping, contenente per ogni CMS il document ID principale associato. Tale tabella sarà utilizzata per aggiungere nelle tabelle contenenti i dati delle letture per CMS, il Document ID principale associato. I dati provenienti dagli xml sono uno a uno tra Document ID e CMS. Tramite questa tabella di mapping è quindi possibile collegare i dati delle letture identificate per CMS con i dati anagrafici dei documenti.

- *Mapping CMS*: tab che carica un file di appoggio contenente per ogni CMS, tutti i possibili Document ID ad esso associati. Durante le fasi di progetto ci si è accorti che, nei dati delle letture provenienti dalle piattaforme, i document ID presenti non erano solo quelli presenti nei file xml, ma erano stati attribuiti anche altri codici; non essendo presenti in tali dati di lettura i codici CMS, è risultato necessario creare un file di mappatura in cui per ogni nuovo document ID veniva associato il CMS di riferimento; in questo modo è possibile collegare le letture alle anagrafiche dei documenti. Una volta caricato tale file, viene creata una tabella temporanea contenente per ogni document ID il CMS associato, che permetterà di effettuare tale collegamento.

- *Dotmailer*: tab in cui vengono caricati i qvd di primo livello relativi ai dati delle mail aperte e cliccate estratti da Dotmailer. I dati delle mail aperte vengono concatenati in un'unica tabella insieme ai dati delle email cliccate, previa normalizzazione dei dati, negli stessi nomi attribuiti, distinguendo la tipologia di dato tramite un campo che assume il valore "open" o "read" in base al fatto che sia una riga relativa a email aperte o cliccate. Inoltre il CMS viene sostituito dal DocumentID principale tramite funzione applymap riferita alla tabella di mapping creata nella sezione xml. I campi così individuati sono i seguenti:
 - ID campaign: ID della campagna di email. Ogni campagna viene inviata a più destinatari
 - Email: indirizzo email del destinatario della mail
 - Date: data di apertura o click sull'url della mail
 - Flag Type Date: Individua se si tratta di dati su mail aperte tramite la valorizzazione "Open" o cliccate tramite la valorizzazione "click"

- *Mapping Geo*: tab che carica una serie di tabelle di mappatura geografica. Tali tabelle sono in parte provenienti da dei qvd messi a disposizione dalla libreria Qlik, in cui per ogni Stato – città vengono date le coordinate geografiche della città (a livello di punto) e dello stato (a livello di area). A queste tabelle abbiamo provveduto ad aggiungere un file in cui per ogni stato viene definita la capitale e il continente di appartenenza. In questo modo, eventuali buchi nei dati, dovuti ad esempio a città non trovate nei qvd della libreria, saranno facilmente sostituibili tramite funzione di applymap con la capitale dello stato di appartenenza, evitando mancanze di dati di lettura nella mappa geografica che sarà inserita all'interno dell'interfaccia. Inoltre associando le country ai continenti, avremo un ulteriore livello di aggregazione da utilizzare all'interno dei grafici Qlik precedentemente condivisi nel mockup.

- *Platforms temp*: tab che carica lo storico dei dati delle piattaforme dei qvd di primo livello e li concatena in un'unica tabella, previa normalizzazione dei dati al fine di ottenere uno standard di struttura tabellare come segue, in cui ogni riga corrisponde ad una lettura:

- key_Plat_Reader: chiave composta dal Reader ID e il nome della piattaforma da cui proviene la lettura. Tale chiave servirà per agganciare alla tabella i dati di anagrafica provenienti da Sales Force
- Platform: nome della piattaforma di provenienza della lettura
- Read Date: data di lettura
- Reader's ID: identificativo del reader. Uno stesso lettore può avere ID diversi sulle varie piattaforme; in alcune piattaforme i lettori sono anonimi, in quel caso viene a loro attribuito il valore “unknown”
- Reader's ID counting: identificativo che esclude i lettori sconosciuti attribuendo loro il valore nullo. Serve per il calcolo del KPI dei lettori distinti, per i quali è necessario escludere i lettori anonimi, in quanto non possediamo un parametro per identificarli univocamente.
- Reader's Institution Platform: azienda di provenienza del Reader
- City Platform: città di provenienza del Reader
- Country Platform: stato di provenienza del Reader
- Reader's name Platform: nome del Reader (dove presente)
- Document ID: ID del documento di riferimento della lettura
- Platform email: email del Reader

Al fine di rendere più standard possibili i valori che possono essere assunti dagli attributi, soprattutto quelli che consentono di effettuare delle aggregazioni sui dati (come la data di lettura), sono state eseguite le seguenti trasformazioni sui dati:

- Aggiunta dell'attributo platform che identifica la piattaforma di appartenenza del dato di lettura
 - Standardizzazione del formato data: ogni sorgente possiede un diverso formato data, alcune il time stamp, che sono state riportate ad uno stesso formato di tipo “DD/MM/YYYY”. Tale standardizzazione è necessaria al fine di poter effettuare in modo corretto ed efficace filtri sulla data di lettura; ciò permette inoltre di poter creare una tabella di calendario di appoggio in cui sono presenti livelli aggregati di date, come la settimana o il mese, tramite cui sarà possibile filtrare gruppi di date appartenenti alla stessa categoria.
 - Normalizzazione dei dati incompleti, che spesso assumevano diversi valori come “***”, “anonymus”, “NA” ecc. tali casistiche individuate sono state tutte rivalorizzate sotto l'unica voce "Unknown"
 - Standardizzazione della country, che in base alla piattaforma assumeva tipi valori diversi; alcune piattaforme inserivano l'ISO code, mentre altre il descrittivo della città. Tramite la funzione applymap riferita alle tabelle di mapping elaborate nel tab Mapping Geo è stato possibile ricondurre i dati sempre al descrittivo della Città, in quanto tali qvd presentano anche per ogni Stato l'ISO code di appartenenza (sia a 2 cifre che a 3).
- *Sales Force*: tab che carica i dati anagrafici dei lettori provenienti dai qvd1 generati dall' ETL1 – Sales Force e li aggiunge a quelli della tabella creata nel tab Platform temp, tramite la chiave key_Plat_Reader, composta dal Reader ID e il nome della

piattaforma da cui proviene la lettura. Dato che uno stesso reader, in base alla piattaforma può avere differenti ID, su Sales Force la tabella contact è strutturata come nell'esempio seguente:

Reader Name	Sales Force	CAPIQ	Thomson Reuters	Research Tree	email	City	Country
Antonio Rossi	AAAA	BBBB	CCCC	DDDD	antonio.rossi@example.it	Turin	Italy

Per poter attaccare le informazioni di anagrafica sulla tabella Platform, che è strutturata per singola lettura su piattaforma, come nell'esempio:

Reader's ID	Platform	Read Date
AAAA	Fact Set	09/10/2019
BBBB	CAPIQ	05/09/2019
CCCC	Thomson Reuters	12/12/2018
DDDD	Research Tree	23/06/2019
antonio.rossi@example.it	Dotmailer	05/11/2016

Occorre modificare la struttura tabellare dei dati di Sales Force tramite la funzione cross table, che permette di trasformare una tabella con diverse colonne in una tabella in cui le intestazioni delle colonne vengono inserite in una colonna, mentre i loro dati saranno memorizzati in un secondo attributo, il campo dati. In questo modo la tabella di anagrafica diventa:

Reader Name	Reader's ID	Platform	City	Country
Antonio Rossi	AAAA	Fact Set	Turin	Italy
Antonio Rossi	BBBB	CAPIQ	Turin	Italy
Antonio Rossi	CCCC	Thomson Reuters	Turin	Italy
Antonio Rossi	DDDD	Research Tree	Turin	Italy
Antonio Rossi	antonio.rossi@example.it	Dotmailer	Turin	Italy

Risulta possibile a questo punto agganciare per chiave Reader's ID e Platform i dati di anagrafica alla tabella Platform.

Prima di effettuare tale join tra le due tabelle vengono fatte altre trasformazioni:

- la tabella di anagrafica reader viene unita ai dati di anagrafica relativi all'azienda di provenienza del reader, tramite un campo ID chiave di collegamento tra le due tabelle.
- Ad ogni City e Country vengono associate le coordinate puntuali della City, le coordinate dell'area della Country e il nome del continente di appartenenza.

Tali associazioni vengono effettuate utilizzando i dati elaborati nel tab “Mapping Geo” tramite funzioni annidate di applymap che permettono di gestire varie trasformazioni dei dati necessarie per identificare le coordinate geografiche. Durante tale processo, al fine di limitare il numero di buchi presenti all’interno delle coordinate geografiche, dovuti a nomi di city e country non trovati all’interno del qvd della libreria Qlik, vengono gestite le seguenti eccezioni:

- Se la città non viene trovata oppure è sconosciuta, ma è presente lo Stato, viene utilizzata come città la capitale dello Stato.
- Se lo stato non viene trovato oppure è sconosciuto, ma è presente la città, viene inserito come stato quello relativo alla città. Qualora il nome della città appartenga a più stati contemporaneamente nella libreria Qlik, viene preso il primo stato caricato dal qvd.
- Unificazione dei nomi degli stati. Se per esempio alcuni record sono stati trascritti come Italy e altri come Italia, i campi valorizzati in “Italia” vengono sovriscritti come “Italy”. Questo passaggio è molto importante al fine di permettere all’utente, quando applica un filtro geografico, di prendere esattamente i dati di appartenenza senza possibilità di avere delle ambiguità.

Infine un’altra eccezione che viene gestita è relativa all’aggancio con i dati provenienti dalla piattaforma Bloomberg. Per tale piattaforma infatti i dati di lettura sono su utenti anonimi di cui possediamo però il nome dell’azienda di appartenenza. Per tale piattaforma pertanto vengono agganciati solo i dati anagrafici relativi all’azienda, mentre i dati relativi al reader’s vengono compilati come “Unknown”.

I dati aggiunti a quelli già presenti sulla tabella Platform sono:

- Sector: settore dell’azienda del reader
- Research Sub Sectors: sotto settore dell’azienda del reader
- Current Market Cap US\$ (millions): Capitale dell’azienda del reader
- Investor Type contact: Tipo di investitore a livello di reader
- Investor Type account: tipo di investitore a livello aziendale
- Reader First Name: nome del reader
- Reader Last Name: cognome del reader
- Email: email del reader
- Institution Name: nome dell’azienda del reader
- Reader City: città del reader
- Reader Country: country del reader
- Geo.country_GeoInfo: coordinante dell’area geografica della country
- Geo.city_GeoInfo: coordinate della city
- Continent: continente del reader

Google Analytics: in questo tab vengono caricati i dati dei qvd di primo livello relativi alle letture sul sito web provenienti da Google Analytics. Essendo i dati di lettura di Google Analytics anonimi, non è possibile agganciare alcun dato anagrafico sui readers proveniente da Sales Force, ma saranno utilizzati come campi di anagrafica solo quelli presi da Google Analytics stesso. I dati provenienti da Google Analytics non sono per singolo utente, ma per numero di lettori aventi le stesse caratteristiche (Città, Stato, data di lettura, documento letto). Al fine di standardizzare il dato e concatenarlo alla tabella contenente le letture delle altre piattaforme è stato opportuno spezzare i dati per ogni singola di lettura, tramite un ciclo while che per ciascuna riga crea un numero di righe uguale al numero di user individuati, tutti con le stesse caratteristiche, assegnando per ciascuna riga il reader ID = “Unknown”. In questo modo è stato possibile concatenare i dati alla tabella platform, assegnando il valore “Website” al campo platform. Anche nel caso dei dati di Google Analytics è stato necessario effettuare le seguenti manipolazioni geografiche:

- Se la città non viene trovata oppure è sconosciuta, ma è presente lo Stato, viene utilizzata come città la capitale dello Stato.
- Se lo stato non viene trovato oppure è sconosciuto, ma è presente la città, viene inserito come stato quello relativo alla città. Qualora il nome della città appartenga a più stati contemporaneamente nella libreria Qlik, viene preso il primo stato caricato dal qvd.
- Unificazione dei nomi degli stati. Se per esempio alcuni record sono stati trascritti come Italy e altri come Italia, i campi valorizzati in “Italia” vengono sovriscritti come “Italy”. Questo passaggio è molto importante al fine di permettere all’utente, quando applica un filtro geografico, di prendere esattamente i dati i appartenenza senza possibilità di avere delle ambiguità.

