

POLITECNICO DI TORINO

Corso di Laurea Magistrale in
Ingegneria Gestionale
Tesi di Laurea Magistrale



**Politecnico
di Torino**

Soluzione low code per la pianificazione delle
attività e l'allocazione efficiente delle risorse
aziendali mediante Oracle APEX

Relatore:

Prof.ssa Tania Cerquitelli

Relatore Aziendale:

Morassutto Simone

Candidato:

Francesca Saccotelli

Anno accademico 2023/2024

Sommario

Il presente elaborato di tesi si propone di fornire un'approfondita trattazione sull'applicazione realizzata per l'azienda Mediamente Consulting S.r.l, sede di tirocinio, volta a migliorare la gestione dei progetti e delle risorse all'interno dell'organizzazione. Nell'attuale scenario gestionale e operativo delle imprese, infatti, tali aspetti risultano di fondamentale importanza per consolidare la propria presenza in un contesto aziendale sempre più competitivo.

L'applicazione MM Macropianificazione nasce dalla necessità dell'azienda di sostituire diversi fogli di calcolo, utilizzati per tenere traccia delle attività e delle risorse aziendali. Per soddisfare tale requisito, si è reso necessario sviluppare uno strumento in grado di strutturare, in modo organizzato e completo, le informazioni all'interno di una singola fonte centralizzata, garantendo così l'integrità, la coerenza e la sicurezza dei dati.

La descrizione del progetto parte dall'illustrazione del contesto aziendale e dall'analisi dei requisiti, in cui viene specificato anche a chi è rivolto. Vengono poi esaminati la base dati e il funzionamento dell'applicazione, affrontando nel dettaglio le pagine realizzate. Queste includono:

- una pagina dedicata alla pianificazione delle attività;
- tre pagine che presentano report riassuntivi legati alle attività pianificate e alle tecnologie utilizzate;
- una pagina dedicata alla gestione delle risorse;
- una pagina contenente un elenco delle tecnologie utilizzate dall'azienda e che offre la possibilità di integrarne di nuove;
- una pagina dedicata all'allocazione delle risorse pertinenti alle attività pianificate, in cui è possibile visualizzare i risultati attraverso due tipologie di report.

Uno degli elementi caratterizzanti di tale applicazione è l'algoritmo elaborato per automatizzare l'allocazione delle risorse, il cui obiettivo è ottimizzare tale processo e garantire una distribuzione ottimale delle risorse che soddisfi le particolari esigenze di ogni attività. Nello specifico, tale algoritmo, sfruttando una serie di logiche e regole di allocazione, permette di determinare in modo efficiente quali risorse assegnare, tenendo conto di alcuni vincoli chiave, tra cui il livello di seniority delle risorse coinvolte, la loro disponibilità e le tecnologie da loro utilizzate.

L'applicazione è stata realizzata mediante la piattaforma Oracle Application Express (APEX), uno dei principali strumenti low-code per lo sviluppo di applicazioni web e mobile nel settore enterprise. APEX rende semplice e rapida la realizzazione di applicazioni scalabili, sicure, dotate di una ricca interfaccia utente, richiedendo principalmente familiarità con i linguaggi SQL e PL/SQL e senza la necessità di possedere competenze avanzate nello sviluppo web.

La presente tesi si propone inizialmente di esplorare il concetto di sviluppo low-code e successivamente di approfondire l'utilizzo di Oracle APEX come piattaforma di sviluppo.

Nel dettaglio, il primo capitolo analizza le principali caratteristiche del paradigma low-code, insieme ai vantaggi comuni e ai suoi tratti distintivi rispetto allo sviluppo no-code. In seguito, viene effettuata un'analisi del mercato low-code e dei principali attori di questo settore, per concludere con alcune osservazioni sul futuro di questo approccio innovativo allo sviluppo software.

Nel secondo capitolo viene ampiamente esaminata la piattaforma Oracle APEX, esplorando la sua architettura three-tier, i componenti dell'ambiente di sviluppo, il processo di creazione delle applicazioni tramite tale tool e i punti di forza dello strumento. Sono, inoltre, illustrati i principali casi d'uso e la roadmap generale per le future release di Oracle APEX.

Attraverso queste analisi, vengono poste le basi per la successiva esposizione dell'applicazione sviluppata, a cui segue un ultimo capitolo di conclusioni e potenziali evoluzioni future della stessa.

A te nonna,
che ci sei sempre stata
e per sempre ci sarai.

Indice

| | |
|---|----|
| Elenco delle figure | 6 |
| 1 Lo sviluppo Low-Code | 9 |
| 1.1 Il concetto di Low-Code | 9 |
| 1.1.1 Confronto tra sviluppo Low-Code e No-Code | 10 |
| 1.2 Il mercato del Low-Code | 14 |
| 1.2.1 Gli attori del mercato Low-Code | 16 |
| 1.3 Il futuro del Low-Code | 20 |
| 2 Oracle APEX | 23 |
| 2.1 Introduzione ad Oracle Apex | 23 |
| 2.2 L'architettura di Oracle APEX | 24 |
| 2.3 Oracle Application Express Environment | 26 |
| 2.3.1 App Builder | 28 |
| 2.3.2 SQL Workshop | 29 |
| 2.3.3 Team Development | 32 |
| 2.3.4 Gallery | 32 |
| 2.4 Creare applicazioni in Oracle APEX | 33 |
| 2.4.1 Page Designer | 35 |
| 2.5 Vantaggi di Oracle APEX | 37 |
| 2.5.1 Integrazione di Oracle APEX nel database Oracle | 38 |
| 2.6 Casi d'uso di Oracle APEX | 40 |
| 2.6.1 Nuove Applicazioni Cloud | 40 |
| 2.6.2 Sostituzione dei fogli di calcolo | 40 |

| | | |
|----------|---|-----------|
| 2.6.3 | Condivisione di dati esterni | 41 |
| 2.6.4 | Modernizzazione di Oracle Forms | 41 |
| 2.6.5 | Report interattivi | 42 |
| 2.6.6 | Integrazione a sistemi ERP esistenti | 42 |
| 2.7 | Prospettive future di Oracle APEX | 43 |
| 3 | MM Macropianificazione | 45 |
| 3.1 | Il Contesto | 46 |
| 3.2 | Analisi dei requisiti | 46 |
| 3.2.1 | Requisiti aggiuntivi in corso di progetto | 47 |
| 3.3 | Base dati dell'applicazione | 48 |
| 3.4 | Funzionamento dell'applicazione | 51 |
| 3.4.1 | Home Page Macropianificazione | 52 |
| 3.4.2 | Macropianificazione Overviews | 61 |
| 3.4.3 | Info Risorse | 65 |
| 3.4.4 | Add Skill | 69 |
| 3.5 | Algoritmo di allocazione delle risorse | 70 |
| 3.5.1 | Allocazione Risorse | 76 |
| 4 | Conclusioni e Sviluppi futuri | 83 |
| 4.1 | Sviluppi futuri dell'applicazione | 85 |
| | Bibliografia | 87 |

Elenco delle figure

| | | |
|------|--|----|
| 1.1 | Vantaggi Low-Code e No-Code. | 11 |
| 1.2 | Differenze tra Low-Code e No-Code. | 13 |
| 1.3 | Magic Quadrant di Gartner 2023 [1] | 16 |
| 1.4 | Criteri di valutazione Magic Quadrant. | 17 |
| 2.1 | Architettura three-tier Oracle APEX. [2] | 25 |
| 2.2 | Oracle APEX Environment. [3] | 26 |
| 2.3 | Oracle APEX Workspace. | 27 |
| 2.4 | App Builder. | 28 |
| 2.5 | SQL Workshop. | 29 |
| 2.6 | Object Browser. | 30 |
| 2.7 | SQL Commands. | 30 |
| 2.8 | Esempio Quick SQL. | 31 |
| 2.9 | Gallery. | 32 |
| 2.10 | Creare un'applicazione in Oracle APEX. | 34 |
| 2.11 | Page Designer. | 36 |
| 3.1 | Login MM Macropianificazione. | 52 |
| 3.2 | Home Page Macropianificazione. | 52 |
| 3.3 | Form Home Page Macropianificazione. | 53 |
| 3.4 | Tabella T_MACROPIANIFICAZIONE. | 55 |
| 3.5 | Delete Macropianificazione. | 55 |
| 3.6 | Popup LOV Skill. | 57 |
| 3.7 | Select List con Static Values SENIORITY. | 59 |
| 3.8 | Highlight Giorni Da Allocare. | 60 |

| | | |
|------|---|----|
| 3.9 | Macropianificazione Overview. | 61 |
| 3.10 | Funzione <i>get_query_macropianificazione_pivot()</i> | 62 |
| 3.11 | Impostazioni per la gestione dinamica dei mesi. | 63 |
| 3.12 | Skill Overview. | 64 |
| 3.13 | Funzione <i>get_query_skill_pivot()</i> | 65 |
| 3.14 | Info Risorse. | 65 |
| 3.15 | Query Report e Form <i>Info Risorse</i> | 66 |
| 3.16 | Process Form Edit Info Risorse. | 67 |
| 3.17 | Delete Info Risorse. | 68 |
| 3.18 | Update Info Risorse. | 68 |
| 3.19 | Dichiarazione e inizializzazione delle variabili. | 70 |
| 3.20 | Aggiornamento. | 71 |
| 3.21 | Selezione delle attività e azzeramento dei contatori. | 72 |
| 3.22 | Calcolo del numero di risorse. | 73 |
| 3.23 | Allocazione con carico di lavoro non uniforme. | 75 |
| 3.24 | Allocazione con carico di lavoro uniforme. | 76 |
| 3.25 | Allocazione Risorse. | 76 |
| 3.26 | Dynamic Actions. | 78 |
| 3.27 | Pivot Allocazione Risorse. | 80 |
| 3.28 | Impostazioni Report Pivot. | 81 |

Capitolo 1

Lo sviluppo Low-Code

1.1 Il concetto di Low-Code

Nel contesto dell'attuale era digitale, l'innovazione tecnologica e lo sviluppo software sono diventati di fondamentale importanza per il funzionamento e il successo delle imprese in tutti i settori.

Tradizionalmente, la creazione di software prevedeva tempi di sviluppo elevati, risorse significative e una conoscenza approfondita di specifici linguaggi di programmazione. Tale approccio poteva costituire un problema per molte organizzazioni, che si trovavano, ad esempio, a dover far fronte a risorse limitate o ad un'impellente necessità di rispondere rapidamente alle esigenze del mercato.

In risposta a queste sfide, è stato proposto un innovativo paradigma dello sviluppo software: lo sviluppo low-code. Il low-code è una metodologia di sviluppo delle applicazioni che, sfruttando strumenti orientati alla programmazione visuale, riduce notevolmente la necessità di scrivere codice linea per linea. Quest'ultima pratica è tipica della programmazione tradizionale, che tuttavia non viene sostituita dal low-code, bensì si integra con essa, al fine di sfruttare i vantaggi di entrambe.

Alla base del low-code vi è l'idea di automatizzare quanto più possibile la realizzazione di un'applicazione. Tale idea diventa realtà nel momento in cui gli sviluppatori utilizzano piattaforme capaci di automatizzare gran parte del lavoro, andando a sostituire sempre di più la codifica manuale. Al giorno d'oggi, tutto ciò viene enfatizzato dall'Intelligenza Artificiale, utilizzata come supporto alla programmazione.

Il vantaggio principale di questo approccio risiede nel fatto che, disponendo di codice “pronto all’uso”, gli sviluppatori possono intervenire nella scrittura del codice solo per quanto riguarda le modifiche e le integrazioni relative alle esigenze specifiche di ogni progetto.

Le piattaforme low-code offrono un’interfaccia visuale intuitiva per gestire i blocchi funzionali dell’applicazione; ciò semplifica notevolmente il processo di sviluppo, consentendo ad una vasta gamma di utenti, anche non esperti, di parteciparvi. Inoltre, ne traggono beneficio anche gli sviluppatori professionisti che, liberati dalle attività ripetitive tipiche dell’approccio tradizionale, possono concentrarsi su compiti ad alto valore aggiunto e gestire contemporaneamente più progetti.

Lo sviluppo low-code rappresenta un’importante svolta nel modo in cui vengono create e gestite le applicazioni e i suoi benefici coinvolgono non solo gli sviluppatori, ma anche le aziende stesse. Dati i numerosi vantaggi offerti da questa tipologia di sviluppo, si stima che in futuro sempre più utenti si orienteranno a soluzioni di tipo low-code, promuovendo così una significativa trasformazione nell’ambito dello sviluppo software. [4]

1.1.1 Confronto tra sviluppo Low-Code e No-Code

Data la crescente domanda di automazione e modernizzazione nel settore dell’Information Technology (IT) e la volontà da parte delle imprese di mantenere la propria competitività in rapporto al time-to-market, è emerso un nuovo approccio nello sviluppo software, noto come no-code, che si propone come alternativa al low-code. Il no-code rappresenta un approccio aggiuntivo nell’ambito del Rapid Application Development (RAD) rispetto al low-code. Si distingue da quest’ultimo per il fatto che non richiede alcun tipo di codifica o conoscenza tecnica, bensì la creazione di software si basa esclusivamente su strumenti visivi, come le interfacce drag-and-drop o point-and-click. La caratteristica principale delle piattaforme no-code consiste nel fatto che possono essere utilizzate non solo da sviluppatori esperti, ma anche da utenti che non possiedono alcun tipo di conoscenza nel campo della programmazione.

