



# UNIVERSITÀ DI PARMA

Dipartimento di Ingegneria e Architettura

Corso di Laurea in Ingegneria Informatica, Elettronica e delle Telecomunicazioni

Reingegnerizzazione di una Dashboard di Business Intelligence

Reengineering of a Business Intelligence Dashboard

Relatore:

Chiar.mo Prof. Michele Amoretti

Tesi di Laurea di:

Leonardo Minaudo

Correlatore:

Dott. Antonio Calò

ANNO ACCADEMICO 2021/2022

Il seguente elaborato riassume la Tesi derivante dal tirocinio svolto presso l'azienda *Maps Group* che sviluppa soluzioni tecnologiche che trasformano i dati disomogenei e disaggregati in una base di informazioni, e dunque di conoscenza, messa al servizio delle scelte di Operational Business Intelligence. Il tirocinio svolto ha avuto come obiettivo la **reingegnerizzazione della dashboard di Business Intelligence (BI)** dell'applicativo Gzoom. Gzoom, facente parte delle diverse soluzioni che l'azienda Maps Group propone, è un software gestionale open source creato per migliorare risultati, gestire le risorse umane e tenere sotto controllo i risultati ottenuti da amministrazioni, enti e aziende. Nello specifico l'attività svolta durante il tirocinio, la quale è stata il punto di partenza per la suddetta Tesi, ha avuto il compito di migliorare l'architettura e fornire nuove funzionalità alla dashboard, contribuendo a un processo di rimodernizzazione dell'intero applicativo, migliorando la struttura di un sistema Legacy in modo tale da rendere più agevole la futura manutenibilità, aumentare le prestazioni e la velocità d'implementazione di nuove funzionalità.

Le tecnologie e i pattern architetturali scelti per la reingegnerizzazione della dashboard sono stati selezionati in modo tale da creare indipendenza tra le parti dell'applicativo, cosicché da poter scegliere il miglior strumento di sviluppo per ogni specifica applicazione. In particolare, si è scelto di basarsi su un'architettura a **microservizi**, sviluppati con l'utilizzo del framework *Spring Boot*, che comunicano usando **API REST** e si interfacciano tramite una **Single Page Application (SPA)** sviluppata con il framework *Angular*.

La parte di Gzoom reingegnerizzata è basata su due servizi principali uno che opera lato frontend o client e l'altro lato backend o server. Tali servizi, identificati rispettivamente come *Gzoom2Frontend* e *Gzoom2Backend* interagiscono tra loro, come mostrato in Figura 1. *Gzoom2Frontend*, basato sul pattern **MVC (Model - View - Controller)**, si occupa di fornire un'interfaccia grafica all'utente tramite la *view*, attraverso la quale il client può inoltrare richieste al server tramite l'utilizzo del *controller* che è a conoscenza solamente dell'URL identificativo della risorsa richiesta. All'interno dello stesso servizio frontend le informazioni vengono rappresentate da dei *modelli* che facilitano la manipolazione dei dati tra la *view* e il *controller*. *Gzoom2Backend* invece si occupa di ricevere le richieste e fornire delle risposte tramite microservizi; In Figura 1, all'interno del riquadro arancione, sono rappresentati i due servizi sviluppati per la dashboard di BI. Ogni servizio sviluppato all'interno di *Gzoom2Backend* segue il pattern **Controller – Service – Repository** in modo tale da separare i controller REST, la logica di business e il livello che si occupa di dialogare con il database. In quest'ultimo livello sono

stati utilizzati i framework *QueryDSL* e *MyBatis*, che semplificano la costruzione delle query e l'interfacciamento tra il database e gli oggetti in Java. Per il trasferimento dei dati tra le tre parti di ogni servizio viene poi utilizzato un oggetto di trasferimento dati (*DTO – Data Transfer Object*). Infine, per l'inoltro della risposta il contenuto viene formattato in formato JSON e fornito al client che lo gestirà di conseguenza.