I campi che sono stati concatenati alla tabella Platform sono dunque:

- Platform: in questo caso è stato attribuito il valore “Website”
- Read Date: data di lettura
- Reader's ID: identificativo del reader, che essendo qui anonimo, assume il valore “Unknown”
- Reader City: città di provenienza del Reader
- Reader Country: stato di provenienza del Reader
- Document ID: ID del documento di riferimento della lettura. Nel caso di Google Analytics è presente il CMS, ma grazie alla funzione di Applymap è stato possibile sostituire ciascun CMS con il document ID principale preso dalla tabella di mapping creata nella sezione xml
- Geo.country_GeoInfo: coordinante dell’area geografica della country
- Geo.city_GeoInfo: coordinate della city
- Continent: continente del reader

Alcuni dei KPI provenienti da Google Analytics non sono presenti nelle altre piattaforme, dunque risultava inutile integrarli nella tabella platform. Per mantenere tali KPI allora è stata lasciata anche la tabella aggregata di Google Analytics, contenente anche i seguenti KPI:

- # users: conteggio aggregato delle letture dei documenti dal sito Finance
 - Avg visit duration: media della durata delle letture dei documenti (in minuti)
 - Research Link Visitors: numero di utenti che accedono al sito Finance tramite link diretto al sito
- *Platform final table*: tab in cui vengono effettuate ulteriori trasformazioni sulla tabella dei fatti platform, al fine di gestire alcune eccezioni applicando le seguenti regole:
 - I dati di anagrafica dei lettori (Investor Type, City, Country ecc.) da visualizzare nell'interfaccia devono essere quelli presi da Sales Force; qualora su Sales Force essi non siano presenti, allora essi dovranno essere presi da quelli forniti dalla piattaforma su cui è avvenuta la lettura (cioè i campi Platform Country, Platform City ecc.). Ciò è sempre vero ad eccezione del campo "Investor type" che dovrà essere preso solo dall'anagrafica di Sales Force con la seguente logica: se presente a livello di readers (tabella contact), prendere quello, altrimenti prendere l'Investor Type a livello di Institution (tabella account)
 - Per le letture provenienti dalla piattaforma Bloomberg i dati di anagrafica da visualizzare nell'interfaccia devono essere quelli presi dal livello di istituzione.
 - Per le letture del sitoweb provenienti da Google Analytics i dati anagrafici non sono presenti, pertanto saranno presi solo quelli disponibili da Google Analytics (City e Country in cui è stata individuata la lettura)
 - Per i campi geografici che non sono disponibili su Salesforce e che dunque in base alla prima regola vengono presi dai dati delle piattaforme, occorre effettuare le manipolazioni geografiche effettuate nei precedenti tab:
 - Se la città non viene trovata oppure è sconosciuta, ma è presente lo Stato, viene utilizzata come città la capitale dello Stato.
 - Se lo stato non viene trovato oppure è sconosciuto, ma è presente la città, viene inserito come stato quello relativo alla città. Qualora il nome della città appartenga a più stati contemporaneamente nella libreria Qlik, viene preso il primo stato caricato dal qvd.
 - Unificazione dei nomi degli stati. Se per esempio alcuni record sono stati trascritti come Italy e altri come Italia, i campi valorizzati in "Italia" vengono sovrascritti come "Italy". Questo passaggio è molto importante al fine di permettere all'utente, quando applica un filtro geografico, di prendere esattamente i dati i appartenenza senza possibilità di avere delle ambiguità.

- Al fine di rendere i dati geografici coerenti tra loro, essi vengono presi entrambi dalla stessa sorgente secondo il criterio per cui se la city o la country è nulla su Sales Force, ma la stessa è presente sulla piattaforma, entrambi i campi cCity e Country sono presi dai dati della piattaforma. Questo metodo evita possibili incoerenze tra i dati geografici, che potrebbero emergere con l'utilizzo di due sorgenti diverse per City e Country.
- Creazione di alcuni campi di flag che identificano se per quella riga sono stati gestite le seguenti eccezioni sopra indicate, distinguendo quelle di carattere anagrafico da quelle di carattere geografico. Tali flag vengono poi utilizzati per agevolare i calcoli sull'app di Data Quality.

Le tabelle finali risultato di queste elaborazioni vengono poi storicizzate in dei qvd di secondo livello (uno per ciascuna tabella).

3.5.4. Presentation – Oggetti QlikSense

L'app di presentation è quella all'interno della quale viene creato il modello dati, sulla base delle elaborazioni dei dati effettuate nell'app ETL2. Infatti tale app carica i qvd di secondo livello al fine di ottenere il modello relazionale su cui si baseranno gli oggetti del frontend.

Il primo step è stato dunque quello di caricare i dati provenienti da ciascun ETL2. In questo caso, le chiavi di aggancio tra le tabelle erano state già create in ETL2, dunque non è risultato necessario effettuare alcuna elaborazione a riguardo. E' stato necessario invece creare una tabella di calendario, in cui per ogni data sono presenti dei campi che ne individuano la settimana, il mese e l'anno di appartenenza; tale tabella è molto utile per poter selezionare le gruppi di date aggregati, ad esempio in base al mese, oppure per utilizzare diversi livelli di aggregazione dei dati nei grafici.

Una volta caricate tutte le tabelle, la nuvola dati che ne deriva è la seguente:

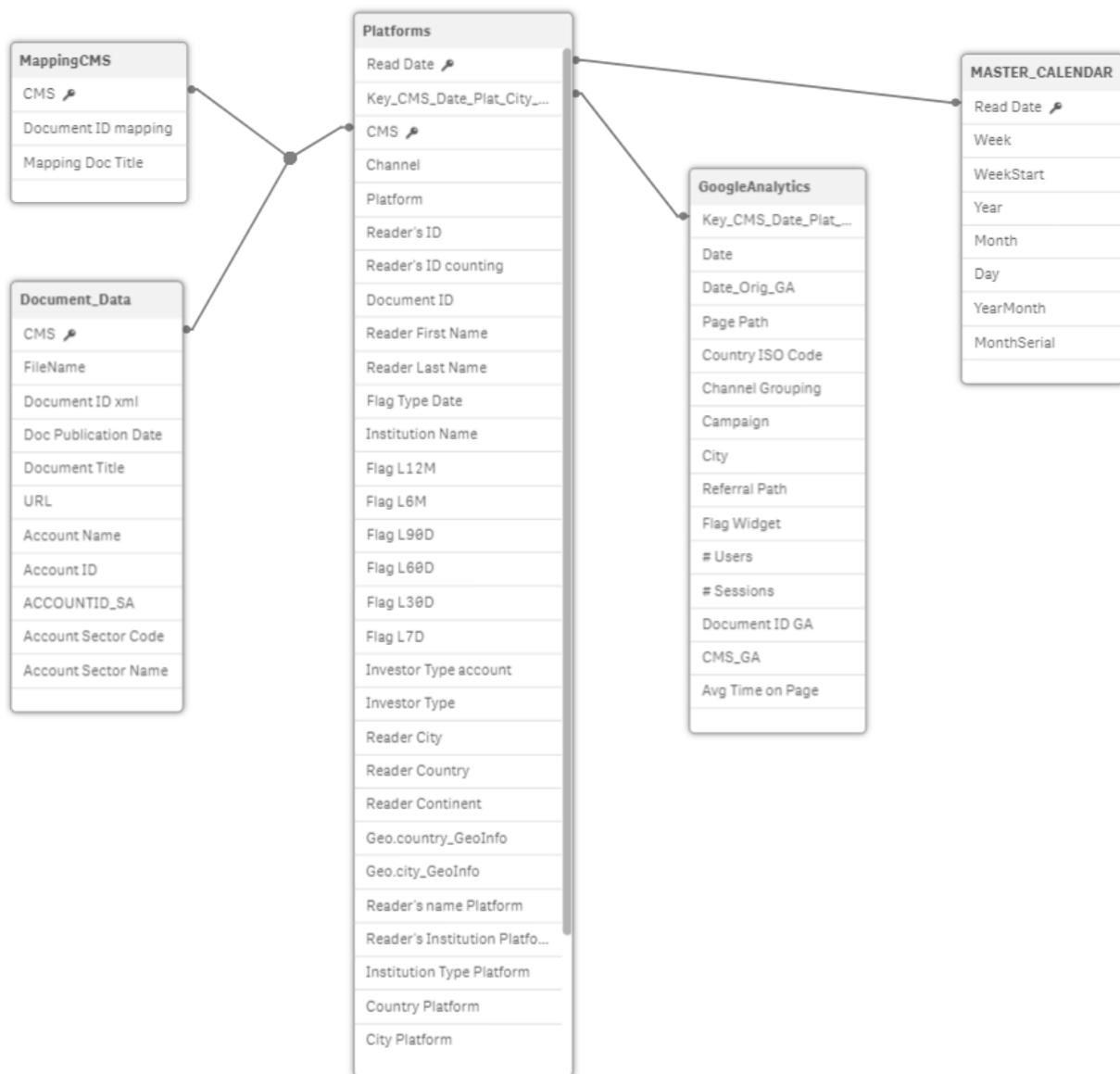


Figura 18. Data Model finale dell'app di Presentation

Il data model finale è pertanto costituito dalle seguenti tabelle:

- **Platforms**: tabella dei fatti principale: al suo interno contiene i dati di lettura di tutte le piattaforme, del website di Finance e delle mail aperte e cliccate. Per ciascun reader sono presenti anche i dati anagrafici. Tali dati non sono stati inseriti in una tabella apposita in quanto essi vengono presi dalla piattaforma o dall'anagrafica Sales Force in base al fatto che essi siano presenti o meno su quest'ultima
- **Document Data**: tabella di anagrafica contenente per ciascun documento, dati anagrafici come il nome, la società di appartenenza, la data di pubblicazione ecc. Si collega alla tabella dei fatti platform tramite il campo CMS, che identifica il documento in modo univoco.
- **MappingCMS**: tabella di anagrafica contenente per ciascun CMS l'elenco dei codici documento associati. Essendo state create le chiavi in base al CMS, tale tabella non risulta strettamente necessaria, ma dato che contiene poche righe, è stata lasciata per poter velocizzare eventuali controlli dei dati, dovuti ad esempio all'inserimento di

nuovi codici documento associati a dei CMS. Anche questa tabella è collegata alla tabella Platforms tramite il codice CMS.

- *Google Analytics*: tabella contenente i dati aggregati di Google Analytics in cui sono presenti specifici KPI non divisibili, per i quali risulta necessario mantenere il dato della tabella allo stato di aggregazione originario. Questa tabella viene utilizzata esclusivamente per il calcolo di tali KPI, infatti i dati di lettura di Google Analytics sono presenti anche all'interno della tabella Platforms, in modo da avere accesso allo stesso livello di aggregazione per tutte le piattaforme. La chiave di collegamento della tabella con la tabella Platforms è data dalla concatenazione dei seguenti campi:

- CMS
- Read Date
- Platform
- City
- Country

Tale chiave rappresenta il massimo livello di dettaglio a cui arriva la tabella Google Analytics, e permette quest'ultima di essere correttamente filtrata sull'interfaccia in base alle selezioni applicate dall'utente.

- *Master Calendar*: tabella contenente, per ciascuna data di lettura, gli attributi relativi alla settimana, mese e anno di appartenenza.

E' utile notare che il modello dati finale si discosta parzialmente da quello ipotizzato inizialmente (rif. paragrafo 3.5.1); questo è dovuto al fatto che durante il progetto, una volta emerse tematiche come la gestione dei dati anagrafici, è risultato indispensabile provvedere trovare nuove soluzioni, che hanno portato a questi cambiamenti. La struttura di base però non è cambiata, ma risulta abbastanza fedele a quanto preventivato.