Gli ambiti adatti allo sviluppo no-code includono le applicazioni self-service destinate agli utenti aziendali, le dashboard e le applicazioni web e mobile. Questo

paradigma si dimostra particolarmente idoneo alla rapida creazione di applicazioni standalone, all'implementazione di interfacce utenti intuitive e all'automazione di processi semplici. Inoltre, viene utilizzato in diversi tipi di strumenti e applicazioni specifiche; in particolare negli strumenti di pianificazione del calendario e di gestione delle risorse e nelle applicazioni di BI reporting, con colonne e filtri configurabili. [5] [6]

VANTAGGI COMUNI AI DUE APPROCCI

Gli approcci low-code e no-code presentano diversi aspetti in comune; ad esempio, si avvalgono entrambi di interfacce visive e modelli predefiniti per sostituire la codifica tradizionale. Inoltre, le piattaforme di sviluppo sono offerte come soluzioni basate sul cloud (PaaS) e si caratterizzano per l'adozione di un design orientato al flusso di lavoro, finalizzato a definire la progressione logica dei dati. Per tali analogie, condividono una serie di vantaggi, i quali possono essere sintetizzati nella seguente figura: [5] [6]

| VANTAGGIO | DESCRIZIONE |
|---|---|
| Maggiore accessibilità allo sviluppo software | Non necessitano di sviluppatori altamente specializzati |
| Aumento della produttività | Aumentano la velocità di sviluppo, riducendo le tempistiche dei progetti e favorendo un lancio più rapido dei prodotti |
| Facilità di prototipazione | Gli sviluppatori possono creare prototipi facili da costruire, in modo da poter ottenere feedback dai clienti anticipati e riducendo, quindi, costi e rischi legati alla decisione di procedere o meno nei progetti |
| Elevata possibilità di personalizzazione | Grazie alla loro semplicità di personalizzazione interna, favoriscono la creazione di un'applicazione, piuttosto che l'acquisto di un prodotto pronto all'uso |
| Coerenza architetturale | Le piattaforme centralizzate garantiscono uniformità nel design e nel codice, semplificando i moduli più comuni come logging, audit e debug delle applicazioni |
| Convenienza economica | Richiedono meno risorse, costi di infrastruttura e di manutenzione minori e, grazie ad aggiornamenti più rapidi, permettono di ottenere un ROI migliore |
| Interconnessione tra l'ambito business e quello IT | La maggior partecipazione di utenti aziendali allo sviluppo contribuisce a generare maggior equilibrio tra gli ambiti business e IT |

Figura 1.1. Vantaggi Low-Code e No-Code.

DIFFERENZE TRA I DUE APPROCCI

Sebbene le metodologie low-code e no-code abbiano obiettivi simili, è possibile individuare diversi aspetti per i quali si distinguono. La scelta su quale approccio utilizzare non è standard e deve essere stabilita in base alle condizioni e alle esigenze del contesto specifico.

La prima grande differenza tra i due riguarda il target a cui si rivolgono tali paradigmi. Il low-code, infatti, si rivolge ad un pool elevato di sviluppatori, esperti e non. In particolare, con tale metodologia viene semplificato il lavoro degli sviluppatori professionisti che, grazie a strumenti che automatizzano gli aspetti standard della codifica, possono concentrarsi su compiti più complessi. Il no-code, invece, è progettato per gli utenti di business che non possiedono competenze di programmazione, bensì hanno una vasta conoscenza di dominio. Quest'ultimo è anche adatto a team ibridi composti da utenti aziendali e sviluppatori software, nonché per piccole imprese e team non IT.

Altrettanto significative sono le differenze tra low-code e no-code per quanto riguarda i casi d'uso. Il no-code viene utilizzato per la creazione rapida di applicazioni front-end attraverso interfacce intuitive di tipo drag-and-drop. È molto utilizzato anche per applicazioni interne prive di funzionalità estese o di piccole dimensioni e per sostituire attività amministrative ripetitive come i report basati su Excel, spesso trascurate dagli IT, ma cruciali per i team aziendali. Il low-code, invece, è adatto ad applicazioni con logiche di business complesse, grazie ad una vasta libreria di componenti ed è inoltre preferibile al no-code per l'integrazione con altre applicazioni e API esterne, per la connessione a più fonti di dati e per la creazione di sistemi, in cui il tema della sicurezza può richiedere l'intervento dell'IT.

Un altro aspetto importante da considerare è la velocità di sviluppo. Il low-code richiede più tempo e studio per essere appreso e implementato, tuttavia garantendo ampio spazio alla personalizzazione. Al contrario, il no-code è altamente configurabile e plug-and-play, richiede meno tempo per lo sviluppo e i tempi di test sono ridotti, grazie alla minore probabilità di errori.

Il low-code si configura come sistema aperto, che consente agli utenti di estendere le funzionalità attraverso il codice, mentre il no-code è un sistema maggiormente

chiuso, che può essere esteso solo tramite set di funzionalità predefinite. Di conseguenza, il low-code offre maggiore flessibilità e riutilizzo del codice, un numero maggiore di casi d’uso, ma gli aggiornamenti e le patch devono essere gestiti in maniera più attenta, al fine di garantire la compatibilità con il codice implementato manualmente.

Nel contesto del low-code e no-code è rilevante inoltre considerare i rischi che possono essere associati all’adozione di tali metodologie, in particolare il rischio di Shadow IT. Entrambi gli approcci possono portare a infrastrutture parallele non attentamente monitorate, con problemi di sicurezza e debiti tecnici; tuttavia, il rischio si presenta maggiormente nel caso del no-code, in quanto l’intervento dei team IT è limitato.

Infine, è possibile individuare un ultimo elemento di differenziazione nel range architetturale. Il low-code supporta la scalabilità e la compatibilità cross-platform e, grazie a plugin e codice personalizzati, agevola la realizzazione di una vasta gamma di soluzioni e la possibilità di lavorare con più piattaforme. Il no-code, invece, presenta una scalabilità ridotta ed è rivolto ad un numero ristretto di casi d’uso; ciò è dovuto ad una minore capacità di estensione e ad un potenziale limitato nella connessione a sistemi legacy o nell’integrazione con altre piattaforme. Le differenze tra i due approcci sono riassunte in figura. [5] [6]

| DIFFERENZA | LOW-CODE | NO-CODE |
|-----------------------------|--|--|
| Target degli utenti | Sviluppatori professionisti e non | Utenti di business |
| Casi d’uso | Applicazioni con logiche di business complesse, integrazione con altre applicazioni e API esterne, connessione a più fonti dati, applicazioni in cui deve essere garantita sicurezza | Applicazioni di rapida creazione, applicazioni aziendali interne, sostituzione di report basati su Excel |
| Velocità di sviluppo | Tempi di apprendimento e di sviluppo più elevati | Tempo di sviluppo e di test ridotti |
| Tipologia di sistema | Aperto | Chiuso |
| Rischio di Shadow IT | Rischio contenuto | Rischio elevato |
| Range architetturale | Scalabilità elevata, compatibilità cross-platform | Capacità di estensione limitata, potenziale ridotto nella connessione a sistemi legacy e nell’integrazione con altre piattaforme |

Figura 1.2. Differenze tra Low-Code e No-Code.

1.2 Il mercato del Low-Code

Negli ultimi anni, il mercato delle piattaforme di sviluppo low-code ha sperimentato una significativa trasformazione e una notevole crescita, attribuibile ad una domanda sempre più elevata di soluzioni di sviluppo software agili ed efficienti.

L'espansione di tale mercato è stata motivata da diversi elementi fondamentali, tra cui l'esigenza crescente di distribuire rapidamente le applicazioni, la difficoltà nel reperire sviluppatori qualificati e la tendenza alla digitalizzazione dei vari settori.

Un fattore chiave che ha contribuito alla rapida adozione del low-code è stata la pressione, sempre maggiore sulle aziende, ad adattarsi velocemente ai cambiamenti del mercato. Ciò a cui le aziende ambiscono, infatti, è mantenere il passo con le mutevoli dinamiche di settore e con le crescenti aspettative dei clienti, una sfida che risulta difficile da affrontare con i prolungati tempi di sviluppo e il lavoro di codifica richiesto dalle metodologie tradizionali. Un ulteriore aspetto che ha condotto alla diffusione delle piattaforme low-code è stata la carenza di sviluppatori professionisti, che ha reso difficile per le aziende attirare e trattenere programmatori qualificati.

Il mercato delle piattaforme di sviluppo low-code è caratterizzato da una vasta gamma di fornitori che offrono diverse funzionalità e capacità. Ogni piattaforma mette a disposizione diversi strumenti per lo sviluppo, la costruzione, il rilascio e la gestione delle applicazioni, in modo tale da poter soddisfare le esigenze e le preferenze degli utenti. Tra i principali attori del settore vi sono Microsoft, Salesforce, OutSystems, Mendix, ServiceNow e Appian.

Con l'aumento della consapevolezza sulle caratteristiche vantaggiose di cooperazione, velocità e agilità nello sviluppo software, si prevede che sempre più aziende si avvicineranno all'utilizzo di piattaforme di tipo low-code. [7]

Secondo un'analisi condotta da Gartner (società americana che si occupa di consulenza strategica, ricerca di mercato e analisi nell'ambito della tecnologia dell'informazione [8]), sulle tecnologie emergenti, infatti, il low-code viene identificato come una delle tecnologie più influenti per le imprese nei prossimi anni. La società di analisi stima che, a partire dal 2024, gli sviluppatori al di fuori dei dipartimenti IT costituiranno circa l'80% degli utilizzatori delle tecnologie low-code, rispetto al 60% circa registrato nel 2021. Si prevede inoltre che, nel corso del 2024, le funzionalità

di iperautomazione diventeranno un elemento chiave nel contesto della concorrenza per queste piattaforme. A partire dal 2027, invece, la visione strategica delineata da Gartner indica che almeno la metà degli investimenti effettuati dai fornitori di piattaforme low-code si concentrerà sull'integrazione di PBC (Packaged Business Capabilities), ovvero soluzioni software con funzionalità di processo integrate, dedicate a specifiche aree e attività aziendali. Gli investimenti futuri saranno anche diretti sempre di più verso lo sviluppo di soluzioni che hanno come obiettivo quello di semplificare i processi aziendali, così da renderli maggiormente accessibili ad utenti meno esperti. In particolare, tali soluzioni includeranno interfacce utenti sempre più intuitive, logiche di business integrate, workflow guidati e dati a supporto dei processi aziendali. Ciò consentirà di automatizzare i meccanismi decisionali e di rispondere alla crescente complessità delle architetture applicative in un contesto sempre più orientato ad un “business” digitale e al cloud. [9]

1.2.1 Gli attori del mercato Low-Code

Il Magic Quadrant di Gartner è un noto rapporto nel campo dell'analisi del mercato tecnologico. Viene redatto annualmente da tale società e il suo obiettivo è quello di fornire una panoramica dello stato del mercato per una determinata tecnologia o settore.



Figura 1.3. Magic Quadrant di Gartner 2023 [1]

Nel caso specifico, il Magic Quadrant relativo alle LCAP (Low-Code Application Platforms) analizza e valuta diverse piattaforme di sviluppo low-code. Ciò rappresenta un prezioso strumento per gli appassionati di tale metodologia, che sono dunque aiutati nel valutare e prendere decisioni informate sulle tecnologie da utilizzare per lo sviluppo delle proprie applicazioni.

Il Magic Quadrant è un grafico costituito da quattro quadranti, in cui le piattaforme vengono posizionate in base alle loro prestazioni rispetto a due metriche: la completezza della visione (Completeness Of Vision) e la capacità di esecuzione

(Ability To Execute). Tali metriche prendono in considerazione diversi elementi che, in base alla loro importanza, contribuiscono al posizionamento delle aziende nei quadranti. I criteri di valutazione per i due fattori sono riassunti in figura:

| COMPLETENESS OF VISION | | ABILITY TO EXECUTE | |
|-----------------------------|-----------|------------------------------|-----------|
| EVALUATION CRITERIA | WEIGHTING | EVALUATION CRITERIA | WEIGHTING |
| Market Understanding | High | Product or Service | High |
| Marketing Strategy | Medium | Overall Viability | Medium |
| Sales Strategy | Medium | Sales Execution/Pricing | High |
| Offering (Product) Strategy | High | Market Responsiveness/Record | Medium |
| Business Model | Medium | Marketing Execution | Low |
| Vertical/Industry Strategy | Low | Customer Experience | Medium |
| Innovation | High | Operations | Low |
| Geographic Strategy | Low | | |

Figura 1.4. Criteri di valutazione Magic Quadrant.

Le quattro aree del quadrante suddividono i venditori e le loro piattaforme in quattro categorie, etichettate come *Leaders*, *Challengers*, *Visionaries* e *Niche Players*. Alla categoria dei *Leaders* appartengono quelle imprese, le cui piattaforme hanno una presenza globale e sono considerate le migliori nel loro settore. Tali aziende si distinguono sia per la loro grande visione del prodotto, sia per la loro forte capacità di esecuzione.

Tra i *Leaders* troviamo in cima Mendix, prodotto creato dall'azienda omonima, che offre una solida piattaforma low-code con frequenti aggiornamenti alle funzionalità. Si tratta di uno strumento ottimale per lo sviluppo di app mobile native, per la creazione della logica del sistema senza richiedere codifica, per le integrazioni e funzionalità messe a disposizione, nonché per le sue avanzate caratteristiche di sviluppo assistito dall'Intelligenza Artificiale.

OutSystems è un'ulteriore piattaforma low-code di alto livello, che permette di realizzare software di diversa complessità, fornendo strumenti avanzati e continui miglioramenti nel campo dell'Intelligenza Artificiale. Tuttavia, potrebbe non risultare la scelta ottimale per risolvere problemi settoriali specifici.

Power Apps di Microsoft è una piattaforma low-code che si rivolge sia ad utenti che non possiedono conoscenze nell'ambito della programmazione, sia a sviluppatori esperti. Offre diverse integrazioni e strumenti, quali ad esempio Dataverse, AI

Builder, Power Pages e Power Automate. La sua peculiarità consiste nella possibilità di accedere ad alcune sue funzionalità di base tramite piani aziendali di Microsoft 365 o Dynamic 365, con pieno supporto da parte di Microsoft e numerose estensioni disponibili su AppSource.

Salesforce fornisce numerosi strumenti low-code e di sviluppo software generale, che variano a livello di complessità e uso. In termini di innovazione, non rappresenta una soluzione all'avanguardia, ma si tratta di una piattaforma che si aggiorna costantemente, con un ampio ecosistema e una comunità attiva. La piattaforma si focalizza sulle operazioni aziendali legate ai clienti e può essere eseguita su server Salesforce o AWS (Amazon Web Services, disponibile tramite Salesforce Hyperforce).

Gli ultimi due *Leaders* individuati da Gartner sono ServiceNow e Appian. Entrambe le piattaforme vantano un ecosistema robusto; in particolare, ServiceNow con il suo servizio App Engine, parte della Now Platform, fornisce funzionalità per automatizzare i processi aziendali e i flussi di lavoro ed è un'ottima soluzione per le grandi aziende che utilizzano già gli strumenti SaaS (Software as a service) dell'azienda, mentre Appian, pur avendo costi di licenza elevati, garantisce la gestione di un elevato numero di utenti e offre avanzate funzionalità di logica aziendale e di flusso di lavoro, assicurando una notevole scalabilità.