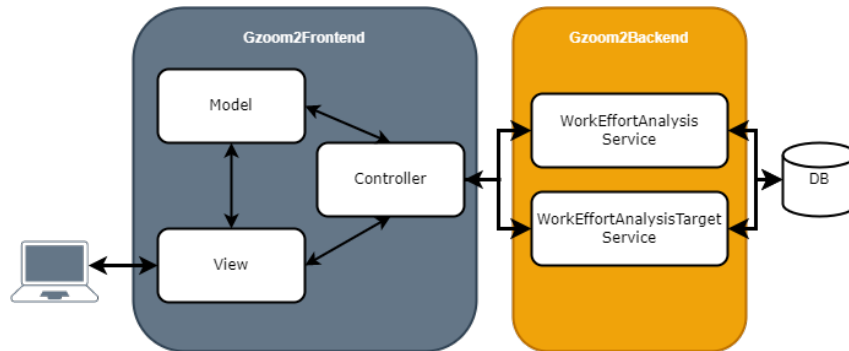


Figura 1: Diagramma dell'interazione client-server

La dashboard di BI realizzata è formata da due schermate principali. Una racchiude la lista delle analisi e l'altra la lista degli obiettivi di una determinata analisi con i relativi valori di risultato. In quest'ultima i dati possono essere ricaricati man mano che si seleziona un obiettivo figlio mostrando quelli che sono gli obiettivi che compongono gerarchicamente il livello inferiore. La schermata della lista degli obiettivi può assumere diverse personalizzazioni grafiche per la visualizzazione degli obiettivi e dei loro risultati. In base a dei parametri preimpostati precedentemente, i risultati possono essere visualizzati tramite l'utilizzo di grafici, tachimetri o emoticon card, in Figura 2 è presente una possibile configurazione. I grafici e i tachimetri sono stati implementati tramite l'utilizzo di librerie aggiuntive JavaScript, rispettivamente *Chart.js* e *ngx-gauge*. In caso in cui la rappresentazione dei risultati avvenga tramite grafici i tipi selezionabili sono: a radar, a barre, a linee, a torta, polare e a ciambella. Inoltre, la dashboard può fornire anche informazioni riguardanti gli indicatori di performance di un obiettivo, indicandone la lista e il loro singolo stato di avanzamento.

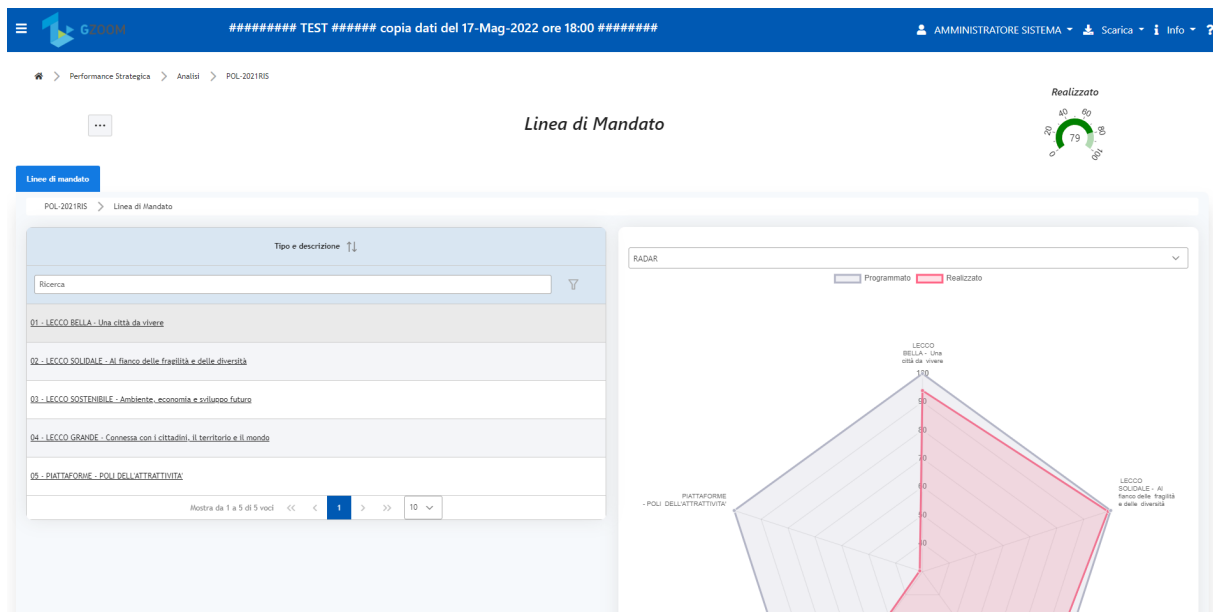


Figura 2: Configurazione possibile della dashboard

Possibili sviluppi futuri sono: (i) l'implementazione di algoritmi basati sull'analisi predittiva che permetterebbe alle organizzazioni di fare previsioni sulle tendenze e sui risultati futuri sulla base di dati storici e (ii) rendere il cruscotto responsive, accessibile da diversi dispositivi.