Una volta creato il modello dati è necessario passare alla progettazione degli oggetti Qlik Sense che verranno inseriti all'interno dell'interfaccia web QlikSense (mashup). Tali oggetti, prendono spunto da quelli condivisi tramite mockup nelle prime fasi del progetto, anche se con parziali modifiche. Di seguito elenchiamo gli oggetti definitivi creati, suddivisi per pagina di appartenenza sul mashup:

HomePage

- *Tabella dei KPI*: tabella contenente tutti i KPI che saranno inseriti nella Homepage

KPI						
= 'Total'	Q	Reports published	Total reads	Unique readers	Countries reached	New readers
Total		4,879	8,355,525	96,006	220	21,161

Figura 19. Tabella contenente i KPI della HomePage

Le logiche di calcolo di tali KPI sono le seguenti:

- Reports published: conteggio distinto dei CMS
- Total reads: conteggio dei lettori (non distinto)

- Unique readers: conteggio distinto dei lettori esclusi quelli sconosciuti
 - Countries reached: conteggio distinto degli Stati
 - New readers: conteggio dei readers che hanno letto un solo documento una sola volta
- **Grafico di Trend:** andamento delle letture nel tempo. La dimensione utilizzata è una dimensione di drill che in base alla selezione permette di passare alla visualizzazione del numero di readers negli anni, mesi o giorni. Le misure calcolate sono il numero di letture e la cumulata delle letture nel tempo:
 - **Total Reads:**
`count ([Reader 's ID])`
 - **Cumulated Reads:**
`RangeSum (above (column (1) , 0 , RowNo ()))`

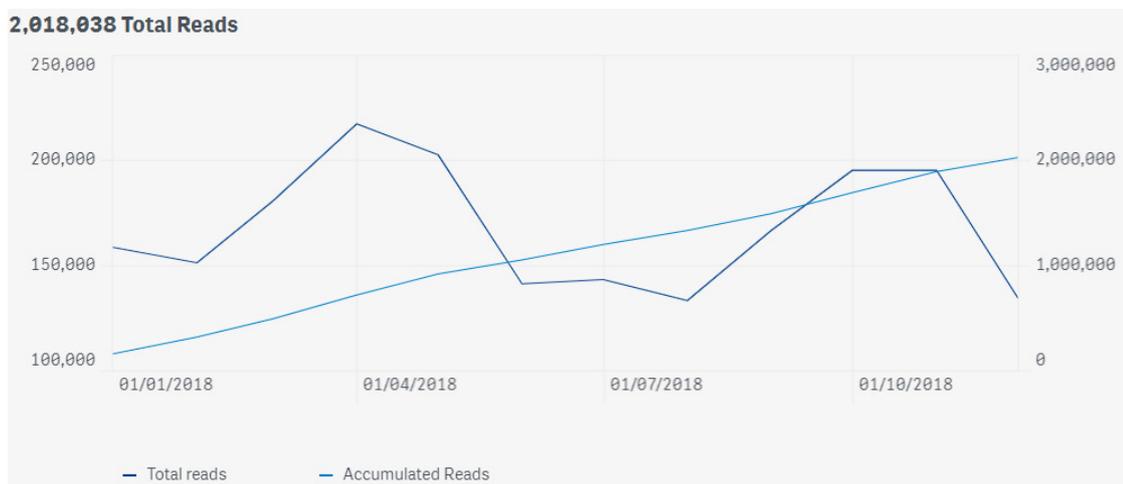


Figura 20. Grafico di trend

- **Grafici a torta:** identificano la suddivisione percentuale delle letture in base alle dimensioni Investor Type, Channel e Country. Dal conteggio vengono esclusi i valori delle dimensioni “Unknown” e il massimo numero di valori della dimensione visualizzabili sono 10, gli altri valori vengono inseriti in un unico valore denominato “others”. Per ciascun grafico viene inserita la dimensione limitata ai primi 10 valori, mentre la misura è data dal conteggio delle letture.
 - **Total Reads Investor Type:**
`count ({ $ < [Investor Type] -= { 'Unknown' } > } [Reader 's ID])`
 - **Total Reads Country:**
`count ({ $ < [Reader Country] -= { 'Unknown' } > } [Reader 's ID])`

➤ Total Reads Channel:

Count ([Reader 's ID])

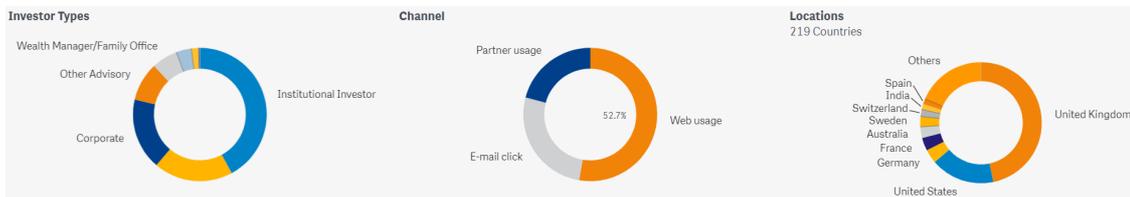


Figura 21. Grafici a torta

Data Analysis

- **Grafici a torta:** identificano la suddivisione percentuale delle letture in base alle dimensioni Investor Type, Channel e Location. Dal conteggio vengono esclusi i valori delle dimensioni “Unknown” e il massimo numero di valori della dimensione visualizzabili sono 10, gli altri valori vengono inseriti in un unico valore denominato “others”. Per ciascun grafico viene inserita la dimensione limitata ai primi 10 valori, mentre la misura è data dal conteggio delle letture. In altre parole vengono utilizzati gli stessi oggetti che erano presenti nella pagina Homepage. Questa duplicazione è dovuta al fatto che mentre la Homepage è una tabella di sommario delle analisi, Data Analysis è una tabella di dettaglio in cui per analizzare meglio i dati è comodo avere gli stessi grafici senza necessità di cambiare di volta in volta pagina.
- **Sankey Chart:** grafici che identificano visivamente quali documenti sono visualizzati da certe tipologie di utente e in che quantità. Tale tipologia di grafico necessita di essere settato con almeno 2 dimensioni e una misura che nel nostro caso sono:
 - **Dimensione 1: Investor Type/ Institution/ Country:** i grafici creati sono tre, ciascuno per ciascuna di queste dimensioni. Per evitare di avere troppe dimensioni poco influenti, che renderebbero poco leggibile il grafico, è stato deciso di settare al massimo 5 valori per ciascuna dimensione, che corrispondono ai 5 valori con più letture. Non essendo su questo grafico possibile settare questa impostazione, abbiamo creato delle dimensioni calcolate che seguono questa logica. Segue il calcolo dalla dimensione top 5 “Investor Type”:

```
if(GetPossibleCount([Investor Type]) >= 10,
  aggr (
    if(rank(count({<[Investor Type]--{'Unknown'}>}[Reader's
ID]),1,1)<=10,[Investor Type],null())
    ,[Investor Type]),
  [Investor Type])
```

Per le dimensioni Institution, Country il calcolo è identico, cambia solo il nome della dimensione su cui fare il calcolo

- **Dimensione 2: Document Name:** nome del documento oggetto di letture. Anche qui per evitare di avere troppe dimensioni poco influenti, è stato deciso di settare al massimo i 5 documenti più letti:

```
if(GetPossibleCount([Document Name]) >= 10,
  aggr(
    if(rank(count([Reader's ID]),1,1)<=10[Document Name],null())
    , [Document Name]),
  [Document Name])
```

- **Misura: conteggio delle letture**

```
count([Reader's ID])
```

In figura un esempio di Sankey chart per Investor Type:

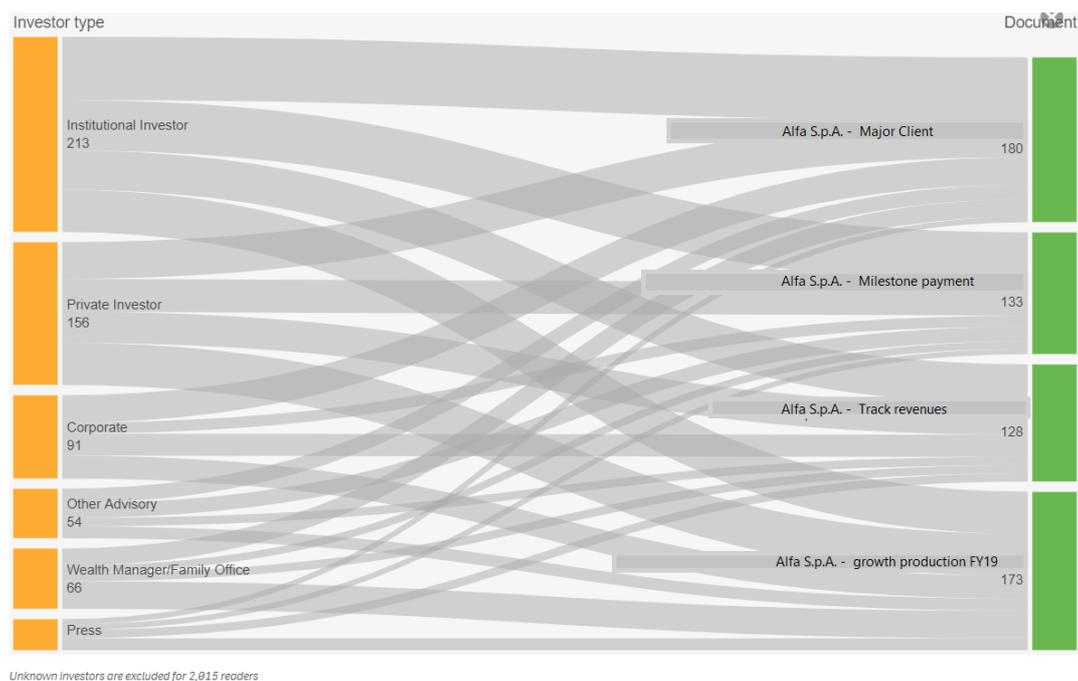


Figura 22. Sankey chart

- **Bar Chart:** grafico che identifica quali sono i 5 documenti più letti e qual è il numero di letture ad essi associato. In questo caso dunque la dimensione inserita è il Document Name settato per i soli primi 5 in base alla misura, che è data dal conteggio delle letture:

```
count([Reader's ID])
```

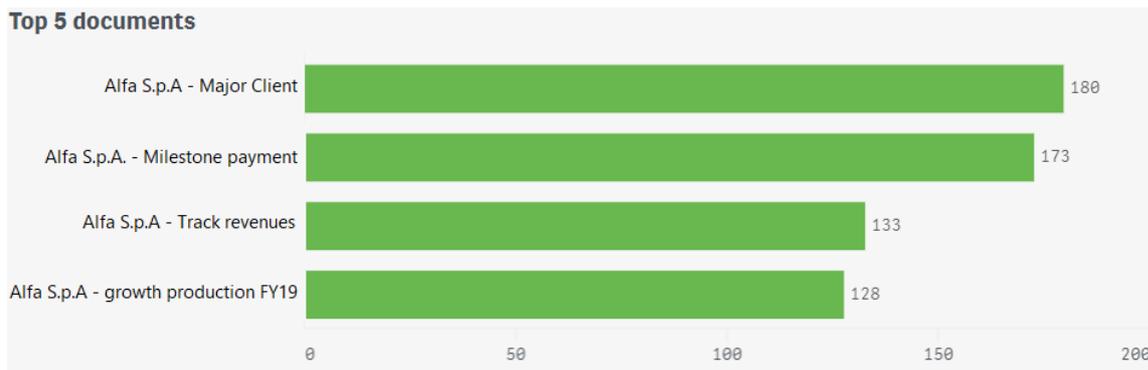


Figura 23. Grafico a barre

- *Tabella*: identifica per ogni documento, il numero di letture percentuale di letture sul totale. La dimensione utilizzata pertanto è il Document Name, mentre le misure sono date dal conteggio delle letture e dal conteggio delle letture diviso il conteggio totale delle letture presenti in tabella.