I *Challengers* sono abili nell'implementazione dei loro progetti, ma mancano di una visione sufficientemente forte per il loro prodotto. Questo limite potrebbe rendere difficile ampliarsi ad una gamma diversificata di clienti, raggiungendo solo quelli appartenenti al loro parco clienti. Il quadrante dei Challengers include: Oracle, Retool, HCLSoftware e Zoho.

La soluzione proposta da Oracle è Oracle Application Express (APEX), trattato in maniera approfondita nel successivo capitolo. Oracle APEX è una solida piattaforma low-code gratuita per gli utenti di Oracle Database, con supporto e assistenza da parte di un'azienda considerata leader nel settore IT. Tale tool possiede ottime capacità di gestione dati, reportistica e sicurezza, sebbene non disponga di un supporto nativo per dispositivi mobili. Tuttavia, per molte aziende, in particolare per quelle che già utilizzano un database Oracle, APEX rimane la scelta migliore.

Retool è una piattaforma low-code rivolta principalmente a sviluppatori esperti,

che possiede un'interfaccia drag-and-drop intuitiva (personalizzabile ulteriormente utilizzando JavaScript, SQL, HTML e CSS) e ampie capacità di integrazione con diversi database e API.

Volt MX di HCLSoftware integra il low-code con piattaforme di sviluppo multi-esperienza MXDP (MultieXperience Development Platform) e mette a disposizione diversi strumenti per lo sviluppo front-end e back-end, con disponibilità sia in locale, sia su cloud AWS.

Zoho Creator fornisce una suite di funzionalità aziendali pronte all'uso, viene spesso utilizzato per il miglioramento delle soluzioni Zoho esistenti, ma anche per lo sviluppo di applicazioni multi-esperienza. Inoltre, supporta diversi linguaggi di programmazione, come JavaScript, Java, Node.js e XPath, offrendone anche uno proprio, noto come Deluge.

Le aziende appartenenti alla categoria dei *Visionaries* possiedono idee ambiziose per i loro prodotti; tuttavia, al momento hanno difficoltà nel tradurle efficacemente in azioni concrete. In particolare, Pegasystems è uno strumento che presenta complessità di utilizzo e costi di licenza elevati, difficilmente sostenibili dalle piccole imprese. Creation Studio, similmente a Zoho, è intenzionata a integrarsi con i suoi strumenti Saas, ovvero Service Creatio, Sales Creatio e Marketing Creatio. La soluzione proposta da Globant, GeneXus, è orientata verso sviluppatori professionisti, ma pone particolare attenzione all'esperienza utente, a tal punto da poter essere considerata una piattaforma di sviluppo multi-esperienza (MXDP).

Infine, i Niche Players rappresentano quei venditori che sono specializzati in un particolare segmento di mercato, una regione geografica o hanno costruito una forte reputazione in altri settori. Per questo motivo, possono avere un impatto significativo in contesti specifici, ma sono troppo specializzate per essere adottate su larga scala. Questi sono Huawei, Newgen, Unqork e l'emergente prodotto proposto dall'azienda giapponese Kintone che, grazie alle sue caratteristiche di sviluppo con partecipazione dei cittadini, possibilità di creazione di flussi di lavoro, diverse estensioni e plugin, ha ricevuto il consenso di 30.000 clienti, principalmente tra piccole e medie imprese, solo nella prima metà del 2023. [1]

1.3 Il futuro del Low-Code

Il futuro dello sviluppo software mediante l'utilizzo delle piattaforme low-code è influenzato da diversi aspetti e sviluppi chiave:

- ***Aumento dell'adozione di piattaforme low-code***: dati i numerosi benefici offerti da tali piattaforme, Gartner stima che entro il 2025 più del 50% delle medio-grandi imprese implementerà applicazioni sviluppate tramite low-code. Di conseguenza, gli attori del mercato low-code saranno spinti a migliorare le proprie piattaforme, al fine di rendere disponibili soluzioni sempre più complete e in grado di adattarsi a diversi casi d'uso;
- ***“Democratizzazione” dello sviluppo software***: le piattaforme low-code stanno fornendo la possibilità, anche a coloro che non sono esperti di programmazione, di creare applicazioni personalizzate. Questo è il motivo per cui si parla di “democratizzazione” dello sviluppo software e si prevede che tale tendenza proseguirà, in direzione di uno sviluppo sempre più accessibile;
- ***Ambienti di sviluppo collaborativi***: offrendo la possibilità di utilizzo anche a coloro che non possiedono un alto livello di competenze, le piattaforme low-code consentiranno una migliore coordinazione nel processo di sviluppo tra i gruppi di sviluppatori e gli altri dipartimenti aziendali, con la conseguente realizzazione di applicazioni maggiormente efficaci ed efficienti;
- ***AI e apprendimento automatico nello sviluppo low-code***: l'integrazione tra intelligenza artificiale e piattaforme low-code contribuirà sempre di più all'ottimizzazione del processo di sviluppo. L'intelligenza artificiale, infatti, permetterà di automatizzare le attività ripetitive, con un conseguente aumento dell'efficienza nel processo di scrittura del codice;
- ***Migliore integrazione con altre tecnologie***: è probabile che il futuro dello sviluppo low-code veda un incremento dell'integrazione con diversi strumenti, tecnologie e piattaforme. Inoltre, un miglioramento dell'interoperabilità renderà più semplice il collegamento delle applicazioni low-code con i sistemi e le infrastrutture preesistenti, riducendo così la necessità di complesse operazioni di integrazione;

- **Attenzione alla sicurezza e alla conformità:** con l'aumento della domanda di piattaforme low-code, il tema relativo alla sicurezza diverrà sempre più importante. Ciò si traduce in una necessità crescente di soddisfare elevati standard e requisiti imposti dalle normative del settore. In futuro, tali piattaforme potrebbero concentrarsi, pertanto, sul miglioramento delle caratteristiche di sicurezza e conformità, al fine di instaurare un rapporto di fiducia con gli utenti.

In conclusione, è possibile osservare come le piattaforme low-code stiano complessivamente rivoluzionando il settore dello sviluppo software, rendendo il processo sempre più accessibile, collaborativo ed efficiente, e promuovendo un'innovazione continua. [10]

Capitolo 2

Oracle APEX

2.1 Introduzione ad Oracle Apex

Oracle Application Express, conosciuto come Oracle APEX, è una piattaforma di applicazioni low-code che permette di creare e distribuire in modo rapido applicazioni Web e Mobile scalabili e sicure in cloud e on-premise, senza la necessità di possedere competenze avanzate di sviluppo web. Tale strumento, integrato all'interno del Database Oracle, è in grado di rispondere in maniera efficiente a diverse esigenze aziendali: da applicazioni ad uso interno, utilizzate da pochi utenti, ad applicazioni pubbliche rivolte a clienti e partner. Oracle APEX, infatti, tramite una semplice interfaccia basata su browser, guida gli utenti nella realizzazione di soluzioni adatte a risolvere rapidamente problematiche aziendali, producendo dunque valore all'organizzazione e riducendo i costi associati allo sviluppo software. [11] [12] Inoltre, dato che non richiede un'esperienza professionale nell'uso di linguaggi di programmazione per il back-end, come C++, Java o Python, presenta una curva di apprendimento notevolmente rapida e limita la possibilità di errori che si possono generare quando è necessario scrivere un numero elevato di linee di codice. [13] La diffusione di Oracle APEX è ampiamente testimoniata anche dai numeri: sono state implementate oltre 19.000.000 di applicazioni da più di 700.000 sviluppatori. Ogni giorno, infatti, vengono sviluppate tra le 3.000 e le 5.000 applicazioni per soddisfare le esigenze di clienti distribuiti in tutto il mondo, evidenziando la vasta adozione e diffusione di Oracle APEX a livello globale. [14]

2.2 L'architettura di Oracle APEX

Oracle APEX è caratterizzato da un'architettura three-tier.

L'architettura three-tier è un modello utilizzato per la progettazione di applicazioni software, che organizza le applicazioni in tre livelli logici e fisici distinti, ciascuno dei quali svolge un ruolo specifico:

- **Il tier presentazione**, o *interfaccia utente*, rappresenta il punto di comunicazione dell'utente con l'applicazione. È, infatti, responsabile della presentazione dei dati in modo che siano accessibili e comprensibili all'utente finale e dell'interazione con essi. Le tecnologie più utilizzate per lo sviluppo di tale tier sono HTML, CSS e JavaScript.
- **Il tier applicazione**, denominato anche *tier logico*, è il livello in cui avviene l'elaborazione delle informazioni raccolte nel tier presentazione e ciò viene effettuato applicando la logica di business e interagendo con il tier dati. Inoltre, può svolgere operazioni quali l'aggiunta, l'eliminazione o la modifica dei dati nel tier dati, con il quale comunica tramite chiamate API. I linguaggi di programmazione Python, Java, Perl, PHP e Ruby sono i più comuni per lo sviluppo di tale tier.
- **Il tier dati**, o *tier database*, è il luogo di memorizzazione e gestione dei dati. Può essere implementato tramite sistemi di gestione di database relazionali, come PostgreSQL, MySQL, Oracle, Microsoft SQL Server, o NoSQL, come CouchDB e MongoDB.

L'architettura three-tier risulta essere l'architettura più diffusa e vantaggiosa per numerose implementazioni; ciò è dovuto al fatto che offre la possibilità di sviluppare, aggiornare e scalare ciascun livello in maniera indipendente dagli altri. Dato, infatti, che ogni tier viene eseguito sulla propria infrastruttura, si annoverano i seguenti vantaggi: uno sviluppo più rapido, una maggiore scalabilità e affidabilità dell'applicazione. Inoltre, un tier applicazione ben progettato può svolgere il ruolo di "firewall interno", migliorando la sicurezza e prevenendo attacchi, come ad esempio quello di SQL Injection. [15]

3 Tier Architecture

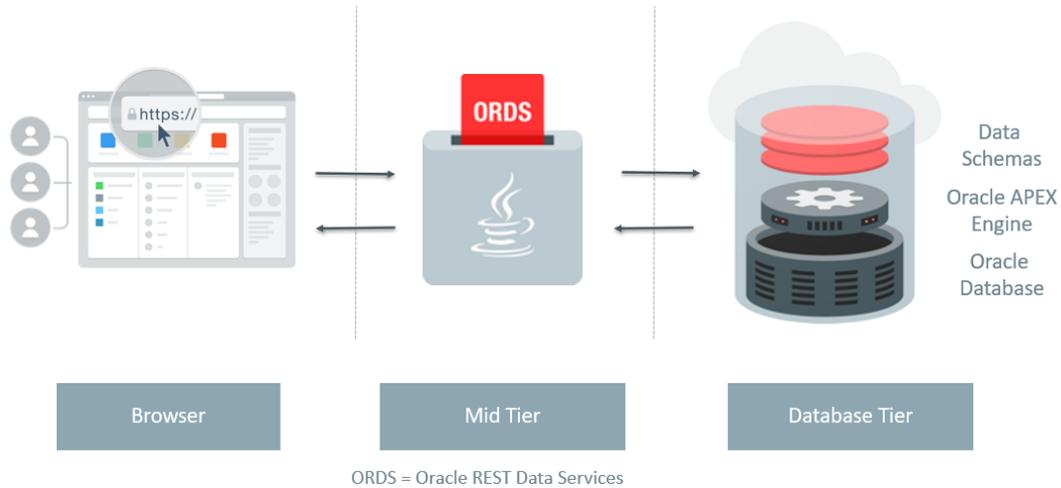


Figura 2.1. Architettura three-tier Oracle APEX. [2]

Nel caso specifico di APEX, l'architettura comprende un browser Web, un server Web, noto come Oracle REST Data Services (ORDS) e un database Oracle, che ospita Oracle APEX stesso. Il browser invia una richiesta web a Oracle REST Data Services, il quale la inoltra al database Oracle. All'interno di quest'ultimo, viene gestita la richiesta eseguendo l'elaborazione, la manipolazione dei dati e la logica di business. Al termine di questa, ORDS fa da tramite per restituire il risultato al browser. Tale approccio garantisce un accesso istantaneo ai dati, massime prestazioni e scalabilità. [16]

APEX non richiede una connessione al database dedicata per ogni utente; quando viene, infatti, effettuata una richiesta, si crea una nuova sessione del database, favorendo un minimo consumo di risorse della CPU. Chi si occupa di gestire nelle tabelle del database lo stato della sessione dell'applicazione è l'APEX engine.

L'APEX engine esegue il rendering ed elabora le pagine, gestendo anche altre importanti attività, quali appunto la gestione dello stato della sessione, oltre che servizi di autenticazione e autorizzazione, controllo del flusso della pagina e validazione dei dati. Quando si sviluppa o si espande un'applicazione utilizzando APEX, il sistema crea o aggiorna i metadati salvati nelle tabelle del database. Durante l'esecuzione dell'applicazione, l'APEX engine legge i metadati e li utilizza per visualizzare l'applicazione stessa. [17]

2.3 Oracle Application Express Environment

Oracle Application Express trasforma un singolo database Oracle in un ambiente condiviso, utilizzabile da diversi gruppi di lavoro. Gli sviluppatori vi possono accedere senza la necessità di installare software aggiuntivi mediante un semplice browser web.

Le applicazioni vengono sviluppate in ambienti denominati workspaces. Ogni workspace funge da database virtuale privato, all'interno del quale possono lavorare più utenti, mantenendo riservati i propri oggetti, dati e applicazioni.

Un tipico ambiente di sviluppo è caratterizzato da un singolo workspace condiviso da tutti i membri del team; tuttavia, è possibile anche regolare l'accesso agli elementi del workspace solo a utenti autorizzati, creando spazi di lavoro dedicati a specifici sviluppatori o progetti.