- **Reads:** `count ([Reader 's ID])`
- **% Reads:** `count (Reader 's ID) / count (total [Reader 's ID])`

Document	Reads	% Reads
Total	614	100,00%
Alfa S.p.A - Major Client	180	29,32%
Alfa S.p.A. - Milestone payment	173	28,18%
Alfa S.p.A - Track revenues	133	21,66%
Alfa S.p.A - growth production FY19	128	20,85%

Figura 24. Tabella letture per documento

Data Analysis

- *Tabella*: identifica per ogni documento la sua data di pubblicazione. Inoltre la selezione dalla tabella di uno qualsiasi dei documenti apre la pagina web contenente il report del documento.

Document	Doc Publication Date
Total	
Alfa S.p.A - Major Client	24/06/2019
Alfa S.p.A. - Milestone payment	09/04/2019
Alfa S.p.A - Track revenues	24/12/2018
Alfa S.p.A - growth production FY19	13/12/2018

Figura 25. Tabella documenti pubblicati

- **Grafico di Trend:** permette di visualizzare l'andamento delle letture nel tempo. Analogamente al grafico presente nella Homepage, anche qui la dimensione utilizzata è una dimensione di drill che in base alla selezione permette di passare alla visualizzazione del numero di letture negli anni, mesi o giorni.

La misura inserita è data dal seguente calcolo:

```
count ([Reader 's ID])
```

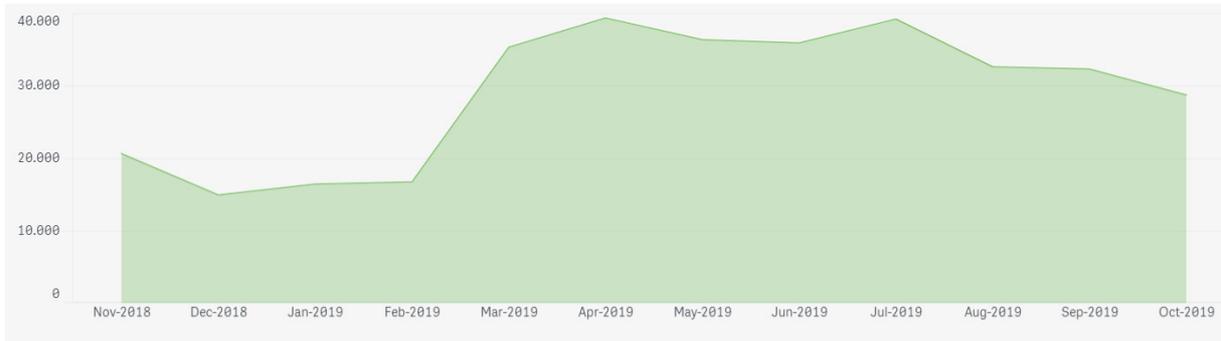


Figura 26. Grafico di trend per anno - mese

- **Table:** contengono i KPI relativi al sito di Finance e delle mail aperte e cliccate. I KPI calcolati sono:
 - **Unique visitors:** numero di lettori dei documenti presi da Google Analytics
`sum ([# Users])`
 - **Avg visit duration:** media della misura “Avg Time On Page” della tabella Google Analytics
`Avg ([Avg Time on Page])`
 - **ResearchLink visitors:** numero di letture avvenute tramite widget, calcolato in base alla somma della misura “Flag Widget” presa dalla tabella Google Analytics
`Sum ([Flag Widget])`
 - **E-mail opens:** numero di mail aperte dai lettori, date dal conteggio delle righe della tabella platforms aventi il campo “Flag Type Date” = “Open”.
`Count ({$<[Flag Type Date]='Open'}>)[Flag Type Date])`
 - **E-mail click throughs:** numero di email in cui è stato aperto il link che rimanda al documento oggetto della mail, date dal conteggio delle righe della tabella platforms aventi il campo “Flag Type Date” = “Click”.
`Count ({$<[Flag Type Date]='Click'}>)[Flag Type Date])`

KPI GA				
= 'Total'	Q	Unique visitors	Avg visit duration	ResearchLink visitors
Total		197,954	60.64	8,978
Total		197,954	60.64	8,978

Figura 27. Tabella contenete i KPI relativi al website

KPI DM			
= 'Total'	Q	E-mail opens	E-mail click-throughs
Total		7,098,855	332,928
Total		7,098,855	332,928

Figura 28. Tabella contenete i KPI relativi alle mail

Geo Analysis

- *Mappa Geografica:* creata su due livelli di densità, Country e City. Il livello country è un livello area con 100% trasparenza che consente lo zoom automatico sullo stato di appartenenza dei dati, qualora vengano applicati i filtri. La posizione della Country è individuata sulla mappa grazie al campo Geo.country_GeoInfo che contiene le coordinate dell'area geografica. Il livello city è un livello densità, in cui la coordinata della città è individuata dal campo Geo.city_GeoInfo che contiene la coordianta puntuale della città. Il peso e il raggio sono calcolati tramite il conteggio delle letture su ciascuna coordinata; essendo un livello densità con mappa di calore, maggiore sarà il valore della misura per una determinata coordinata o area, più rossa e grande sarà evidenziata la zona.



Figura 29. Mappa Geografica a densità

La figura seguente mostra come, grazie all'inserimento del livello area Country, selezionando una città come Londra, per la quale è associata una coordinata puntuale, lo zoom automatico non si riduce al singolo punto, che sarebbe visivamente poco significativo, ma permette di visualizzarlo ad un livello di zoom tale da vedere lo stato di appartenenza:



Figura 30. Mappa Geografica filtrata su una città

- **Bar graph:** grafici che permettono di vedere come le letture sono suddivise per dimensione geografica, tipologia di investitore e piattaforma di lettura. I grafici creati nello specifico sono due, ciascuno con le seguenti dimensioni e misure:

- Continent, Country, City: dimensione drill down che cambia livello di aggregazione rispetto ai filtri applicati dall'utente. Di default è visibile la suddivisione per continente, ma se ad esempio si seleziona uno Stato, la dimensione cambia in automatico e diventa a livello di città
- Investor Type, Channel: ogni grafico possiede una di queste dimensioni, che lo differenzia dall'altro
- Reader City Geo: conteggio delle letture escludendo i dati che hanno città, stato e continente nulli. La formula implementata è la seguente:

```
count({$<[Reader City]=={'Unknown'}, [Reader Continent]-
    ={'unknown'}, [Reader Country]=={'Unknown'}>}[Reader's ID])
```

In figura il grafico settato per Channel, a livello di aggregazione geografica massimo (Continent):

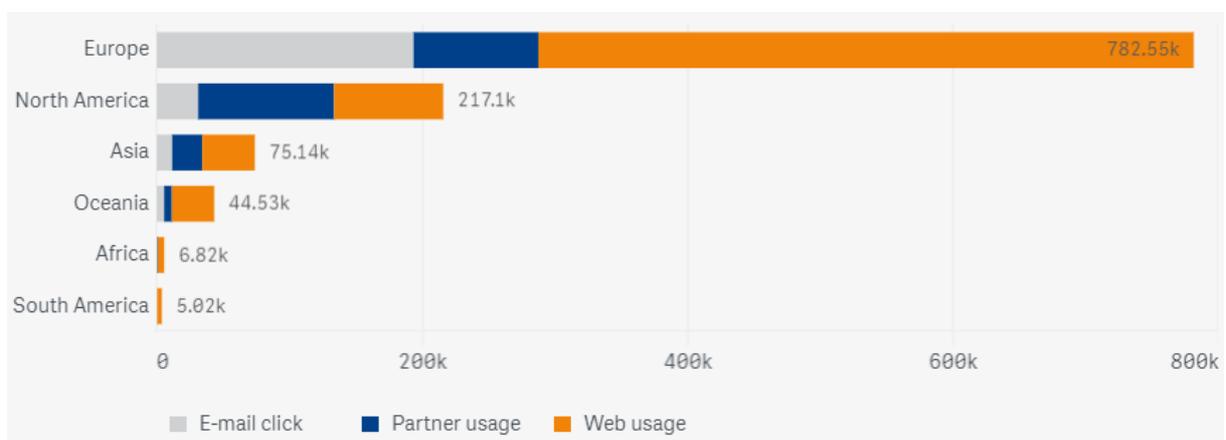


Figura 31. Grafico a barre letture per continente

Readers Report

- **Tabella:** tabella contenente per Institution, Institution Investor Type, City e Country, le seguenti misure:
 - **Website:** conteggio delle letture avvenute sul sito di Finance
`Count({$<Platform={'Edison Website'}, [Flag Type Date]='Read'}) [Reader's ID]`
 - **E-mail opens:** conteggio delle mail aperte dai lettori di ciascuna società
`Count({$<[Flag Type Date]='Open'}) [Flag Type Date]`
 - **E-mail click:** conteggio delle mail nelle quali è stato cliccato l'url che porta al documento pubblicato sul web
`Count({$<[Flag Type Date]='Click'}) [Flag Type Date]`
 - **Institutional Platform:** conteggio delle letture avvenute tramite le altre piattaforme
`Count({$<Platform--={'Edison Website'}}) [Reader's ID]`
 - **Reads:** conteggio delle letture totali, comprensive di mail esclusivamente aperte
`Count([Reader's ID])`

Institution	Institution Investor Type	City	Country	website (google analytics only)	E-mail opens	E-mail click-throughs	Institutional Platform	Reads
Total				10	1,620	26	23	1,649
Alfa S.p.A	Unknown	Perth	Australia	0	21	0	0	21
Beta s.r.l	Institutional Investor	Singapore	Singapore	0	0	0	3	3
Gamma s.r.l	Institutional Investor	Auckland	New Zealand	0	25	0	0	25
Teta S.p.A	Corporate	Cambridge	United Kingdom	0	1,454	26	10	1,480
Delta S.p.A	Unknown	Lone Tree	United States	0	13	0	0	13
Firm one s.r.l	Wealth Manager/Family Office	London	United Kingdom	0	19	0	0	19
Firm two S.p.A	Corporate	Berlin	Germany	0	64	0	0	64
Company s.r.l	Corporate	Heek	Germany	0	24	0	10	24
Unknown	Unknown	Milano	Italy	7	0	0	0	7
Unknown	Unknown	Kirkland	United States	3	0	0	0	3
	Wealth Manager/Family Office	Leeds	United Kingdom	0	1	0	0	1

Figura 32. Tabella di dettaglio letture per institution

Filters

- **Caselle di elenco:** costituiscono i filtri che saranno resi disponibili nell'interfaccia utente. Ogni filtro corrisponde ad una casella di elenco

Tutte le misure calcolate sono state create come master item, che verranno poi utilizzati all'interno del mashup.

L'app di presentation è inoltre messa a disposizione degli utenti come strumento di data quality, in quanto permette loro di fare controlli tra i dati di anagrafica ricevuti dalle piattaforme partners e i dati da loro censiti sull' anagrafica Sales Force. Fare queste verifiche per loro è semplice grazie alle funzioni di self -service BI QlikSense, che permettono in modo facile e intuitivo di creare tabelle e grafici semplicemente trascinando oggetti, dimensioni e misure sul proprio foglio di lavoro dell' App QlikSense.

3.5.5. Presentation – Mashup

Usualmente è consigliabile che la pagina web del mashup venga implementata in un server differente dal Qlik Sense server. Nel caso in esame infatti il mashup viene processato sulla QAP (Qlik Analytics Platform), una piattaforma di sviluppo per la creazione di applicazioni analitiche personalizzate basate su API front-end e back-end avanzate.

La QAP Offre l'accesso completo alle API al motore associativo di Qlik per creare applicazioni Web per la distribuzione extranet e Internet.

Le componenti della QAP sono:

- Qlik Management Console (QMC) e Dev Hub
- API e SDK di Qlik Sense
- Motore associativo di Qlik e servizi di supporto di Qlik Sense

Una delle differenze sostanziali che hanno motivato l'utilizzo della QAP per il mashup è il differente modello di licenza rispetto a Qlik Sense Enterprise. Infatti la QAP non necessita di token e pass di accesso (acquisto di un numero definito di utenze), ma la distribuzione è vincolata per capacità del Server (core della CPU). Questo è un enorme vantaggio in quanto il numero di clienti di Finance che possono accedere diventa limitato solo alla capacità del server, non dal numero di utenze assegnate [21].

I costi della licenza sono dunque relativi al numero di core che vengono messi a disposizione sul server QAP.

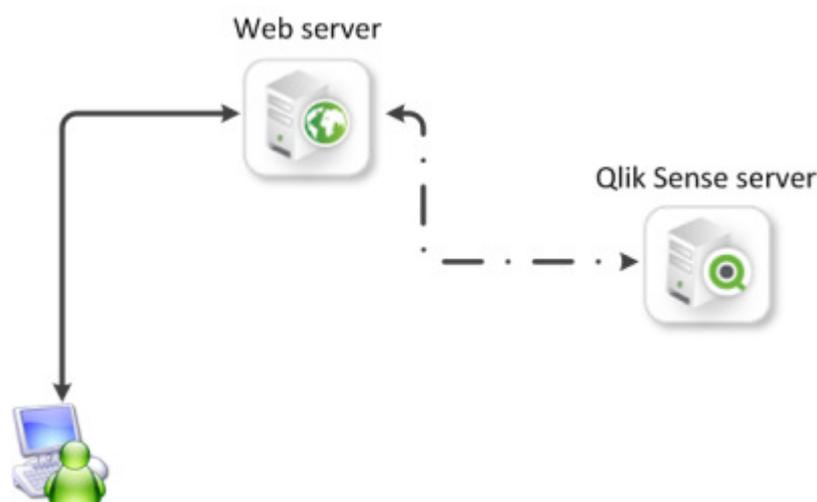


Figura 33. Schema scambio informazioni tra Qlik Sense Server – QAP e client [21]

Al fine di utilizzare gli oggetti dell'app di Presentation sul Mashup è necessario ricavarne gli ID e collegare l'app con il mashup; è importante dunque che l'app di presentation venga inserita anche all'interno della QAP. In questo caso la schedulazione oraria di aggiornamento dell'app, non dipendendo da altri task ETL come nel caso del Serve Qlik Sense, è stata schedulata alle ore 9:00 di ogni giorno. Tale scelta è dovuta ad un calcolo medio effettuato sui tempi di caricamento di tutta la catena ETL sul server Qlik Sense. È infatti necessario che l'orario di inizio caricamento dell'app sia successivo alla fine del caricamento di tutti gli ETL per poter attingere dai qvd aggiornati. Per precauzione abbiamo infatti inserito circa un paio di ore di buffer tra la fine del caricamento degli ETL e l'inizio del caricamento dell'app di presentation della QAP.

Di seguito un'immagine dello strumento di sviluppo del mashup, in cui vengono inseriti gli ID degli oggetti Qlik da inserire. Per quanto riguarda la creazione del layout della pagina, lo strumento è analogo a quello utilizzato per le pagine web:

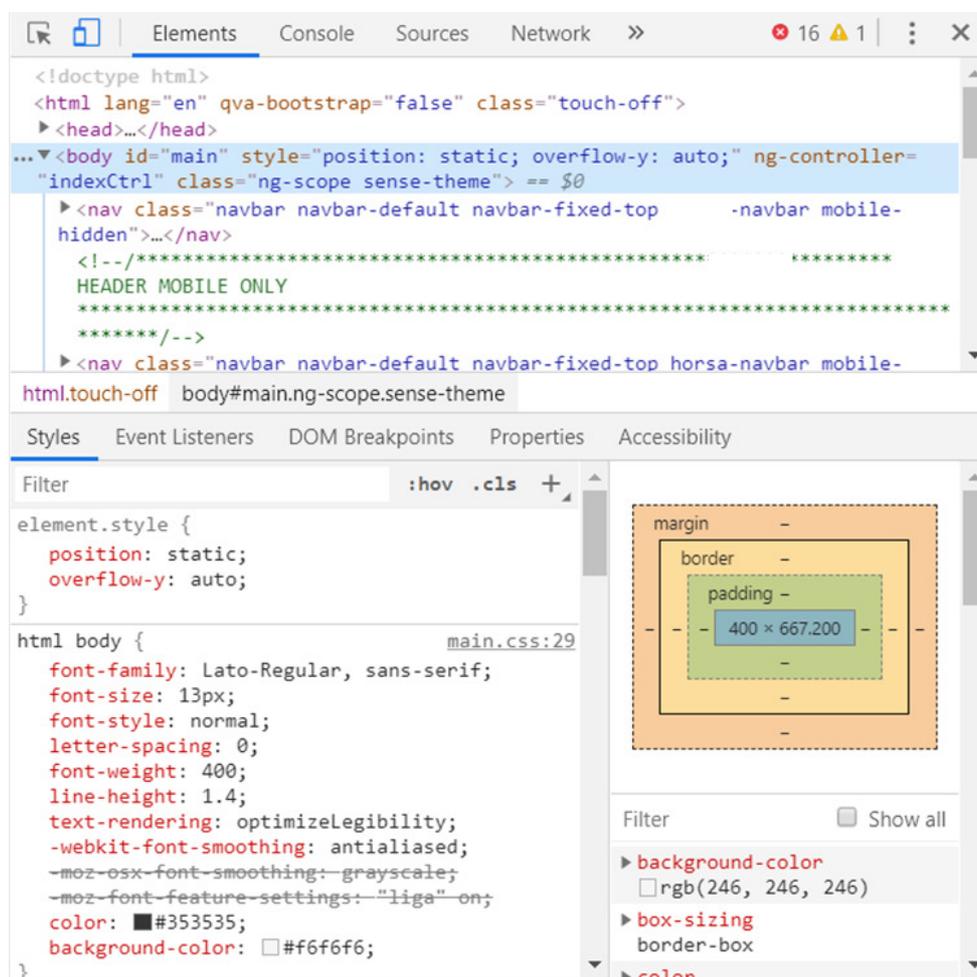


Figura 42. Editor Mashup

Tale paragrafo è da intendersi solo come *descrizione generica* sul funzionamento del mashup e delle scelte architettoniche effettuate; non essendo nel mio perimetro di competenza di progetto la creazione dell'interfaccia web, non sono riportati i dettagli sulla costruzione di quest'ultima, per le quali sono necessarie competenze specifiche in ambito Java, html e css.

3.6. Cartelle di progetto e schedulazione task di aggiornamento

I dati elaborati dagli ETL Qlik di cui abbiamo parlato nei paragrafi precedenti, storicizzano i file in formato qvd e talvolta attingono da file esterni invece che da tabelle presenti in dei data base. Per tale motivo, è necessario creare una cartella di progetto all'interno della quale mantenere tali file. Al fine di rendere ordinato e strutturato il processo, è utile creare un framework di progetto in cui dividere i dati in sottocartelle.

Nel caso del progetto in esame il framework di progetto creato è il seguente:

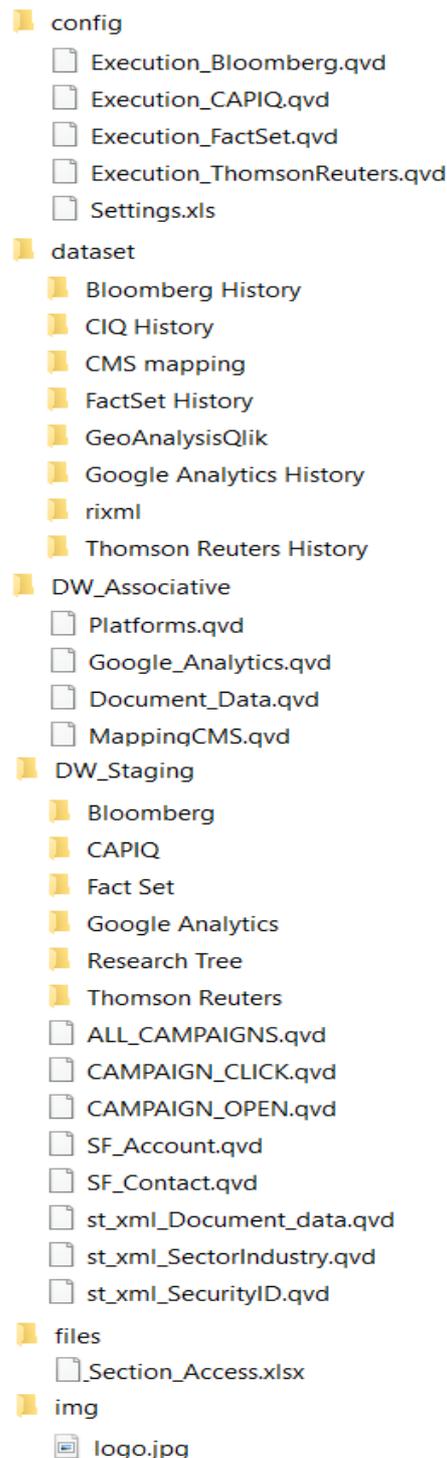


Figura 34. Framework di progetto

Tale Framework è strutturato su 6 cartelle principali:

- **Config:** cartella contenente il file di mappatura delle variabili utilizzato per centralizzare i paths ai file e dei qvd di appoggio che contengono la lista dei file già caricati e storicizzati per alcune sorgenti di cui estraiamo di volta in volta i dati da directory contenenti elenchi di file. In questo modo è possibile in ciascun caricamento estrarre solo i file effettivamente nuovi contenuti in queste directory.
- **Dataset:** contiene i file di storico iniziali che sono stati dati da caricare one-shot, non presenti nelle sorgenti da cui estraiamo i dati con logica incrementale. Tali file di storico sono stati inseriti in cartelle diverse suddivise per piattaforma di riferimento. Inoltre è presente la cartella GeoAnalyticsQlik, in cui sono presenti i file della libreria Qlik per l'individuazione delle coordinate geografiche, e il file di supporto da noi creato per individuare il collegamento tra Country, Capitale e Continente.
- **DW_Associative:** contiene i file qvd di secondo livello corrispondenti alle tabelle del data model che viene caricato sull'app di presentation. Tali qvd sono frutto dell'elaborazione dell'ETL2.
- **DW_Staging:** contiene i file qvd di primo livello suddivisi per piattaforma. All'interno di ogni cartella i qvd sono suddivisi per anno-mese. Tali qvd sono frutto delle storicizzazioni effettuate negli ETL1.
- **Files:** contiene il file utilizzato per la Section Access, necessario per la profilazione sugli utenti e sui dati.

Per ciò che concerne la schedulazione dell'aggiornamento degli ETL, è necessario accedere alla Qlik Management Console (comunemente chiamata qmc) per pubblicare le apps, inserirle in degli stream di lavoro (opzionale) e creare dei task ad esse associate che permetta loro di aggiornarsi in automatico secondo i criteri orari e di precedenza stabiliti.

Tramite la qmc è dunque possibile configurare, gestire e monitorare gli ETL e le apps QlikSense, gestendo le licenze, regole di modifica, lettura e accesso alle apps. È inoltre possibile modificare o creare nuove connessioni alle sorgenti dati, importare extension e apps.

Una volta pubblicate tutte le apps di ETL e Presentation (azione che permette agli utenti censiti di visualizzarle ed eventualmente duplicarle/modificarle all'interno dell'Hub QlikSense), è possibile creare i task di caricamento e dare loro una schedulazione.