Al momento della loro creazione, i workspaces vengono associati a nuovi o esistenti *schemas*, ovvero contenitori logici al cui interno sono presenti oggetti del database come tabelle, viste o procedure memorizzate. Gli *schemas* non sono univoci per un dato workspace e possono, perciò, essere associati a uno o più spazi di lavoro. [3]

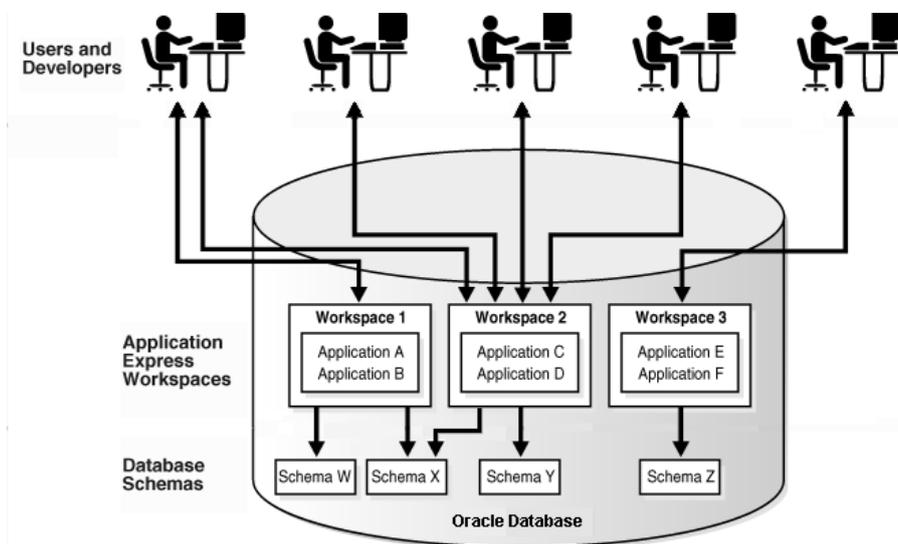


Figura 2.2. Oracle APEX Environment. [3]

La Home Page di un workspace consente di accedere ai principali componenti di sviluppo di Oracle APEX, ovvero App Builder, SQL Workshop, Team Development e Gallery. [17]

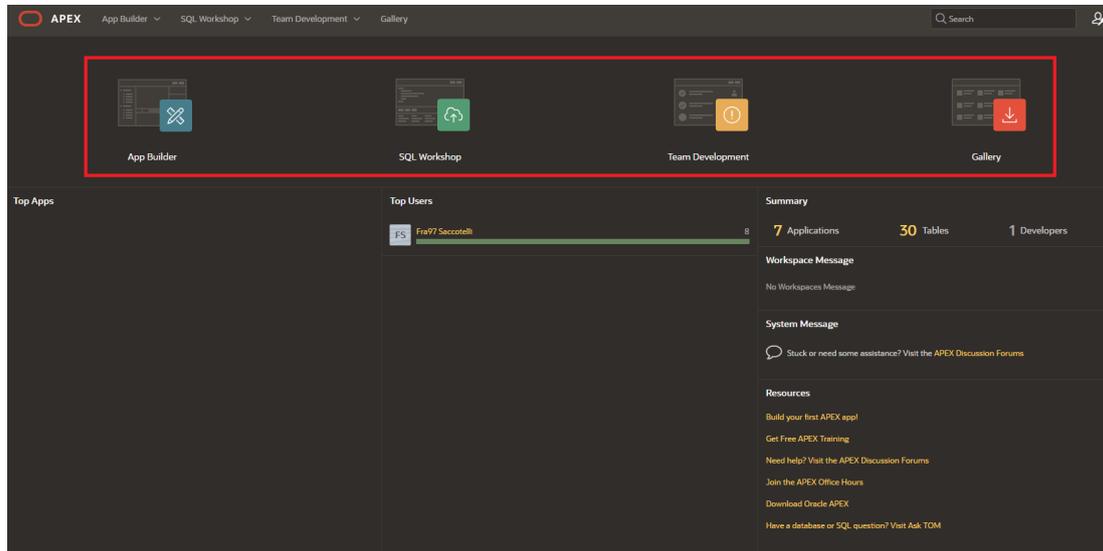


Figura 2.3. Oracle APEX Workspace.

2.3.1 App Builder

L'Application Builder è un elemento essenziale di Oracle APEX, che viene utilizzato per una serie di scopi. Questi includono la creazione di nuove applicazioni, l'importazione di applicazioni già esistenti, la visualizzazione di Dashboard, l'accesso alle Utilities dell'area di lavoro o la scelta di un'applicazione specifica, al fine di visualizzarla e modificarla.

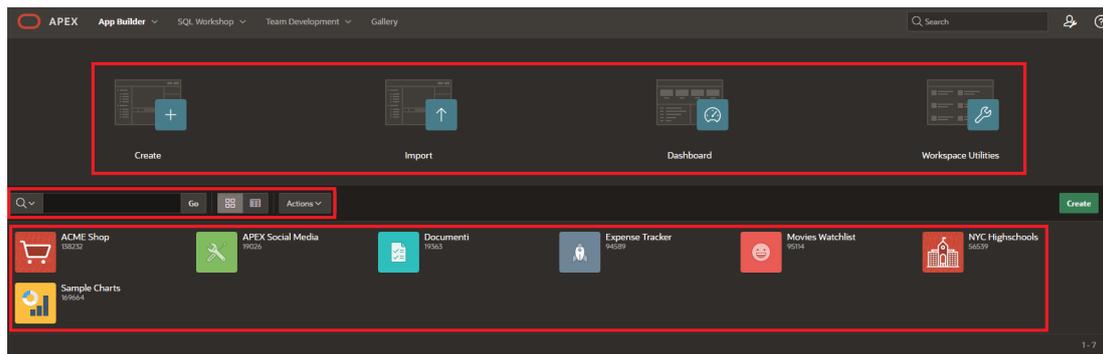


Figura 2.4. App Builder.

In particolare, la pagina iniziale dell'App Builder è suddivisa in due aree: la parte superiore è caratterizzata da quattro icone (Create, Import, Dashboard, Workspace Utilities), mentre la parte inferiore contiene tutte le applicazioni attualmente installate.

L'icona Create consente di avviare la creazione guidata di un'applicazione completa, mentre l'Import propone una procedura guidata per importare applicazioni che precedentemente sono state esportate o sviluppate in un ambiente esterno. Le icone Dashboard e Workspace Utilities, invece, permettono rispettivamente di collegarsi ad una dashboard che mostra le metriche delle applicazioni che si trovano nel workspace in uso e di accedere a numerosi servizi, quali servizi REST abilitati per SQL, server remoti, gruppi di applicazioni, viste Oracle APEX, gestione dei backup e catalogo delle fonti REST.

Infine, è possibile personalizzare la visualizzazione delle applicazioni tramite la barra di ricerca presente a metà tra le due sezioni. Tale possibilità semplifica la definizione di diverse tipologie di criteri di controllo sull'elenco delle applicazioni, garantendo agli utenti un maggior grado di precisione e caratterizzazione nell'esperienza

d'uso. [17]

2.3.2 SQL Workshop

L'SQL Workshop è un altro componente fondamentale di Oracle APEX, progettato per visualizzare e gestire in modo efficace gli oggetti del database. Tramite tale strumento, gli sviluppatori possono interagire con i dati, utilizzando diverse funzionalità. In particolare, possono creare, modificare, interrogare gli oggetti del database, tra cui tabelle, procedure, funzioni, viste e triggers. Inoltre, l'SQL Workshop mette a disposizione capacità di debugging e sicurezza avanzate tali da rendere l'esperienza di sviluppo produttiva e sicura.

L'SQL Workshop è composto da cinque elementi chiave: Object Browser, SQL Commands, SQL Scripts, Utilities e RESTful Services.

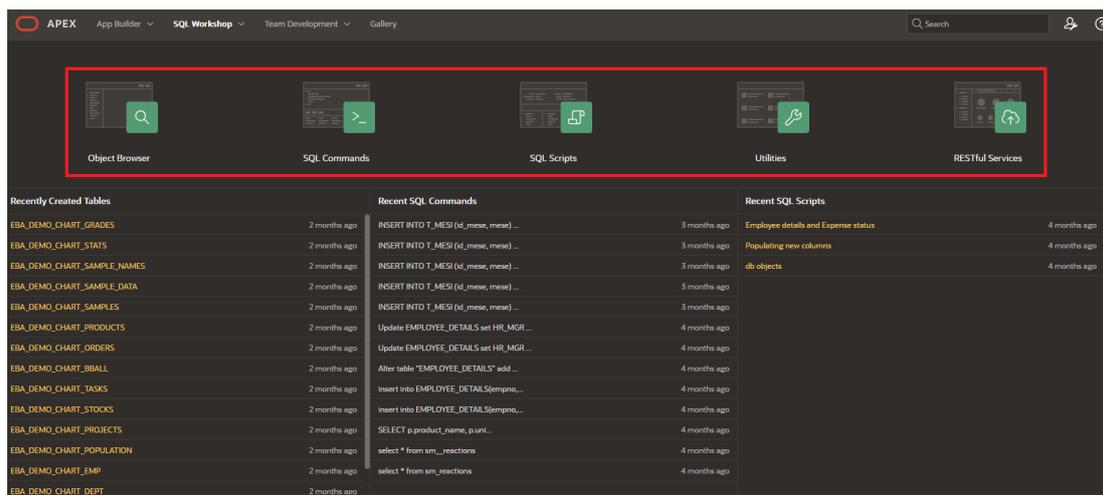


Figura 2.5. SQL Workshop.

L'Object Browser è il luogo in cui è possibile effettuare diverse operazioni sugli oggetti del database Oracle, tra cui la creazione, l'aggiornamento e l'eliminazione degli stessi. Tale componente è suddiviso in due riquadri: una struttura ad albero a sinistra contenente un elenco dei tipi di oggetto, al cui interno sono presenti quelli disponibili per lo schema corrente, e a destra la visualizzazione dei dettagli dell'oggetto selezionato, con schede specifiche per l'oggetto in cui poter svolgere le azioni desiderate.

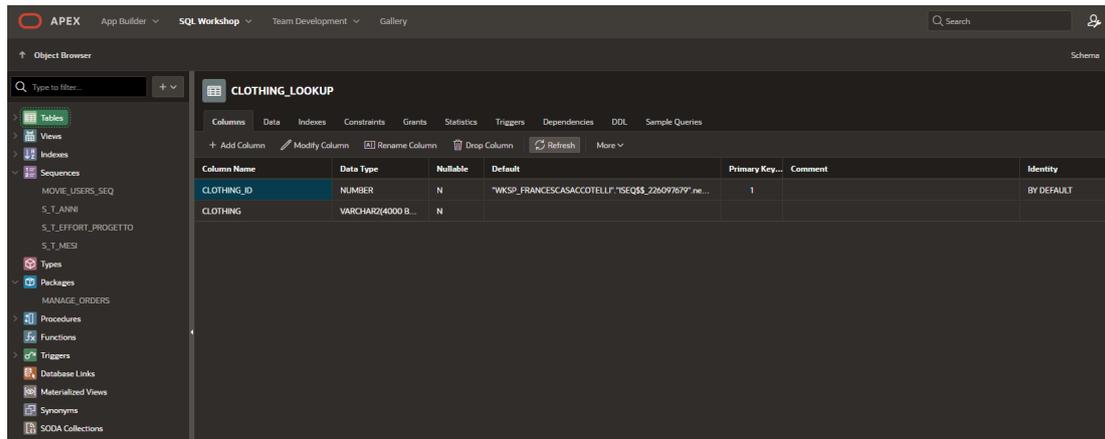


Figura 2.6. Object Browser.

L'SQL Commands è invece l'elemento di APEX che consente di creare, modificare, visualizzare ed eliminare oggetti del database tramite l'esecuzione di query SQL o blocchi PL/SQL. Nello specifico, nell'area superiore è presente l'Editor dei comandi SQL o PL/SQL, mentre la zona inferiore è dedicata alla visualizzazione dell'output, della cronologia dei comandi e degli elenchi relativi alle istruzioni salvate.

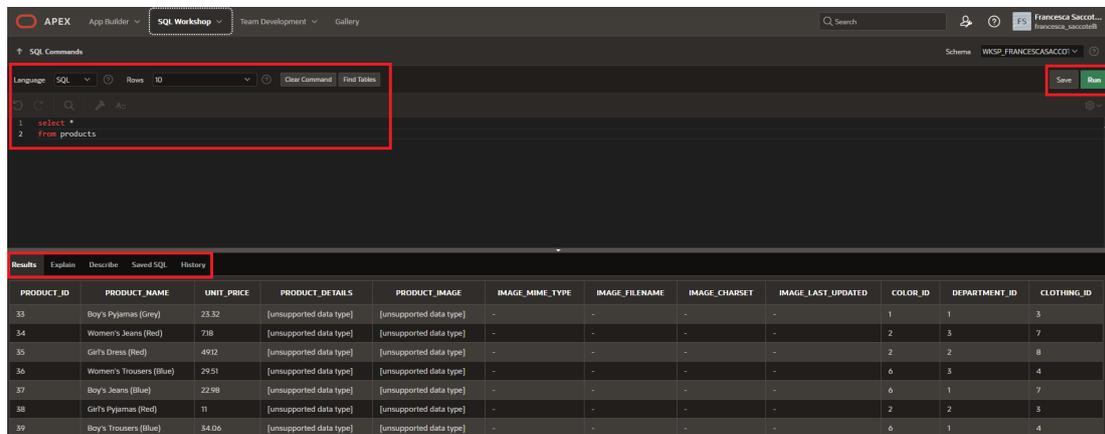


Figura 2.7. SQL Commands.

L'SQL Script è la sezione che permette di creare, aggiornare, visualizzare, eseguire ed eliminare script SQL. Questi ultimi non sono altro che insiemi di una o più istruzioni SQL o blocchi PL/SQL, usati per effettuare diverse operazioni sugli oggetti del database e memorizzati all'interno del repository SQL Script. È, inoltre,

possibile esportare ed importare script da un workspace ad un altro.

L'SQL Workshop Utilities offre allo sviluppatore diverse funzionalità, tra cui caricare dati nei formati CSV, XLSX, XML e JSON e scaricarli da un database Oracle, generare DDL (Data Definition Language), monitorare il database e visualizzarne i dettagli e generare in maniera semplificata l'SQL necessario per la creazione degli oggetti del database. Quest'ultimo strumento prende il nome di Quick SQL ed è un elemento estremamente innovativo di Oracle APEX, in quanto consente, tramite una sintassi abbreviata, di creare in maniera rapida uno script per tabelle, viste semplici e altri oggetti del database.

```

1 # profile: sm
2 # apex: true
3 posts /auditrows
4 post_comment vc4000
5 file_blob blob
6 file_mime vc255
7 file_name vc255
8 lat num
9 lon num
10 reactions /auditrows
11 post_id /fk posts
12 reaction vc10
13 lat num
14 lon num

```

```

1 -- create tables
2
3 create table sm_posts (
4   id          number generated by default on null as identity
5   constraint sm_posts_id_pk primary key,
6   post_comment varchar2(4000 char),
7   file_blob   blob,
8   file_mime   varchar2(255 char),
9   file_name   varchar2(255 char),
10  lat         number,
11  lon         number,
12  created     date not null,
13  created_by  varchar2(255 char) not null,
14  updated_by  date not null,
15  updated_by  varchar2(255 char) not null
16 );
17
18
19 create table sm_reactions (
20  id          number generated by default on null as identity
21  constraint sm_reactions_id_pk primary key,
22  post_id     number
23  constraint sm_reactions_post_id_fk
24  references sm_posts,
25  reaction   varchar2(10 char),
26  lat       number,
27  lon       number,
28  created   date not null,
29  created_by varchar2(255 char) not null,
30  updated_by date not null,
31  updated_by varchar2(255 char) not null
32 );
33
34 -- table index
35 create index sm_reactions_i1 on sm_reactions (post_id);

```

Figura 2.8. Esempio Quick SQL.