In figura è mostrato l'elenco dei task creati, uno per ciascuna app:

Name	Associated resource	Type	Enabled	Status	Last execution	Next execution
ETL1 Bloomberg	ETL1 Bloomberg	Reload	Yes	✓ Success	2019-11-05 02:00	2019-11-06 02:00
ETL1 CAPIQ	ETL1 CAPIQ	Reload	Yes	✓ Success	2019-11-05 02:00	On task event trigg...
ETL1 Dotmailer	ETL1 DotMailer	Reload	Yes	✓ Success	2019-11-05 02:04	On task event trigg...
ETL1 FactSet	ETL1 FactSet	Reload	Yes	✓ Success	2019-11-05 02:03	On task event trigg...
ETL1 Google An...	ETL1 Google Anal...	Reload	Yes	✓ Success	2019-11-05 07:09	On task event trigg...
ETL1 Research ...	ETL1 Research Tree	Reload	Yes	✓ Success	2019-11-05 02:03	On task event trigg...
ETL1 SalesForces	ETL1 SalesForces	Reload	Yes	✓ Success	2019-11-05 07:08	On task event trigg...
ETL1 Thomson ...	ETL1 Thomson Re...	Reload	Yes	✓ Success	2019-11-05 02:03	On task event trigg...
ETL1 XML	ETL1 XML	Reload	Yes	✓ Success	2019-11-05 07:07	On task event trigg...
ETL2	ETL2	Reload	Yes	✓ Success	2019-11-05 07:09	On task event trigg...
Presentation	Presentation	Reload	Yes	✓ Success	2019-11-05 07:55	On task event trigg...

Figura 35. Schedulazione tasks sulla qmc

Come si evince dalla figura, l'aggiornamento della catena è giornaliero con partenza alle 2:00. Il primo task che esegue l'aggiornamento dei dati è quello relativo all' ETL1 – Bloomberg; al termine dell'esecuzione dello script di Bloomberg, parte il task relativo all'aggiornamento del task ETL1 - CAPIQ. Il processo continua con questo standard fino all' aggiornamento dell'app di presentation.

Come detto nel paragrafo 3.5.5 relativo al Mashup, l'app che viene utilizzata per collegarsi agli oggetti Qlik Sense è quella presente nella QAP.

Quella presente sul Server QlikSense, è una copia di quella presente sulla QAP, è la versione che gli utenti utilizzano per effettuare analisi di Data Quality, essendo possibile accedere all'Hub (che invece la QAP non ha).

3.7. Section Access

L'autenticazione degli utenti su QlikSense avviene al momento dell'accesso sull'Hub, applicazione o mashup nel caso di link diretto questi ultimi. In tale contesto Qlik (o il sistema operativo Windows) richiede di inserire la UserID e password per l'accesso, in modo da verificare quest'ultimo è autorizzato ad accedere alla risorsa.

Il controllo degli accessi tramite Section Access viene gestito mediante una o più tabelle caricate secondo la normale procedura applicata a tutti i dati Qlik. Le istruzioni di script che gestiscono le tabelle di sicurezza devono essere incluse in una apposita sezione relativa al controllo degli accessi, riconoscibile nello script perché viene inizializzata dall'istruzione `Section Access`. La restante parte dello script destinata al caricamento dei dati normali deve invece fare parte di una sezione differente, inizializzata dall'istruzione `Section Application` [20].

Nel caso del progetto in esame è stato creato un file Excel esterno, con all'interno tutti gli USERID autorizzati ad accedere all'applicazione e di conseguenza al mashup. La tabella è formata dalle seguenti colonne:

- **ACCESS:** tipo di accesso, può assumere i valori ADMIN o USER e serve per distinguere i ruoli degli utenti. Gli accessi ADMIN sono abilitati alla duplicazione e modifica dell'app all'interno dell'Hub Qlik Sense, mentre gli accessi USER sono coloro che possono solamente visualizzare il mashup
- **USERID:** Id dell'utenza, comprensiva di dominio
- **ACCOUNTID_SA:** codice Account visualizzato. Ogni user infatti può visualizzare solo i dati gli account (e dunque dei clienti) per i quali è abilitato. Tale campo infatti si lega all'interno della nuvola dati con un campo analogo presente nella tabella `Document_Data`, che permette di visualizzare solo i dati associati agli `ACCOUNTID_SA` appartenenti alla `USERID` che accede all'app e al mashup.

Possiamo dunque concludere che la Section Access assume una duplice funzionalità:

- Permette l'accesso solo agli user censiti, previa autenticazione degli stessi
- Filtra per ciascun utente la visualizzazione dei soli dati per cui l'utente è censito

Di seguito lo script che è stato inserito all'interno dell'app di presentation QlikSense:

```
SA_Table:
LOAD DISTINCT
    ACCESS,
    Upper(USERID) as USERID,
    if(len([ACCOUNTID_SA])=0, '', [ACCOUNTID_SA]) as [ACCOUNTID_SA]
FROM [lib://folders/files/Section_Access.xlsx]
(ooxml, embedded labels, table is Foglio1);
Section Application;
Drop Table SA_Table;
```

3.8. UAT e passaggio in produzione

La fase finale di test con gli utenti è molto delicata per la buona riuscita di un progetto IT. In questa fase vengono infatti effettuati i test sui dati che determinano la correttezza delle logiche implementate durante la fase di sviluppo, nonché l'attendibilità delle analisi visualizzate.

Un buon metodo per limitare le incorrettezze ed evitare una fase finale di test piena di bug o modifiche ai requisiti è quello di rilasciare parti di applicazione, e in questo caso dell'interfaccia web, durante fasi intermedie del progetto, in modo da dare agli utenti la possibilità di chiedere preventivamente delle modifiche o risoluzioni di bug, che saranno risolti durante le fasi di sviluppo del progetto.

Nel nostro caso, come è ben visibile dal Gantt di progetto presente nel paragrafo 3.4, sono state inserite 4 milestones:

- **AWS:** milestone che segnala la fine dell'installazione e messa a punto della nuova infrastruttura QlikSense. In tale circostanza sarà dunque possibile traslare quanto già sviluppato nei server interni del fornitore sul server definitivo di Finance. In tale circostanza il cliente non possiederà ancora gli strumenti per effettuare test sull'applicazione, ma qualora emergano delle criticità dovute ad esempio alla

ricreazione dei connettori, è possibile ricevere il supporto da quest'ultimo, qualora risulti necessario.

- P1: milestone in cui viene rilasciato il primo prototipo del mashup contenente alcune delle pagine preventivate. In questa situazione è stato possibile per l'utente iniziare ad effettuare i primi test sui dati, capire se le analisi create sono effettivamente utili per lo scopo finale e testare la navigabilità dell'interfaccia
- P2: rilascio completo dell'interfaccia web, con tutte le pagine e le analisi preventivate, più eventuali cambi di logiche, layout e modifiche emerse successivamente al primo rilascio. In questa fase è compito degli users effettuare tutti i test possibili sui dati, le analisi e la navigabilità dell'interfaccia.

Eventuali bug o modifiche vengono scritte all'interno di un file condiviso tra tutti stakeholders del progetto, in cui si segnala il tipo di anomalia o modifica da fare, e se ne tracciano i rilasci. Ogni volta che viene fatto un rilascio della modifica sull'interfaccia occorre segnalarlo al cliente che provvederà a testarne la correttezza.

Questa è una fase molto delicata, di continua collaborazione tra cliente, PM e team di sviluppo.

- P3: termine ultimo per il rilascio dell'interfaccia con tutte le modifiche segnalate nella fase precedente. A questo punto è possibile aprire l'interfaccia web agli user finali.

4. I risultati

Di seguito viene descritto il funzionamento dell'interfaccia finale, risultato del progetto, composta da 5 sezioni:

- Homepage
- Data Analysis
- Trend Analysis
- Geo Analysis
- Readers Report

In testata a tutte le pagine sono presenti i seguenti oggetti:

- Menù di navigazione: permette all'utente di passare da una pagina di analisi ad un'altra
- Caselle di elenco dei filtri: permettono di applicare dei filtri alle analisi visualizzate. In questo modo i dati visualizzati saranno relativi esclusivamente ai dati compatibili con i filtri impostati.

È possibile filtrare le seguenti dimensioni:

- City: città del lettore
 - Country: stato del lettore
 - Continent: continente del lettore
 - Investor type: tipo di investitore
 - Institution: azienda del lettore
 - Platform: piattaforma di provenienza della lettura
 - Account: nome del cliente; ciascun cliente vede solo i dati della sua azienda, quindi non può filtrare per altri clienti. Gli utenti di Finance invece possono filtrare l'azienda cliente
 - Document: nome del report
 - Date range: oggetto calendario tramite cui è possibile selezionare un intervallo temporale specifico su cui analizzare le letture, di default viene ricoperto un range temporale di 3 anni + 1 rolling.
- Pulsante di pulizia dei filtri: tramite pop-up permette di visualizzare i dettagli dei filtri applicati, oltre a consentire la cancellazione di questi ultimi tramite click.

Di seguito una figura che mostra la testata di una pagina:

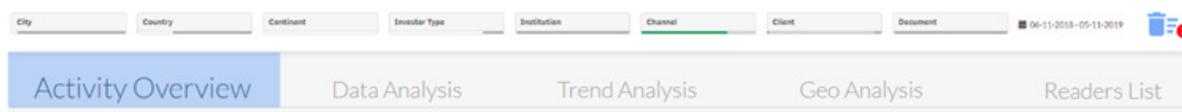


Figura 36. Mashup - filtri di selezione e pulsanti di navigazione tra le pagine

4.1. Activity Overview

Questa pagina fornisce un sommario dei principali KPI e dell'andamento delle letture, dando evidenza delle principali caratteristiche dei lettori.
Di seguito l'immagine dell'interfaccia:

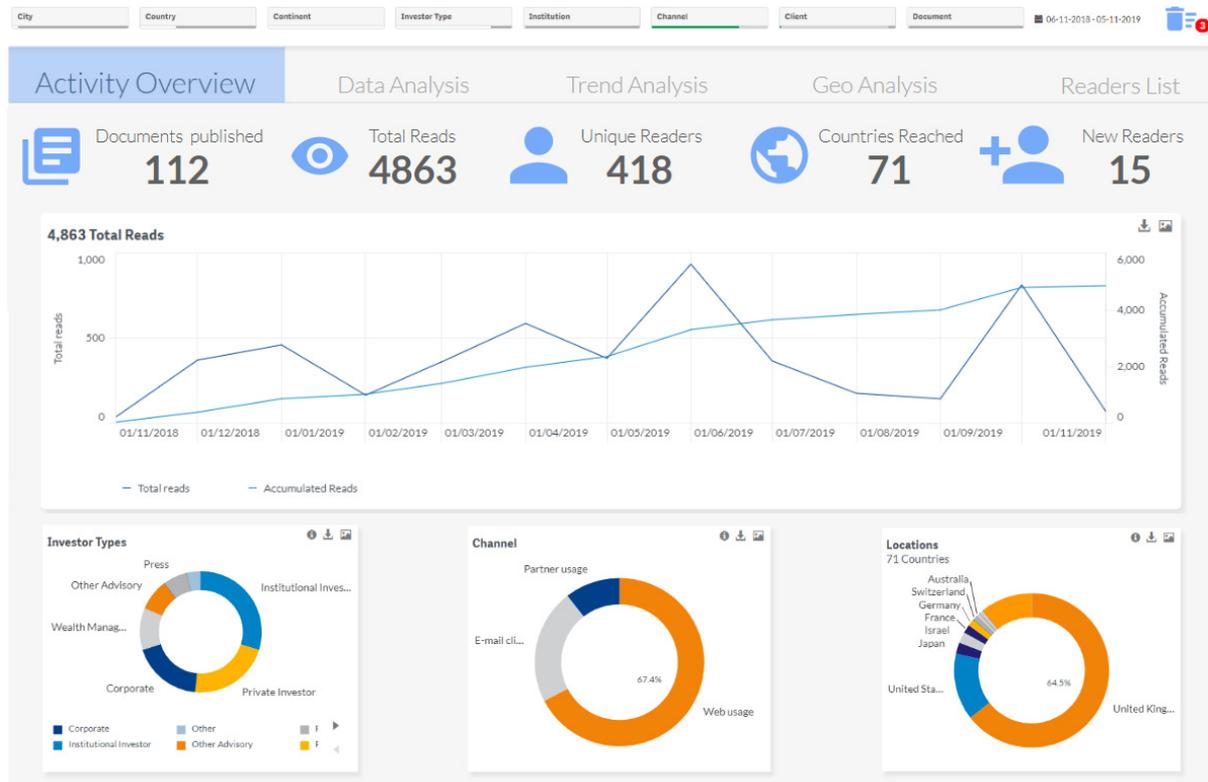


Figura 37. Mashup – Activity Overview

Nella parte alta sono presenti 5 KPI:

- **Reports Published:** numero di reports pubblicati
- **Total Reads:** numero di letture dei reports
- **Unique Readers:** conteggio distinto dei lettori; sono esclusi gli utenti anonimi
- **Countries Reached:** conteggio distinto delle countries dei lettori
- **New Readers:** conteggio distinto dei lettori che hanno letto un solo documento

Nella parte alta della zona di analisi è presente un grafico di trend, che mostra l'andamento del numero di letture nel tempo (per anno-mese); selezionando un mese specifico è possibile visualizzarne l'andamento nei giorni.