In ultimo luogo, attraverso i RESTful Services si possono definire in modo dichiarativo i servizi RESTful usati per accedere al database. Tali servizi permettono di creare una serie di API standards-based sugli oggetti del database presenti nel proprio workspace, le quali consentono ai sistemi esterni di interrogare, inserire, aggiornare o eliminare dati in modo sicuro, senza dover accedere direttamente al database sottostante.

Le definizioni dei servizi REST (Representational State Transfer) creati in APEX vengono conservate nel repository Oracle REST Data Services (ORDS) e sono identificate come ORDS-based REST Services. [18]

2.3.3 Team Development

La sezione Team Development ha come obiettivo quello di rendere più semplice la collaborazione tra i membri del team di sviluppatori durante il processo di creazione e gestione delle applicazioni. Tramite tale strumento si può tenere traccia e gestire eventuali criticità, impostare tappe fondamentali relative ai progetti, nonché assegnare la risoluzione di specifici problemi alle varie risorse del gruppo. [17] [19]

2.3.4 Gallery

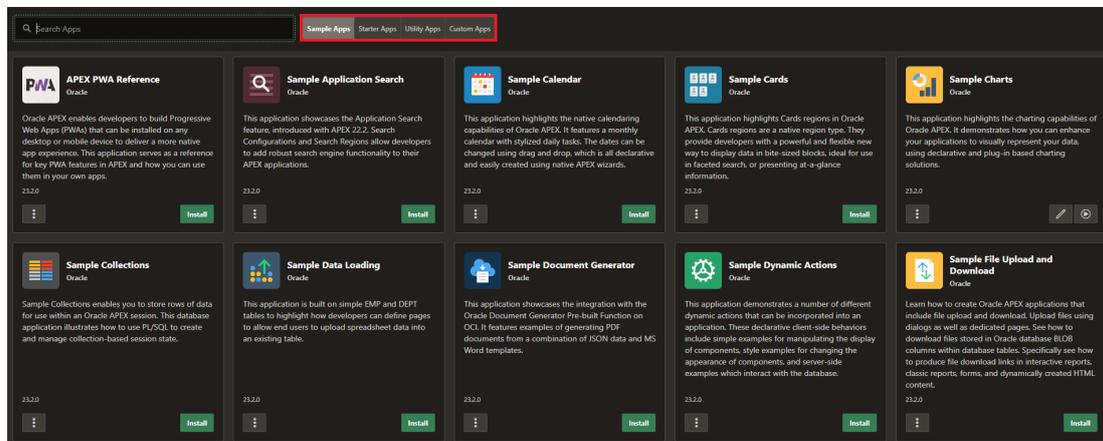


Figura 2.9. Gallery.

Infine, l'App Gallery mette a disposizione degli utenti una raccolta di applicazioni preconfigurate che possono fungere come punto di partenza per l'implementazione delle proprie applicazioni. Nello specifico, l'App Gallery permette di accedere a:

- **Sample Apps**, un gruppo di applicazioni che mostrano agli sviluppatori funzionalità specifiche;
- **Starter Apps**, applicazioni che propongono soluzioni autonome per requisiti semplici e che possono essere estese con funzionalità personalizzate;
- **Custom Apps**, gruppo di applicazioni personalizzate, installabili nel proprio workspace se messe a disposizione dall'amministratore dell'istanza.

Nella versione 23 di Oracle APEX, l'App Gallery presenta anche la sezione Utility Apps, contenente un'applicazione che permette di vedere le fonti dati REST disponibili nelle applicazioni Oracle Cloud del proprio workspace. Questa applicazione offre, dunque, una guida per recuperare informazioni di riferimento rilevanti, oltre alla possibilità di scoprire nuovi endpoint REST disponibili per l'utilizzo. [17]

2.4 Creare applicazioni in Oracle APEX

Nel precedente paragrafo è stato trattato il concetto di workspace, analizzando nel dettaglio i suoi elementi costituenti. In particolare, è stato definito l'Application Builder, componente tramite il quale è possibile procedere con la creazione delle proprie applicazioni.

Un'applicazione APEX è un'interfaccia HTML, che si poggia su oggetti del database, come tabelle, funzioni o procedure.

Per creare un'applicazione, è necessario cliccare sul bottone *Create* presente nella home page di App Builder, il quale rimanda ad una pagina in cui è possibile selezionare uno tra i seguenti metodi di creazione:

- ***Create Application***, per creare un'applicazione contenente solo una Global Page, una Home Page e una Login Page, a cui potranno essere aggiunte successivamente pagine eseguendo la Create Page Wizard;
- ***Use Create App Wizard***, per creare un'applicazione utilizzando la Create Application Wizard completa, che consente di implementare una soluzione basata su tabelle selezionate dall'utente o tramite SQL. A questa è possibile aggiungere in un secondo momento pagine con vari componenti, funzionalità e opzioni di configurazione per abilitare o disabilitare funzionalità specifiche. Si tratta di un metodo completo e flessibile, in quanto permette una configurazione e personalizzazione dettagliata dell'applicazione;
- ***Create App From a File***, per creare applicazioni tramite il caricamento di un file di tipo CSV, XLSX, XML o JSON oppure copiando e incollando i dati;

- *Create App From Quick SQL*, per creare un'applicazione a partire da uno script generato con lo strumento Quick SQL;
- *Browse App in Gallery*, per installare una delle numerose applicazioni presenti nella sezione Gallery;
- *Copy Existing App*, per creare una copia di un'applicazione già esistente.

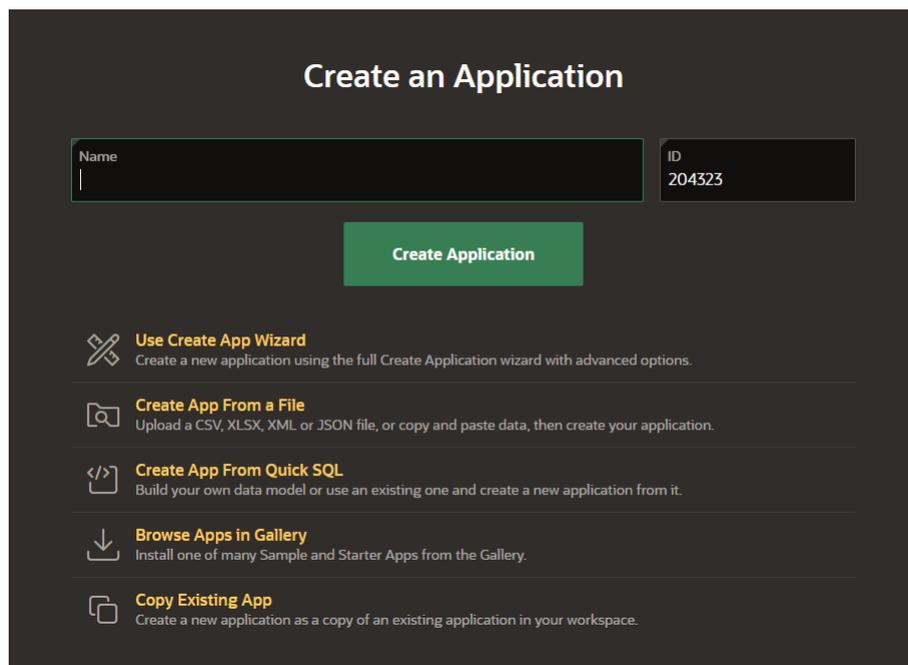


Figura 2.10. Creare un'applicazione in Oracle APEX.

Le applicazioni sono composte da una o più pagine collegate tra loro attraverso un menu di navigazione, schede, bottoni o links. Le singole pagine sono strutturate utilizzando *regions*, contenitori che possono racchiudere diversi elementi, tra cui report, grafici, mappe, calendari e form. I form, a loro volta, sono caratterizzati da campi denominati *items*, che possono variare da semplici campi di testo a select lists o checkboxes. Inoltre, gli sviluppatori possono estendere le funzionalità predefinite tramite l'uso di plug-in personalizzati per soddisfare specifiche esigenze dell'applicazione.

Le pagine, oltre alla rappresentazione dei dati, possono includere la logica dell'applicazione come processi, calcoli e validazioni. Esse condividono un'autenticazione

e uno stato di sessione comuni, quest'ultimo gestito da APEX in maniera trasparente. La logica dell'applicazione è separata dalla presentazione dell'interfaccia utente, così che gli sviluppatori possano gestire l'aspetto grafico senza dover modificare il codice che gestisce le funzionalità dell'applicazione stessa.

La sezione in cui è possibile visualizzare e modificare una pagina di un'applicazione è il Page Designer. [17]

2.4.1 Page Designer

Il Page Designer è un ambiente di sviluppo integrato (IDE – Integrated Development Environment), nel quale sono definiti tutti gli elementi fondamentali che caratterizzano una pagina di un'applicazione. Esso è contraddistinto dalla presenza di una toolbar e tre riquadri principali, la cui dimensione può essere regolata dall'utente in base a necessità.

La toolbar del Page Designer, collocata nella parte superiore di ogni pagina, comprende una serie di funzionalità, tra cui pulsanti e opzioni di menu. Oltre alla visualizzazione e alla ricerca di una pagina, al salvataggio delle modifiche e all'esecuzione dell'applicazione, assume rilevanza il suo collegamento alla pagina *Shared Components*. Quest'ultima, al suo interno, offre una serie di strumenti e procedure guidate utilizzabili sia a livello generale nell'applicazione, sia in specifiche pagine.

Il pannello di sinistra è costituito da quattro tab distinti: Rendering, Dynamic Actions, Processing e Page Shared Components. Ciascuna di queste schede contiene un elenco dei vari tipi di componenti disponibili, nonché dei componenti specifici già creati nella pagina corrente. Nel Rendering vengono create le regioni, gli items e i bottoni, al cui interno è definita la logica dell'applicazione.

Nelle Dynamic Actions, invece, è possibile definire comportamenti complessi lato client per la pagina corrente, senza l'impiego di codice Javascript, mentre nel Processing è mostrata la logica applicativa definita nella pagina, dove i componenti sono organizzati in base al modo in cui APEX li elabora. Infine, nella Page Shared Components sono visualizzabili i componenti condivisi tra la pagina corrente e altre pagine dell'applicazione.

Il pannello centrale contiene nella parte superiore le schede di Layout, Page Search

e Help, mentre nella parte inferiore la sezione Gallery. La scheda di Layout mostra gli elementi presenti e la loro disposizione all'interno della pagina. La scheda Page Search consente, invece, di cercare tutti i metadati della pagina (comprese regioni, azioni dinamiche, colonne, ecc.), mentre Help fornisce una spiegazione per ogni attributo del riquadro di destra, anche noto come Property Editor. La Gallery presenta un elenco delle regions, degli items e dei bottoni che, tramite azioni drag-and-drop, possono essere aggiunti in una specifica area del riquadro centrale superiore.

In ultimo luogo, nella parte laterale destra è presente il Property Editor, in cui sono presenti tutti gli attributi aggiornabili del componente attualmente selezionato nel pannello centrale o di sinistra. Questi dettagli racchiudono caratteristiche identificative dell'elemento (nome e tipo), configurazioni relative alla sicurezza e in particolare l'origine dei dati associati all'elemento. Tale origine può essere determinata da una query SQL o da una tabella del database. È inoltre possibile selezionare più componenti contemporaneamente, di cui saranno visualizzabili e modificabili gli attributi comuni. [17]

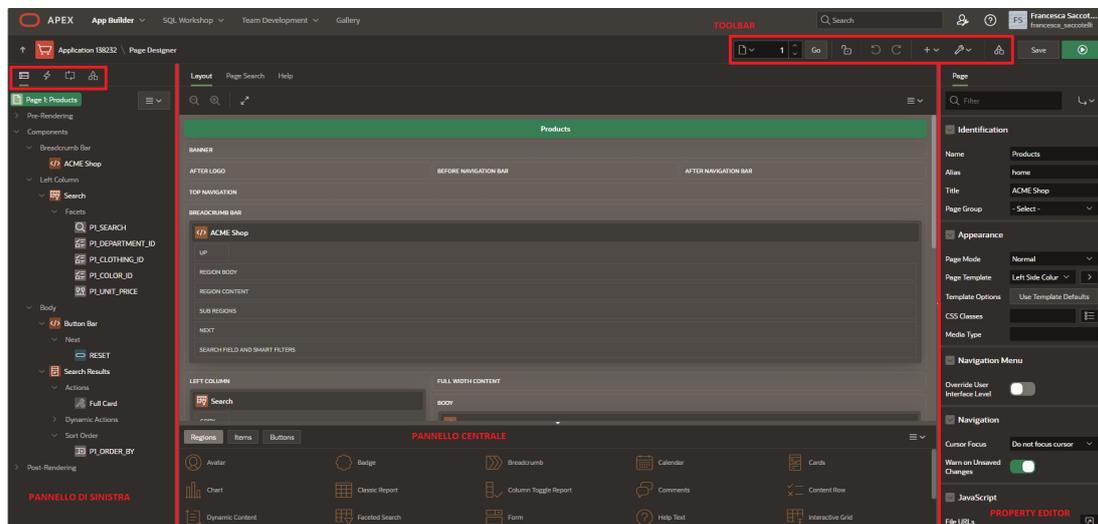


Figura 2.11. Page Designer.

2.5 Vantaggi di Oracle APEX

Oracle APEX si distingue per i suoi numerosi vantaggi, che contribuiscono a rendere significativamente migliori il processo di sviluppo e gestione delle applicazioni web aziendali.

Uno dei suoi principali benefici è, infatti, l'eliminazione della complessità associata allo sviluppo e alla distribuzione delle applicazioni a diversi livelli, tra cui l'eliminazione della corrispondenza tra gli oggetti relazionali e gli oggetti dell'applicazione e l'eliminazione della proliferazione delle connessioni al database. La sua natura low-code genera un ambiente di sviluppo maggiormente rapido e intuitivo rispetto alle soluzioni tradizionali, offrendo la possibilità agli sviluppatori di concentrarsi sulla sola creazione di valore e non su aspetti tecnici complessi. Oracle APEX, infatti, richiede principalmente familiarità con i linguaggi SQL e PL/SQL e, in minima parte, conoscenze di HTML, CSS e JavaScript. Ciò comporta sia un risparmio di tempo che di risorse, nonché costi ridotti sul lungo termine durante le fasi di sviluppo e di manutenzione delle applicazioni.

Un'altra caratteristica importante consiste nella facilità d'uso e apprendimento. Gli sviluppatori, tramite l'interfaccia intuitiva offerta dall'Application Builder, possono essere assistiti durante l'intero processo di creazione delle applicazioni e aggiungere facilmente funzionalità avanzate alle loro soluzioni, grazie alla presenza di potenti componenti predefiniti.

Inoltre, essendo una piattaforma estremamente potente e collaudata, è in grado di soddisfare una vasta gamma di esigenze applicative in vari settori industriali. Oracle APEX, infatti, tramite un'architettura elegante e scalabile, garantisce prestazioni consistenti nel tempo e la possibilità di creare applicazioni semplici o di importanza critica utilizzate da migliaia di utenti ogni giorno.

Tale piattaforma assicura anche un elevato livello di sicurezza integrato nelle applicazioni web e la conformità agli standard di sicurezza del settore. Questo aspetto, in un panorama in cui le minacce informatiche sono sempre più sofisticate, consente ad APEX di configurarsi come soluzione sicura e affidabile, in grado di proteggere le applicazioni create.

Infine, la portabilità rappresenta un ulteriore vantaggio significativo di Oracle APEX. Le applicazioni create con questo tool possono essere eseguite nel Cloud

Oracle, in locale o in qualsiasi ambiente in cui sia presente un database Oracle. Ciò garantisce una maggiore flessibilità agli sviluppatori e alle imprese, consentendo loro di distribuire e gestire le proprie applicazioni in maniera efficiente. [20]

2.5.1 Integrazione di Oracle APEX nel database Oracle

Come precedentemente introdotto, Oracle APEX è una piattaforma di sviluppo web che offre una serie di vantaggi significativi per coloro che cercano uno strumento semplice ed efficiente per creare applicazioni aziendali.

Una delle caratteristiche più rilevanti di tale piattaforma e da cui derivano i successivi punti trattati risiede nella sua integrazione nativa nel database Oracle; ciò significa che, disponendo di un database Oracle, si dispone già di Oracle APEX senza oneri aggiuntivi legati a licenze e senza vincoli sul numero di sviluppatori, applicazioni create e utenti finali. Inoltre, data questa peculiarità, gli sviluppatori hanno accesso a tutte le funzionalità e capacità del database Oracle, consentendogli di realizzare soluzioni complete e pronte per le aziende fin dal primo giorno.

Se si utilizza il supporto per il database Oracle o ci si avvale di un Oracle Database Cloud Service, è possibile ricevere anche un servizio di assistenza da parte di un team specializzato di analisti. Un altro aspetto rilevante che contribuisce alla sua versatilità consiste nella possibilità di essere eseguito su qualsiasi tipo di supporto in cui operi Oracle Database, che sia in locale, su Oracle Cloud o su altre piattaforme.

Oracle APEX offre anche la possibilità di analizzare e trasformare in informazione ogni tipo di dato. Nel database Oracle, il linguaggio SQL standard consente di definire agevolmente l'accesso, la gestione e l'analisi dei dati, supportando anche dati non relazionali, tra cui JSON, XML, documenti e dati spaziali. Per quanto concerne l'analisi dei dati, tale piattaforma mette a disposizione per le proprie applicazioni diversi strumenti che spaziano dalle query di base, come una semplice istruzione SELECT ad analisi più complesse, effettuate ad esempio mediante l'uso di funzioni analitiche e funzioni statistiche.

Il database Oracle fornisce molteplici funzionalità a cui è possibile associare tre parole chiave: scalabilità, performance e sicurezza. Per quanto riguarda le prime due, con Oracle Real Application Clusters è possibile distribuire in modo trasparente i

propri database su un pool di server in cluster, potendo così sfruttare al massimo la memoria combinata e la capacità di elaborazione. In aggiunta, con Oracle Database In-Memory le query possono essere eseguite a velocità straordinarie, consentendo di effettuare analisi in tempo reale.

Nel contesto della sicurezza, il database Oracle mette a disposizione diversi strumenti per valutare i rischi, impedire che i dati vengano divulgati senza autorizzazione e rilevare attività anomale. Tra questi vi sono: la cifratura dei dati trasparenti (TDE), il Virtual Private Database (VPD), il Real Application Security (RAS) e l'audit del database.

In relazione a Oracle Database è possibile parlare anche di una nuova generazione di database autogestiti che monitorano, eseguono test diagnostici e che si ottimizzano in autonomia, riducendo i costi operativi di IT e garantendo prestazioni migliori.

L'architettura multi-tenant è un'ulteriore caratteristica innovativa di Oracle Database: un singolo container database multi-tenant può contenere molti pluggable database, così da poter gestire più database come se fossero uno solo. Ciò comporta, ad esempio, che task come backup e disaster recovery vengano eseguiti a livello di container database, semplificando notevolmente il processo di gestione dei dati. Grazie alla presenza di nuovi comandi SQL, la creazione, lo spostamento e la duplicazione dei pluggable database richiedono pochi secondi e la gestione delle risorse è ottimizzata grazie a funzionalità implementate nel Resource Manager di Oracle Database 12c.

Infine, tra le caratteristiche di notevole importanza si inseriscono anche le tecnologie flashback del database Oracle, sviluppate per rendere il recupero dei dati danneggiati o accidentalmente eliminati un processo rapido ed efficace. La funzione *Flashback Query*, integrata direttamente nelle griglie o nei report interattivi, permette di eseguire query sui dati in un determinato momento del passato, così da poter ricostruire eventi o modifiche dannose. Le funzioni *Flashback Drop* e *Flashback Database*, invece, consentono rispettivamente il recupero di tabelle eliminate involontariamente e il ripristino dell'intero database in uno specifico punto nel tempo.

In conclusione, l'integrazione tra Oracle APEX e il database Oracle offre alle imprese una piattaforma che è in grado di soddisfare elevati standard a livello di performance, sicurezza e flessibilità, caratteristiche essenziali per avere la possibilità di ottenere un vantaggio competitivo nel contesto digitale odierno. [21]

2.6 Casi d'uso di Oracle APEX

Oracle Application Express permette di sviluppare soluzioni in grado di trasformare e ottimizzare diversi processi aziendali. In questo paragrafo, saranno esaminati alcuni dei casi d'uso più comuni in cui tale piattaforma può essere impiegata con successo.

2.6.1 Nuove Applicazioni Cloud

Uno dei principali casi d'uso è rappresentato dalla creazione di applicazioni cloud, in grado di rispondere rapidamente alle opportunità che si possono generare in un ambiente aziendale in continua evoluzione. APEX consente di ideare prototipi funzionanti in pochi minuti, grazie a strumenti come Quick SQL e all'interfaccia completa e intuitiva dell'Application Builder. Inoltre, il suo approccio low-code, favorendo il processo di sviluppo, semplifica anche la fase di revisione con gli utenti finali, con aggiornamenti che possono essere apportati in tempo reale e, dunque, essere perfettamente in linea alle esigenze del business. [22]

2.6.2 Sostituzione dei fogli di calcolo

Un altro caso d'uso molto importante riguarda la sostituzione dei fogli di calcolo. La maggior parte delle organizzazioni utilizza fogli di calcolo per distribuire e creare report sui dati; tuttavia, la loro diffusione può portare ad esempio a gestire diverse copie nei dipartimenti aziendali, con conseguente possibilità di dati differenti e processi aziendali imperfetti. Oracle APEX si inserisce come soluzione a questo problema, consentendo di trasformare velocemente i fogli di calcolo in tabelle contenenti l'informazione completa dei dati nel database Oracle, i quali potranno

successivamente essere manipolati e visualizzati tramite form, report e altri componenti. Dunque, attraverso una singola fonte di informazioni centralizzata viene garantita integrità, sicurezza avanzata e coerenza dei dati, i quali potranno, quindi, essere consultati e modificati da qualsiasi risorsa aziendale di competenza. [22]

2.6.3 Condivisione di dati esterni

Oracle APEX può essere anche utilizzato per condividere dati con utenti esterni all'organizzazione, ovvero clienti e partner. È possibile, infatti, creare applicazioni pubbliche personalizzate, attraverso le quali i clienti e i partner possono interagire in maniera semplice, intuitiva e sicura. Con Oracle Database Cloud Service, si possono ospitare applicazioni su cloud in un ambiente costantemente monitorato da Oracle, senza avere preoccupazioni legate all'infrastruttura o alla sicurezza dei dati. Inoltre, l'integrazione RESTful di APEX consente agli sviluppatori di esporre e utilizzare facilmente i servizi web RESTful per interagire con altri sistemi, fornendo ai clienti numerose e diverse funzionalità. Tali caratteristiche contribuiscono, perciò, non solo a vantaggi interni alle organizzazioni, ma favoriscono anche l'ampliamento e la soddisfazione di coloro che intrattengono rapporti con le aziende dall'esterno. [22]

2.6.4 Modernizzazione di Oracle Forms

Oracle APEX ha posto le basi per consentire la modernizzazione di applicazioni dati ormai obsolete, create tramite Oracle Forms.

Condividendo diversi aspetti, tra cui l'utilizzo di framework dichiarativi, l'orientamento al database, il fatto di basarsi sui linguaggi SQL e PL/SQL e la possibilità di usare funzioni e oggetti del database Oracle, la transizione da Oracle Forms a Oracle APEX è particolarmente semplice per gli sviluppatori.

Per decenni Oracle Forms è rimasta popolare, tuttavia vi sono diverse motivazioni che hanno spinto a migrare verso le alternative disponibili:

- difficoltà di utilizzo a causa di applicazioni spesso non intuitive;
- scarsa adattabilità a dispositivi diversi;

- applicazioni non particolarmente adatte per ambienti esterni, ovvero per clienti e partner;
- difficoltà a reperire e assumere a costi accessibili sviluppatori Oracle Forms esperti.

Le applicazioni APEX, invece, sono moderne, reattive, eseguibili su qualsiasi browser Web e non richiedono una formazione specifica per essere utilizzate; inoltre, sono costituite da un ampio numero di modelli che permettono di personalizzare le applicazioni in base a richieste specifiche. Grazie a queste caratteristiche, con APEX è possibile avere un'esperienza utente superiore rispetto a quella offerta dalle vecchie applicazioni client/server. [22]

2.6.5 Report interattivi

Oracle APEX rende semplice la creazione di report interattivi, basati su dati eterogenei.

Avere un quadro generale o dipartimentale della propria realtà operativa è, infatti, di notevole importanza per le organizzazioni, che desiderano poter prendere delle decisioni consapevoli sulla base di informazioni strutturate. Inoltre, spesso è richiesta la protezione di dati sensibili, con conseguente necessità di stabilire privilegi di accesso ai dati. I report interattivi di APEX rispondono perfettamente a queste esigenze: unificano e trasformano dati provenienti da molteplici fonti in un'unica piattaforma, permettono agli utenti di manipolare i dati in maniera flessibile, applicando filtri, selezionando o ordinando colonne, eseguendo calcoli e il pivot dei dati. Gli utenti possono adattare la visualizzazione delle informazioni in base alle loro specifiche esigenze e i dati possono essere gestiti in modo tale che possano essere visualizzati solo da utenti autorizzati. In questo modo la creazione di dashboard specifiche per le varie community di utenti si semplifica notevolmente. [22]

2.6.6 Integrazione a sistemi ERP esistenti

Un'altra possibilità offerta da Oracle APEX è quella di estendere e integrare i sistemi ERP (Enterprise Resource Planning) delle organizzazioni. Questa esigenza

nasce dalla mancanza, in determinate soluzioni, di alcuni elementi importanti, come report specifici o funzionalità avanzate, specialmente quando si tratta di particolari requisiti del settore o dell'organizzazione. Inoltre, questi sistemi possono risultare inefficaci nei comuni processi aziendali, in quanto caratterizzati da numerosi passaggi e attività. Oracle APEX interviene in questo quadro, sviluppando estensioni che generano processi aziendali semplificati con un numero ridotto di passi e creando applicazioni su misura, che possono utilizzare dati del sistema ERP, ma memorizzare anche dati aggiuntivi in oggetti del database locali. Permettendo di integrare dati provenienti non solo dai sistemi ERP, APEX consente di creare report e dashboard migliori e un maggior numero di business flow complessi. [22]

2.7 Prospettive future di Oracle APEX

Oracle APEX rilascia ogni anno release, al fine di adeguarsi alla rapida evoluzione del panorama tecnologico e per rispondere alle mutevoli esigenze degli sviluppatori e degli utenti finali.

È possibile individuare diversi aspetti chiave che caratterizzeranno i suoi futuri aggiornamenti.

In primo luogo, è prevista l'introduzione di APEX Assistant, uno strumento di sviluppo assistito dall'Intelligenza Artificiale, che promette di facilitare notevolmente il processo di sviluppo delle applicazioni. Tramite comandi scritti con un linguaggio d'uso quotidiano, guiderà gli sviluppatori nella progettazione del modello dati, nella scrittura di query SQL e nella creazione di pagine e intere applicazioni, riducendo il tempo e lo sforzo necessari per l'implementazione di soluzioni di qualità.

Un altro passo importante verso l'Intelligenza Artificiale è rappresentato dall'inclusione di funzionalità di apprendimento automatico direttamente all'interno delle applicazioni APEX. Grazie ad una vasta gamma di algoritmi di machine learning sarà, infatti, possibile integrare facilmente funzionalità di previsione, classificazione e clustering, favorendo così la nascita di nuove opportunità di innovazione e personalizzazione.