Nella parte bassa della zona di analisi sono presenti 3 diagrammi che mostrano le top 5 relative alle seguenti caratteristiche del lettore:

- **Investor Type:** tipologia di investitore (riferito all'azienda del lettore)
- **Channel:** canale di arrivo del lettore
- **Location:** nome della città del lettore

Tutti i grafici sono esportabili sia in formato Excel che come immagini.

4.2. Data Analysis

La pagina di Data Analysis consente di avere maggiori informazioni circa la distribuzione delle letture, in base alle caratteristiche del lettore.

Di seguito uno screen dell'interfaccia:

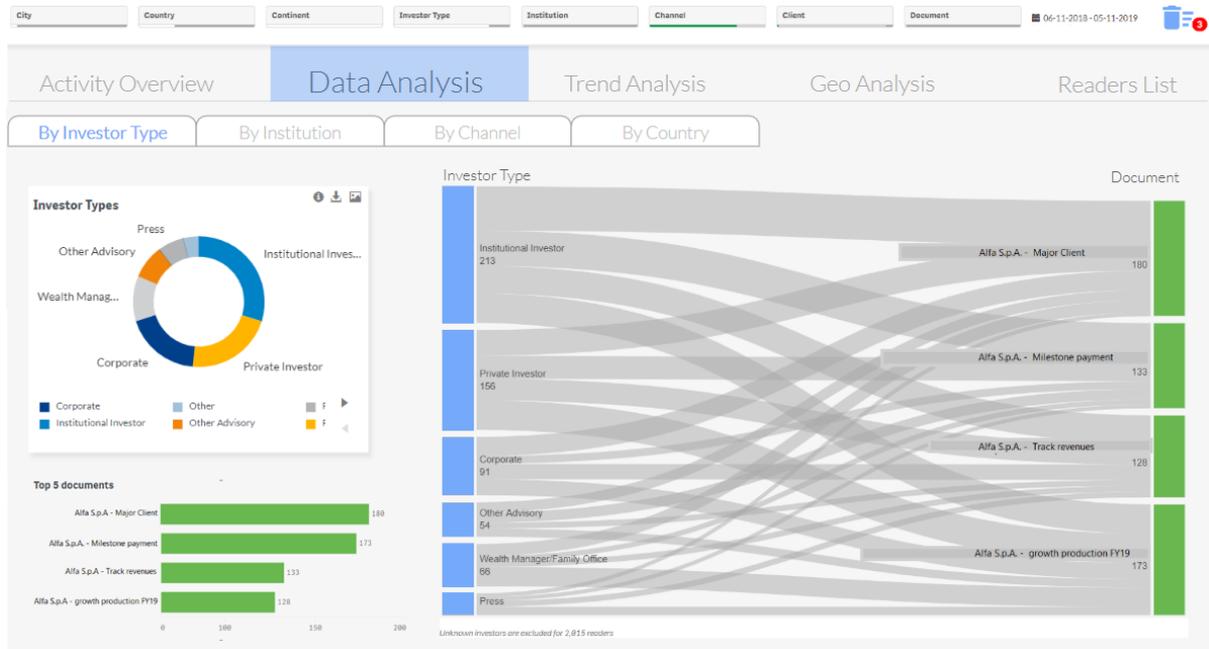


Figura 38. Mashup – Data Analysis

Nel caso presente nell'immagine sono visualizzati i dati per "Investor Type", ma è possibile cambiare la caratteristica di analisi cliccando sulle etichette "By Institution", "By Channel" e "By Country". Indipendentemente dalla caratteristica scelta la tipologia di analisi è la stessa.

In alto a sinistra è presente un diagramma che mostra la suddivisione dei lettori per tipologia di investitore.

Nella parte destra è presente un Sankey chart che permette di avere indicazioni circa la suddivisione delle letture su ciascun documento, da parte dei principali Investor Type.

In basso a sinistra dell'interfaccia, è presente un grafico a barre che mostra il numero di lettori per ciascun documento; quando sono disponibili più di 5 documenti, vengono visualizzati i 5 documenti aventi più letture.

Da notare che originariamente era stata prevista anche una tabella che mostra per ciascun documento il numero di lettori e la ripartizione percentuale delle letture sul totale, ma tale tabella è stato deciso di eliminarla durante le fasi di test, in quanto ritenuta con informazioni ridondanti rispetto a quanto già visualizzato.

Tutti i grafici sono esportabili sia in formato Excel che come immagini.

4.3. Trend Analysis

La pagina Trend Analysis mostra una serie di analisi relative alle letture ed al livello di interesse mostrato dai potenziali lettori, a cui sono state inviate delle e-mail contenenti il collegamento ai report.

In figura è mostrata un'immagine dell'interfaccia:

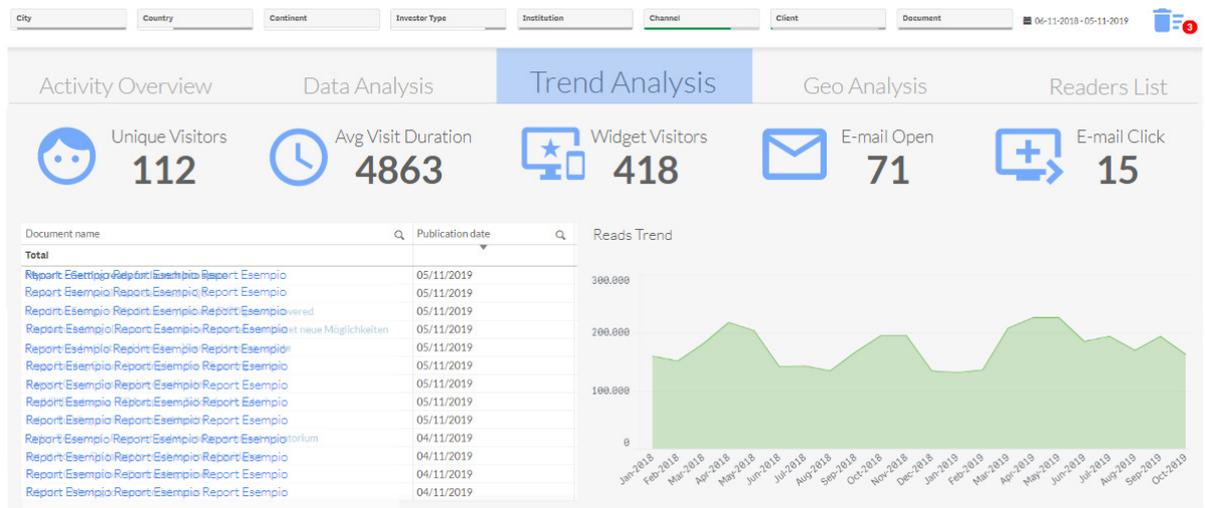


Figura 39. Mashup – Trend Analysis

Sono presenti 3 tipi di analisi:

- Una tabella contenente l'elenco dei reports pubblicati con relativa data di pubblicazione, in cui è possibile visualizzare il report pubblicato facendo click sul titolo dello stesso
- Un grafico lineare contenente l'andamento delle letture nel tempo
- Una serie di KPI relativi alle letture avvenute sul sito di Finance e alle mail inviate da Finance ai potenziali lettori, in particolare:
 - Unique visitors: numero di lettori dei documenti sul Finance website
 - Avg visit duration: media della durata delle letture dei reports
 - Widget visitors: numero di letture avvenute tramite widget
 - E-mail opens: numero di mail aperte dai lettori
 - E-mail click: numero di email in cui è stato aperto il link che rimanda al documento oggetto della mail

Tutti i grafici sono esportabili sia in formato Excel che come immagini. La tabella è esportabile in formato Excel.

4.4. Geo Analysis

La pagina Geo Analysis contiene delle analisi incentrate sulla geografia dei lettori dei documenti.

La figura mostra l'interfaccia della pagina:

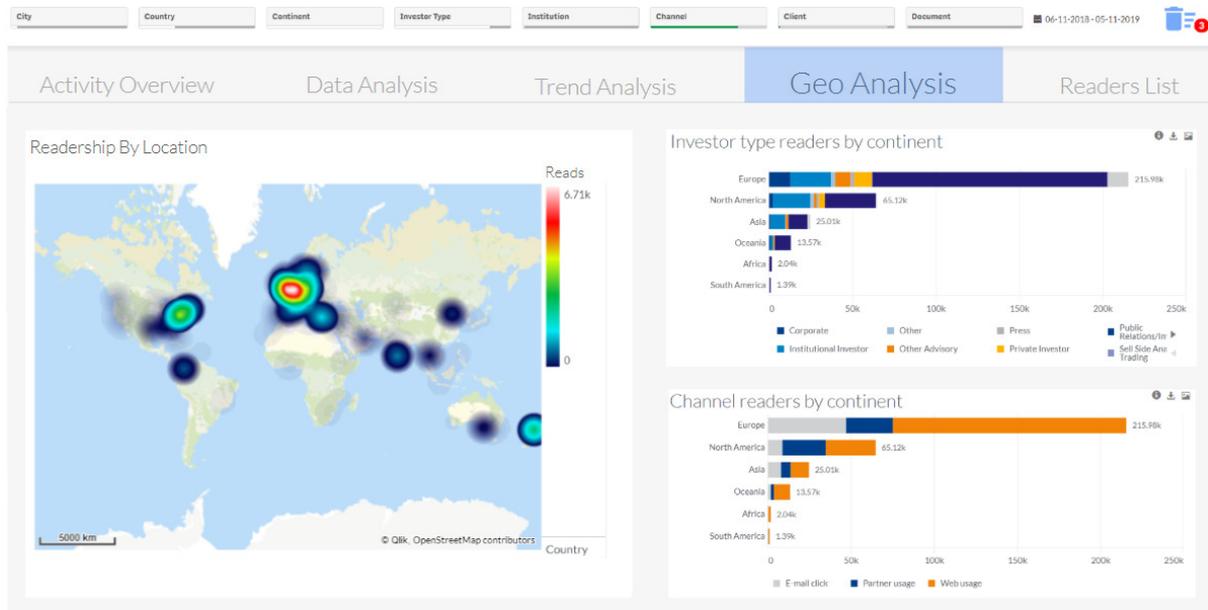


Figura 40. Mashup – Geo Analysis

Di seguito l'elenco delle analisi presenti:

- Readership by Location: mappa di calore che mostra la distribuzione dei lettori nel mondo; è possibile applicare dei filtri geografici direttamente dalla mappa per potersi concentrare su una o più country
- Investor type readers by continent: grafico a barre che permette di suddividere i lettori in base al continente e alla tipologia di investitore di appartenenza
- Channel readers by continent: grafico a barre che permette di suddividere i lettori in base al continente e alla piattaforma cui appartiene la lettura

Tutti i grafici sono esportabili sia in formato Excel che come immagini.

4.5. Readers List

La pagina Readers List mostra una tabella con delle misure relative alle letture per ciascuna istituzione di cui fanno parte i lettori.