Oracle concentrerà i suoi sforzi anche sull'espansione delle capacità visive e analitiche di APEX. Infatti, fornendo una rappresentazione visiva della logica dell'applicazione, consentirà agli sviluppatori di comprendere e visualizzare i dati in maniera più efficace, anche grazie all'integrazione con le funzioni analitiche del database. Ciò favorirà un'esperienza lato utente maggiormente ricca e intuitiva e la realizzazione di applicazioni con visualizzazioni più dinamiche dei dati e analisi avanzate. Oracle si prefigge come ulteriore obiettivo quello di migliorare la capacità di stampa ed esportazione in PDF, mediante l'aggiunta di opzioni avanzate per la stampa dei report e un'interfaccia per i servizi di stampa PDF integrati. Tali aspetti consentiranno una maggiore flessibilità e personalizzazione nella gestione dei documenti. Infine, verranno aggiunte nuove funzionalità alla console del servizio cloud di APEX Service, nonché nuovi tipi di connettori sorgente REST. Questi aggiornamenti mirano rispettivamente a semplificare in generale le attività di gestione delle applicazioni e a facilitare l'integrazione con le API REST di fornitori, incrementando in questo modo le interazioni con sistemi esterni.

In conclusione, tali prospettive future mostrano come tale piattaforma sia costantemente orientata all'innovazione continua, all'adattamento alle condizioni del mercato e all'offerta di strumenti sempre più potenti e flessibili. Con l'introduzione di soluzioni basate sull'Intelligenza Artificiale, la capacità di integrazione del machine learning e nuove funzionalità avanzate di visualizzazione e analisi dei dati, APEX si conferma una piattaforma versatile e all'avanguardia, pronta a rispondere alle sfide del futuro. [23]

Capitolo 3

MM Macropianificazione

Nell'attuale contesto gestionale e operativo delle imprese, l'allocazione efficiente delle risorse rappresenta un aspetto determinante per il successo dei progetti. Nel seguente capitolo verrà esplorato il ruolo svolto dall'applicazione MM Macropianificazione, realizzata tramite la piattaforma Oracle APEX, nel processo di pianificazione delle attività e gestione delle risorse aziendali.

L'applicazione offre una soluzione che consente di fornire una panoramica sulla pianificazione dei progetti nel corso dei mesi e sulle risorse disponibili. Attraverso l'utilizzo di un algoritmo, l'applicazione permette di allocare le risorse alle diverse attività, prendendo in considerazione diversi vincoli, quali ad esempio clienti di competenza, seniority e tecnologie di riferimento delle risorse.

Nel corso di questo capitolo verranno trattati i requisiti stabiliti in fase di progettazione, la base dati, le logiche e le funzionalità chiave dell'applicazione, esaminando nel dettaglio il suo processo di funzionamento.

3.1 Il Contesto

L'azienda Mediamente Consulting S.r.l è una società di consulenza che si occupa della gestione e della valorizzazione dei dati che guidano le medie e grandi imprese nelle decisioni strategiche di business. Le aree di specializzazione riguardano diversi ambiti, quali Technological Infrastructure, Data Integration and Management, CPM (Corporate Performance Management), Business Intelligence e Advanced Analytics. L'azienda, inoltre, offre numerosi servizi, tra cui servizi di Consulting, Assessment, Advising, Training e IT Management. [24]

I progetti e le attività svolte dall'azienda sono pertanto molteplici e richiedono una gestione ottimizzata delle risorse, la quale possa garantire un'allocazione efficace e mirata ai vari progetti in base alle necessità e alle competenze richieste.

3.2 Analisi dei requisiti

Nel secondo capitolo sono stati esaminati i principali casi d'uso di Oracle APEX. Tra questi, quello che risponde perfettamente alle esigenze dell'azienda Mediamente Consulting è rappresentato dalla necessità di sostituire diversi fogli di calcolo utilizzati per la pianificazione delle attività e la gestione delle risorse aziendali. L'azienda necessitava, infatti, di un'applicazione web che fosse in grado di tener traccia di tali informazioni in maniera strutturata, di facile utilizzo e che consentisse di effettuare in autonomia l'allocazione delle risorse ai progetti, precedentemente gestiti manualmente.

I requisiti specificati dall'azienda dovevano includere, dunque, le seguenti funzionalità:

- ***Inserimento dei progetti:*** inserimento dettagliato delle attività da svolgere nell'arco di uno o più mesi, includendo la possibilità di assegnare ad esse una stima dei giorni necessari per il loro completamento. A ciascun progetto dev'essere inoltre associato il calcolo dell'effort equivalente, ottenuto dal prodotto tra la stima dei giorni necessari al completamento dell'attività e la percentuale di affidabilità sulla stima stessa;

- **Creazione reportistica:** report in cui visualizzare in maniera riepilogativa i progetti da realizzare. Ogni report deve mostrare, per ogni attività, il numero di giorni assegnati durante i vari mesi, presentati in una singola riga. Nello specifico, i report devono essere i seguenti: uno in cui le attività sono aggregate per giorni, uno per FTE (Full-Time Equivalent) e uno in cui sono le tecnologie ad essere raggruppate per giorni;
- **Allocazione delle risorse:** creazione di un algoritmo in grado di automatizzare il processo di allocazione delle risorse sui vari progetti.

L'applicazione è rivolta e pensata per la risorsa aziendale che ha, tra le sue responsabilità, quella di assegnare le risorse ai progetti dei clienti e tenere traccia di queste.

3.2.1 Requisiti aggiuntivi in corso di progetto

Durante il processo di sviluppo dell'applicazione, il destinatario finale ha avanzato le seguenti proposte di modifica rispetto alla specifica iniziale, con lo scopo di migliorare l'applicazione e arricchire il progetto complessivo:

- Aggiunta di un campo di testo opzionale alla schermata di inserimento delle attività, nel quale l'utente abbia la possibilità aggiungere delle note dedicate;
- Possibilità di selezionare per la stessa attività più mesi, in modo tale da rendere più efficiente l'inserimento delle stesse informazioni;
- Report riassuntivo sulle tecnologie espresso in FTE, anziché in giorni;
- Possibilità per l'utente di aggiungere in autonomia ulteriori tecnologie, che vengano archiviate direttamente nella tabella di riferimento.

3.3 Base dati dell'applicazione

La costruzione delle tabelle relazionali, contenute nel database Oracle, costituisce il punto di partenza essenziale per lo sviluppo di un'applicazione Oracle APEX. Le tabelle sono strutture del database organizzate in righe e colonne, che vengono progettate per memorizzare e gestire i dati in maniera efficiente. La loro importanza risiede nel fatto che fungono da deposito in cui fluiscono e vengono archiviati i dati utilizzati nell'applicazione.

Ogni tabella è composta da colonne, che determinano i diversi tipi di dati che possono essere memorizzati al suo interno. I tipi di dati sono molteplici (stringhe, numeri, date, ecc) e vengono definiti sulla base delle specifiche esigenze del sistema. Le tabelle vengono progettate in base al modello dei dati dell'applicazione, il quale è strettamente connesso ai requisiti funzionali e strutturali dell'applicazione stessa. Nel caso specifico dell'applicazione MM Macropianificazione, sono state definite le seguenti tabelle:

- ***T_MACROPIANIFICAZIONE***: tabella destinata a contenere la pianificazione delle attività nei mesi;
- ***T_A_ANNI_MESI***: tabella di anagrafica, contenente la data nel formato MM/YYYY;
- ***T_A_TECNOLOGIE***: tabella di anagrafica nella quale vengono inserite le tecnologie utilizzate dall'azienda;
- ***T_A_ATTENDIBILITA***: tabella di anagrafica contenente le percentuali di attendibilità dei giorni richiesti per il completamento dei progetti;
- ***T_A_INFO_RISORSE***: tabella di anagrafica contenente le informazioni sulle risorse aziendali;
- ***T_A_INFO_CLIENTI***: tabella di anagrafica contenente le informazioni sui clienti;
- ***T_ALLOCAZIONE***: tabella contenente le allocazioni delle risorse sulle attività.

Per tenere traccia nel tempo della creazione e degli aggiornamenti dei record contenuti nelle tabelle, tutte le tabelle contengono i seguenti quattro campi:

- ***CREATION_BY***: denota il nome utente che ha creato il record a cui è associato;
- ***CREATION_DATE***: denota la data di creazione del record a cui è associato;
- ***LAST_UPDATE_BY***: denota il nome utente che ha aggiornato per ultimo il record a cui è associato;
- ***LAST_UPDATE_DATE***: denota la data dell'ultimo aggiornamento del record.

Ogni tabella contiene anche un campo denominato *FLAG_CANCELLED* di tipo VARCHAR2, che può assumere i valori 'Y' o 'N'. Tale campo viene utilizzato per tenere traccia delle cancellazioni che vengono effettuate durante l'utilizzo dell'applicazione: assume il valore 'Y' quando una riga è stata cancellata e 'N' quando la riga è ancora attiva. Questa implementazione consente di eseguire delle cancellazioni di tipo logico e non fisico, garantendo, dunque, la permanenza delle informazioni nella tabella.

Le operazioni di inserimento e di modifica dei campi appena trattati vengono gestite in modo automatico grazie alla dichiarazione dei Trigger. I Trigger sono, infatti, unità PL/SQL memorizzate nel database che, se abilitate, vengono eseguite automaticamente in risposta ad un evento specificato. [25]

Di seguito viene mostrato un esempio di Trigger implementato, relativo alla tabella T_ALLOCAZIONE:

Trigger T_ALLOCAZIONE

```
create or replace trigger "TRG_T_ALLOCAZIONE_BIU"
BEFORE
insert or update on "T_ALLOCAZIONE"
for each row
begin
if inserting then
    if :new.id_allocazione is null then
        select s_t_allocazione.nextval
        into :new.id_allocazione
        from dual;
    end if;
    :new.creation_by := v('app_user');
    :new.creation_date := sysdate;
    :new.last_update_by := v('app_user');
    :new.last_update_date := sysdate;
    :new.flag_cancelled := 'N';
elseif updating then
    :new.last_update_by := v('app_user');
    :new.last_update_date := sysdate;
end if;
end;
```

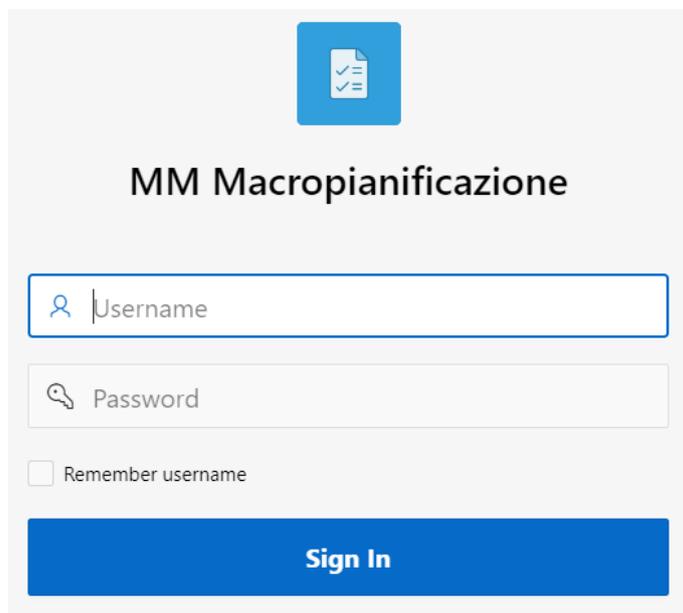
3.4 Funzionamento dell'applicazione

In questo paragrafo verrà approfondito il funzionamento dell'applicazione MM Macropianificazione, analizzando nel dettaglio le pagine web realizzate per l'implementazione di tale soluzione.

Si intende sottolineare che le informazioni relative ai clienti e alle risorse dell'azienda Mediamente Consulting, in quanto sensibili, sono state opportunamente oscurate, al fine di preservarne i contenuti.

L'applicazione MM Macropianificazione, una volta completata l'autenticazione attraverso le proprie credenziali e verificata la sua validità, consente l'accesso ad una serie di pagine mediante un menu di navigazione laterale. Queste includono:

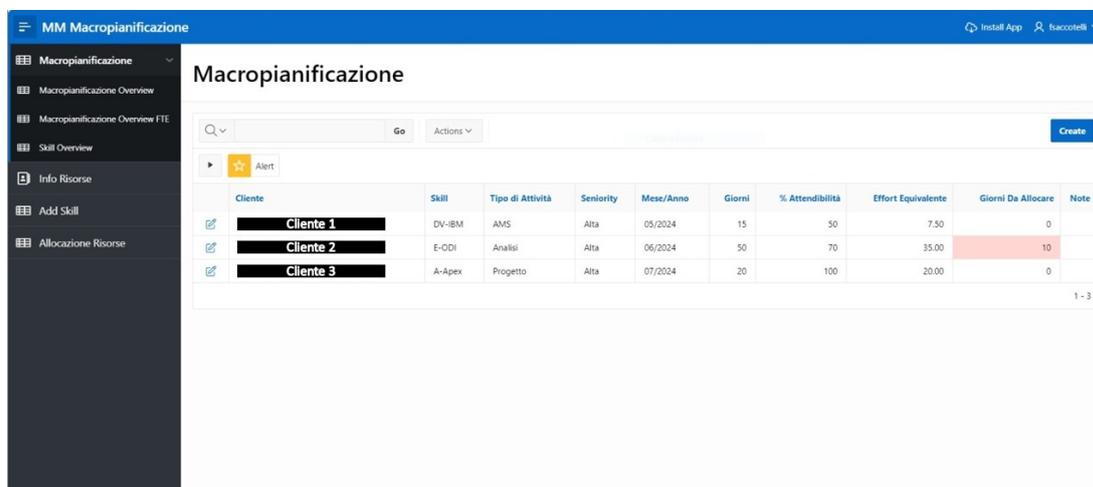
- ***Home Page Macropianificazione***: dedicata alla pianificazione delle attività aziendali;
- ***Macropianificazione Overview, Macropianificazione Overview FTE, Skill Overview***, tre pagine riassuntive contenenti rispettivamente le informazioni relative alle attività pianificate e alle tecnologie utilizzate per completarle;
- ***Info Risorse***: relativa alla gestione delle risorse aziendali;
- ***Add Skill***: in cui l'utente può consultare un elenco delle tecnologie attualmente disponibili nell'azienda e, all'occorrenza, integrarne di nuove;
- ***Allocazione Risorse***: che consente di allocare le risorse disponibili alle specifiche attività e visualizzarne i risultati attraverso due tipologie di report.



The login form features a blue document icon at the top. Below it, the title "MM Macropianificazione" is centered. The form includes a "Username" input field with a user icon, a "Password" input field with a key icon, and a "Remember username" checkbox. A prominent blue "Sign In" button is located at the bottom.

Figura 3.1. Login MM Macropianificazione.