La tabella è così composta:

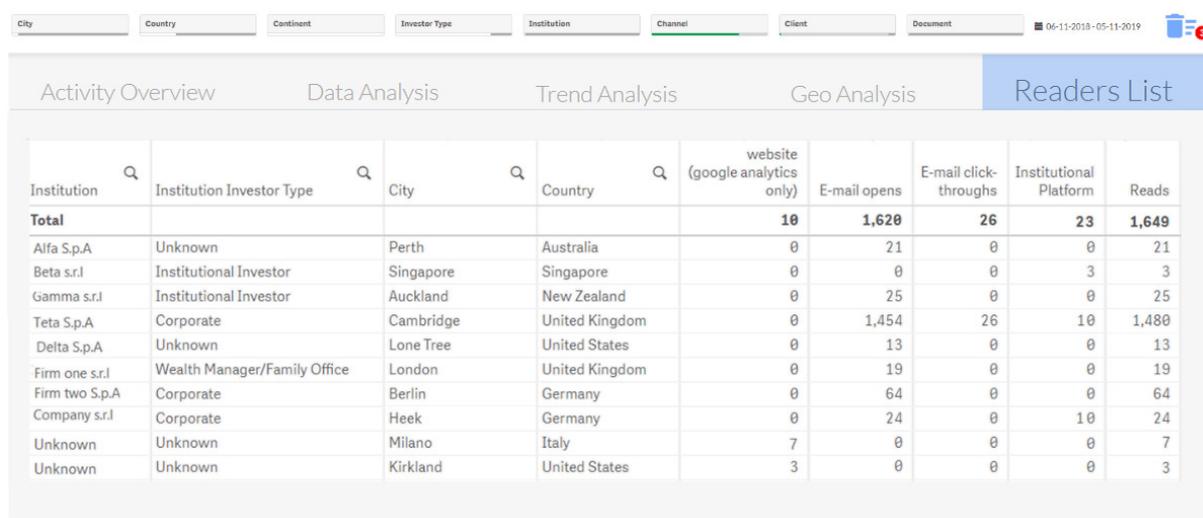
Dimensioni

- Institution
- Institution Investor Type
- City
- Country

Misure

- Website: numero di letture dei documenti avvenute direttamente sul sito di Finance
- E-mail opens: numero di mail aperte dai lettori
- E-mail click throughs: numero di email in cui è stato aperto il link che rimanda al documento oggetto della mail
- Institutional platform: numero di letture avvenute tramite le piattaforme
- Reads: numero totale di letture

In figura l'interfaccia della pagina, esportabile in Excel:



Institution	Institution Investor Type	City	Country	website (google analytics only)	E-mail opens	E-mail click-throughs	Institutional Platform	Reads
Total				10	1,620	26	23	1,649
Alfa S.p.A	Unknown	Perth	Australia	0	21	0	0	21
Beta s.r.l	Institutional Investor	Singapore	Singapore	0	0	0	3	3
Gamma s.r.l	Institutional Investor	Auckland	New Zealand	0	25	0	0	25
Teta S.p.A	Corporate	Cambridge	United Kingdom	0	1,454	26	10	1,480
Delta S.p.A	Unknown	Lone Tree	United States	0	13	0	0	13
Firm one s.r.l	Wealth Manager/Family Office	London	United Kingdom	0	19	0	0	19
Firm two S.p.A	Corporate	Berlin	Germany	0	64	0	0	64
Company s.r.l	Corporate	Heek	Germany	0	24	0	10	24
Unknown	Unknown	Milano	Italy	7	0	0	0	7
Unknown	Unknown	Kirkland	United States	3	0	0	0	3

Figura 41. Mashup – Reader Li

5. Conclusioni

Il forte ambiente collaborativo e l'ottimo lavoro di squadra ci ha permessi di portare a termine gli obiettivi, servendo al cliente uno strumento funzionale, semplice ma anche bello da vedere. Tali requisiti non sono da considerare di minore importanza rispetto agli altri, in quanto trattandosi di una piattaforma di servizio che Finance darà ai suoi clienti, spesso gli users non sono molto esperti di dashboard e reportistica, quindi l'aspetto di semplicità e bellezza sono determinanti per il gradimento della piattaforma.

Il lavoro che ho svolto mi ha permesso di acquisire nuove competenze sulla tecnologia QlikSense, in quanto anche se lavoro sul settore da anni, per la prima volta mi sono approcciata all'estrazione dei dati tramite API, Web Connectors ed FTP/SFTP.

Inoltre la parte di storicizzazione e integrazione dei dati provenienti da numerose sorgenti diverse è stata molto sfidante, e mi ha permesso di constatare come un'attenta analisi preliminare sia utile nella creazione di un progetto strutturato come questo. Ho apprezzato molto anche la parte svolta relativa alla modellazione dei dati geografici per la gestione delle eccezioni nel caso di città o Stati mancanti: l'approccio utilizzato mi ha infatti permesso di comprendere a pieno come funziona la mappatura geografica di Qlik, che solitamente avevo utilizzato solo a livello automatizzato (dunque senza dover apportare modifiche ai dati estratti dalla libreria Geo Qlik).

Il progetto realizzato ha riscosso un buon successo lato cliente, infatti si stiamo già provvedendo a fare altre raccolte requisiti a seguito della richiesta di integrazione di nuovi dati, segno che abbiamo svolto un buon lavoro.

Ringraziamenti

Questa tesi segna la conclusione del mio percorso universitario, un tragitto pieno di emozioni, cadute e rivincite, momenti di gioia e di stress. Tutte queste esperienze mi hanno aiutato molto nella mia crescita, oltre che accademica anche individuale. Grazie all'università sono riuscita a raggiungere degli obiettivi importanti, a crescere come persona e a comprendere che grazie all'impegno e alla costanza è possibile ottenere i risultati desiderati. Studiare e lavorare allo stesso tempo non è stato semplice, ma ad oggi posso dire di essere felice di ciò che sono riuscita ad ottenere.

Volevo ringraziare tutti quelli che mi hanno sostenuto in questo percorso, siete stati tantissimi, in modi e circostanze differenti, mi avete dato la forza di andare avanti passo dopo passo.

Grazie alla mia famiglia, che mi ha sempre sostenuto e mi è sempre stata vicina, esultando per ogni esame superato, indipendentemente dal voto, perché sapeva che ci stavo mettendo tutta me stessa. Grazie per gli incoraggiamenti, per la comprensione per tutte quelle volte che non passavo del tempo con voi "perché dovevo studiare"; grazie per avermi insegnato che il duro lavoro ripaga sempre e per avermi appoggiato nelle mie scelte accademiche. So che questa è anche la vostra vittoria.

Grazie ai miei colleghi universitari incontrati durante gli studi, chi più chi meno, da ognuno di voi ho avuto qualcosa da imparare, uno spunto da prendere sul metodo di studio, la determinazione o il modo di vedere le cose. Grazie in particolar modo ad Angela, che da quando decisi di iniziare a lavorare, mi ha sostenuto ed incoraggiato a non mollare, dimostrando entusiasmo per ogni materia che superavo e aiutandomi nell'inserimento nei lavori di gruppo. Grazie a Lorena, collega di università ma anche di lavoro, che mettendo a disposizione di tutti i suoi appunti, mi ha permesso di limitare lo svantaggio del non seguire le lezioni in aula, riuscendo così a seguire con più facilità il programma trattato dai professori.

Grazie ai miei colleghi di lavoro, vecchi e nuovi, per avermi sempre spronata nell'andare avanti, e per aver sopportato i miei momenti di ansia pre-esame, le mie preoccupazioni per l'inglese, i miei programmi per gli esami futuri.

Grazie ai colleghi del progetto oggetto della tesi, iniziando da Silvia con cui siamo riuscite a lavorare benissimo, accogliendo ognuna le idee dell'altra durante la realizzazione del progetto. Grazie a Loris per il suo modo proattivo di affrontare ogni fase del progetto. Grazie a Diego, per tutto il supporto morale e tecnico che mi ha dato in tutte le fasi del progetto, per avermi aiutato a trovare le soluzioni ai problemi che venivano fuori, per aver affrontato ogni situazione con atteggiamento ottimista e per aver mantenuto il sorriso e la tranquillità anche nei momenti più stressanti.

Grazie a Tiziana, che mi ha dato fin da subito la possibilità di fare la tesi in azienda, suggerendomi questo progetto molto sfidante come argomento di tesi.

Ed infine grazie a Nicolò per tutto quello che hai fatto e continui a fare per me. Grazie per avermi confortato nei momenti più stressanti del mio percorso di studi, anche semplicemente portandomi delle patatine, oppure preparandomi una tisana. Grazie per aver sempre creduto in me, per aver visto in me un potenziale che non mi sentivo, grazie per essermi stato vicino anche quando eravamo distanti. La tua comprensione e la tua costante voglia di farmi ridere, di incoraggiarmi sono state fondamentali. Una buona parte della voglia e della determinazione me l'hai trasmessa tu ogni singolo giorno.

Acronimi e Significati

- App: Applicazione
- BI: Business Intelligence
- BA: Business Analyst
- Dashboard: interfaccia di reportistica
- DSS: Decision Support System
- ETL: Extraction, Transformation, Load
- IT: Information Technology
- MIS: Management Information System
- PM: Project Manager
- QAP: Qlik Analytics Platform
- QMC: Qlik Management Console
- UAT: User Integration Test
- UI: User Interface
- URL: Universal Resource Locator

Bibliografia

- [1] Mollona E. “*Analisi dinamica dei sistemi aziendali*”. Egea, 2000.
- [2] “*BABOK® A guide to use business analysis body of knowledge®*”, International Institute of Business Analysis, Toronto, Ontario, Canada, Version 3.0, 2015
- [3] S. Brobst V. Poe, P. Klaner. “*Building a Data Warehouse for DecisionSupport*”. Prentice-Hall, 1998.
- [4] Paolo Pasini, “*Sistemi Informativi Direzionali*”, Egea, 2004.
- [5] Luca Quagini, “*Business intelligence e knowledge management. Gestione delle informazioni e delle performance nell'era digitale*”, Franco Angeli, 2004.
- [6] M.Golfarelli. New trends in business intelligence. *In proceedings ACM 1th International Conference on Business Intelligence Systems, Opatija, Croatia, 2005.*
- [7] “*QlikView manuale utente*” Versione 11.0 per Microsoft Windows®
Prima Edizione, Lund, Svezia, ottobre 2011, Creato da QlikTech International AB

Sitografia

- [10] <http://www.thebusinessintelligenceguide.com/> ultimo accesso 07/11/2019
- [11] <https://www.chimerarevo.com/guide/windows/sftp-ftps-differenze-221605/> ultimo accesso 07/11/2019
- [12] https://help.qlik.com/en-US/connectors/Subsystems/Web_Connectors_help/Content/Connectors_QWC/Data-Source-Connectors/FTP-Connector.htm ultimo accesso 07/11/2019
- [13] <https://marketingplatform.google.com/intl/it/about/analytics/> ultimo accesso 07/11/2019
- [14] <https://www.qlik.com/it-it/company> ultimo accesso 07/11/2019
- [15] <https://www.qlik.com/it-it/resource-library/3rd-generation-business-intelligence> ultimo accesso 07/11/2019
- [16] https://help.qlik.com/it-IT/sense/June2019/Content/Sense_Helpsites/Tutorials/Tutorials-beginning-basics.htm
- [17] https://help.qlik.com/it-IT/sense/June2019/Subsystems/Hub/Content/Sense_Hub/Scripting/synthetic-keys.htm ultimo accesso 07/11/2019
- [18] https://help.qlik.com/it-IT/sense/June2019/Subsystems/Hub/Content/Sense_Hub/LoadData/understand-circular-references.htm ultimo accesso 07/11/2019
- [19] https://help.qlik.com/en-US/sense-developer/September2019/Subsystems/Mashups/Content/Sense_Mashups/mashups-introduction.htm?l=it-IT ultimo accesso 07/11/2019
- [20] https://help.qlik.com/it-IT/qlikview/April2019/Subsystems/Client/Content/QV_QlikView/Security.htm ultimo accesso 07/11/2019
- [21] https://help.qlik.com/en-US/sense-developer/September2019/Subsystems/Mashups/Content/Sense_Mashups/Howtos/mashups-deploy-scenarios-mashup-normal.htm ultimo accesso 07/11/2019