3.4.1 Home Page Macropianificazione



The screenshot shows the "Macropianificazione" home page. It includes a sidebar with navigation options like "Macropianificazione Overview", "Skill Overview", and "Info Risorse". The main content area displays a table with columns for Cliente, Skill, Tipo di Attività, Seniority, Mese/Anno, Giorni, % Attendibilità, Effort Equivalente, Giorni Da Allocare, and Note. A table with 10 columns and 3 rows is shown below the navigation menu. The second row has a red background for the 'Giorni Da Allocare' cell.

| Cliente | Skill | Tipo di Attività | Seniority | Mese/Anno | Giorni | % Attendibilità | Effort Equivalente | Giorni Da Allocare | Note |
|-----------|--------|------------------|-----------|-----------|--------|-----------------|--------------------|--------------------|------|
| Cliente 1 | DV-IBM | AMS | Alta | 05/2024 | 15 | 50 | 7.50 | 0 | |
| Cliente 2 | E-ODI | Analisi | Alta | 06/2024 | 50 | 70 | 35.00 | 10 | |
| Cliente 3 | A-Apex | Progetto | Alta | 07/2024 | 20 | 100 | 20.00 | 0 | |

Figura 3.2. Home Page Macropianificazione.

La prima pagina in cui verrà reindirizzato l'utente finale una volta effettuato il login è l'home page *Macropianificazione*, la quale, come precedentemente espresso, è la pagina dedicata alla pianificazione delle attività. Tale pagina è stata creata

utilizzando il componente *Interactive Report* di APEX associato ad un *form*. Tale integrazione consente di fornire all'utente finale un'esperienza completa di manipolazione dei dati. In particolare, l'Interactive Report prevede una visualizzazione dei dati in forma tabellare, con la possibilità per l'utente di personalizzare il layout secondo diverse modalità: applicando filtri e ordinamenti, nascondendo o mostrando specifiche colonne, evidenziando particolari righe o celle della tabella. Inoltre, l'utente potrà eseguire operazioni avanzate come il raggruppamento dei dati, la creazione di grafici e l'aggiunta di calcoli. [26]

Contestualmente, il form, presente nella sezione destra della pagina, consente all'utente di inserire nuove righe, modificare ed eliminare quelle esistenti direttamente dalla stessa pagina *Macropianificazione*. L'accesso a queste funzionalità avviene rispettivamente attraverso il pulsante *Create* per l'inserimento di nuove attività e tramite l'icona *Edit* presente su ogni riga del report per la modifica o l'eliminazione delle attività esistenti.

| Cliente | Skill | Tipo di Attività | Seniority | Mese/Anno |
|-----------|--------|------------------|-----------|-----------|
| Cliente 1 | DV-IBM | AMS | Alta | 05/2024 |
| Cliente 2 | E-ODI | Analisi | Alta | 06/2024 |
| Cliente 3 | A-Apex | Progetto | Alta | 07/2024 |

Figura 3.3. Form Home Page Macropianificazione.

Da un punto di vista implementativo, sono state sviluppate due pagine distinte all'interno dell'applicazione. La prima pagina è dedicata alla gestione dell'Interactive Report, mentre la seconda pagina è adibita alla gestione del form. Si intende osservare che è possibile creare queste pagine simultaneamente durante il processo di sviluppo, grazie all'attivazione di un flag nel corso della configurazione del report.

Il form, per l'inserimento di un'attività, richiede da parte dell'utente il completamento dei seguenti campi:

- **Cliente**: il nome del cliente per cui l'attività è pianificata;
- **Skill**: la tecnologia necessaria per svolgere l'attività;
- **% Attendibilità**: la percentuale di attendibilità relativa ai giorni previsti per completare l'attività;
- **Tipo di attività**: la tipologia di attività che deve essere svolta;
- **Seniority**: il livello di esperienza richiesto per eseguire l'attività;
- **Mese/Anno**: il periodo in cui l'attività deve essere svolta;
- **Giorni**: il numero di giorni stimati necessari per completare l'attività;
- **Note**: un campo per eventuali informazioni aggiuntive relative all'attività.

Tutti i campi, tranne quello di note il cui completamento è opzionale, sono stati resi obbligatori per l'inserimento di un'attività. Il nome identificativo dei campi del form può essere stabilito tramite il Page Designer, modificando la label presente nell'Identification del Property Editor.

Per implementare un form, è necessario specificare la tabella del database o la query SQL su cui si basa. Nello specifico, tale form si basa sulla tabella T_MACROPIANIFICAZIONE, rappresentata in figura.

| Column Name | Data Type | Nullable | Default | Primary Key |
|------------------------|----------------|----------|---------|-------------|
| ID_MACROPIANIFICAZIONE | NUMBER | No | - | 1 |
| CLI_COD | VARCHAR2(400) | Yes | - | - |
| ID_TECNOLOGIA | NUMBER | Yes | - | - |
| ID_ATTENDIBILITA | NUMBER | Yes | - | - |
| TIPO_ATTIVITA | VARCHAR2(800) | Yes | - | - |
| SENIORITY | VARCHAR2(400) | Yes | - | - |
| MESE_ID | NUMBER | Yes | - | - |
| GIORNI_EFFORT | NUMBER | Yes | - | - |
| FLAG_CANCELLED | VARCHAR2(1) | Yes | - | - |
| CREATION_BY | VARCHAR2(400) | Yes | - | - |
| CREATION_DATE | DATE | Yes | - | - |
| LAST_UPDATE_BY | VARCHAR2(400) | Yes | - | - |
| LAST_UPDATE_DATE | DATE | Yes | - | - |
| GIORNI_RIMANENTI | NUMBER | Yes | - | - |
| NOTE | VARCHAR2(2000) | Yes | - | - |

Figura 3.4. Tabella T_MACROPIANIFICAZIONE.

Le righe inserite, aggiornate ed eliminate dall'utente attraverso il form vengono memorizzate, modificate e cancellate logicamente nella tabella di riferimento corrispondente.

La cancellazione logica dei record, ovvero l'aggiornamento del campo FLAG_CANCELLED a 'Y', è stata gestita tramite la creazione di un processo, denominato *Delete Macropianificazione*, il quale si verifica alla pressione del bottone *Delete*. Per tale bottone, perciò, è stata selezionata come azione del database un'azione di aggiornamento, ovvero una SQL UPDATE action.

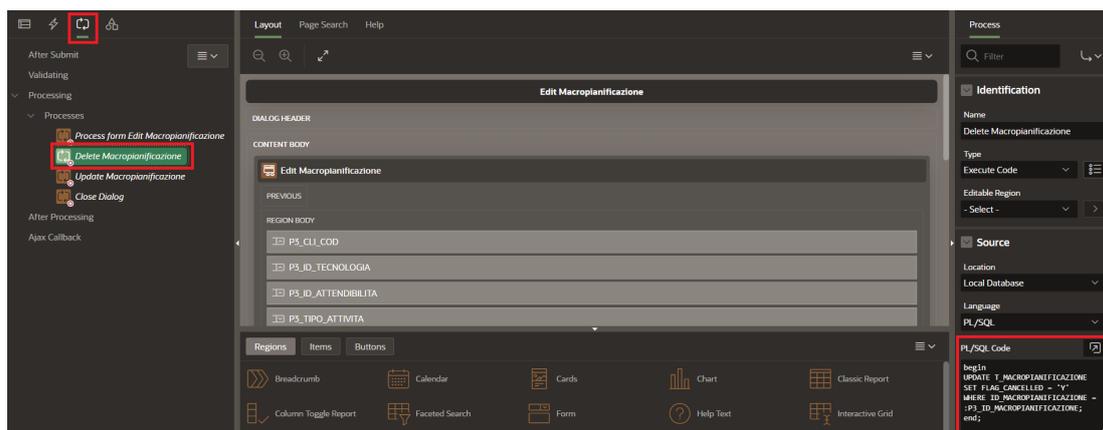


Figura 3.5. Delete Macropianificazione.

Si sottolinea che, nella tabella T_MACROPIANIFICAZIONE, i campi CLI_COD, ID_TECNOLOGIA, ID_ATTENDIBILITA, MESE_ID svolgono il ruolo di chiavi esterne. Una chiave esterna, elemento fondamentale di un database relazionale, è un campo, o una combinazione di campi, di una tabella che stabilisce un collegamento con la chiave primaria di un'altra tabella. Tale relazione consente di creare connessioni tra le tabelle, garantendo che i dati siano coerenti e accurati. Dunque, l'uso di chiavi esterne assicura l'integrità referenziale dei dati, una maggiore coerenza e coesione nelle relazioni tra i dati memorizzati in tabelle diverse. [27] I campi ID_MACROPIANIFICAZIONE, FLAG_CANCELLED, CREATION_BY, CREATION_DATE, LAST_UPDATE_BY, LAST_UPDATE_DATE non sono mostrati tra i campi del form. Sono stati, infatti, settati come tipo *Hidden*, pertanto non sono visibili all'utente finale durante la compilazione, in quanto vengono definiti dal trigger relativo alla tabella T_MACROPIANIFICAZIONE. Anche il campo GIORNI_RIMANENTI è stato impostato come *Hidden*, poichè, al momento di creazione di un'attività, viene gestito con un valore di inserimento pari ai giorni inseriti dall'utente, corrispondenti al campo GIORNI_EFFORT, ai fini del corretto funzionamento dell'algoritmo di allocazione delle risorse, di cui si discuterà in uno dei successivi paragrafi.

Per i campi CLI_COD, ID_TECNOLOGIA, ID_ATTENDIBILITA, MESE_ID è stato scelto nell'Identification il tipo *Popup LOV*; ciò significa che, quando un utente interagisce con il corrispettivo campo del form, si aprirà un pop-up che mostrerà un elenco di valori predefiniti (risultato di una query SQL, in cui vengono selezionate le informazioni da tabelle già esistenti o precedentemente create) tra cui egli potrà scegliere. Si prende come esempio il campo ID_TECNOLOGIA, per mostrare le impostazioni e la query presenti nel Property Editor che consentono di restituire la lista di valori disponibili e selezionabili dall'utente finale.

Figura 3.6. Popup LOV Skill.

La query seleziona due colonne dalla tabella T_A_TECNOLOGIE creata:

- *tipologia_tecnologia*, identificato con l'alias *d* per indicare che si tratta del valore mostrato a video quando l'utente seleziona un'opzione nell'elenco di valori;
- *id_tecnologia*, rinominato come *r*, che rappresenta il valore effettivo che sarà memorizzato nella tabella T_MACROPIANIFICAZIONE quando l'utente seleziona quell'opzione.

Dal form, quando si crea un'attività, è stata data la possibilità di selezionare dal campo *Mese/Anno* più valori dalla lista, così da rendere maggiormente rapida la pianificazione di attività che richiedono diversi mesi per essere completate. Ciò è stato realizzato spuntando il flag *Multiple Values* e specificando `:` come separatore di stringa per riconoscere e suddividere nell'inserimento i valori multipli selezionati. Questo equivale a dire che, quando l'utente creerà un'attività selezionando più mesi, verrà inserita nel report l'attività relativa a ciascun mese in righe separate, così da facilitare un eventuale aggiornamento dei giorni in un mese specifico. L'inserimento con suddivisione dei mesi in righe separate è stato gestito tramite l'implementazione del seguente codice PL/SQL:

Inserimento T_MACROPIANIFICAZIONE

```
DECLARE
    v_id_macropianificazione NUMBER;
BEGIN
    -- P3_MESE_ID contiene una lista di mesi separati da ':'
    FOR line IN (SELECT *
                FROM TABLE(apex_string.split(:P3_MESE_ID, ':')))
    LOOP
        SELECT S_T_MACROPIANIFICAZIONE.NEXTVAL
        INTO v_id_macropianificazione
        FROM DUAL;
        -- Inserisci nella tabella T_MACROPIANIFICAZIONE
        INSERT INTO T_MACROPIANIFICAZIONE VALUES (
            v_id_macropianificazione,
            :P3_CLI_COD, :P3_ID_TECNOLOGIA,
            :P3_ID_ATTENDIBILITA, :P3_TIPO_ATTIVITA,
            :P3_SENIORITY,line.COLUMN_VALUE,
            :P3_GIORNI_EFFORT,'','','','',' ',
            :P3_GIORNI_EFFORT,:P3_NOTE);
    END LOOP;
    COMMIT;
END;
```

Un altro modo per dare la possibilità all'utente di selezionare un valore da una lista di valori predefinita è scegliere nel Property Editor dell'elemento selezionato il tipo *Select list* nell'Identification e il tipo *Static Values* nella List of Values. L'inserimento dei valori mostrati nell'elenco avviene tramite inserimento manuale direttamente nella configurazione dell'elemento. Ciò è stato implementato per i campi TIPO_ATTIVITA e SENIORITY, in quanto costituiti da un numero limitato di valori.

Figura 3.7. Select List con Static Values SENIORITY.

Quando l'utente avrà compilato tutti i campi del form e avrà cliccato sul bottone *Create*, verranno memorizzati i dati nella tabella T_MACROPIANIFICAZIONE e inserita la suddetta riga nel report interattivo. A sua volta, il report interattivo può basarsi su una tabella del database o su una query SQL personalizzata. In questo caso, si è reso necessario implementare la seguente query SQL:

Query report Macropianificazione

```
select m.id_macropianificazione, m.cli_cod,
       m.id_tecnologia, t.TIPOLOGIA_TECNOLOGIA as tecnologia,
       m.ID_ATTENDIBILITA, att.PERCENTUALE as attendibilita,
       m.TIPO_ATTIVITA, m.SENIORITY,
       TO_CHAR(TO_DATE(MESE_ID, 'YYYYMM'), 'MM/YYYY')
       AS mese_anno, m.FLAG_CANCELLED, cl.cli_des, m.GIORNI_EFFORT,
       ((att.PERCENTUALE/100) * m.GIORNI_EFFORT)
       as Effort_Equivalente, m.GIORNI_RIMANENTI, M.NOTE
from T_MACROPIANIFICAZIONE m, T_A_TECNOLOGIE t,
     T_A_ATTENDIBILITA att, tabella_clienti cl
where m.id_tecnologia = t.id_tecnologia
     and m.id_attendibilita = att.id_attendibilita
     and m.cli_cod = cl.cli_cod and m.flag_cancelled = 'N'
order by cl.cli_des, m.mese_id
```

Tale query consente di visualizzare i dati contenuti nella tabella T_MACROPIANIFICAZIONE, ordinati per cliente e mese di riferimento e un'informazione aggiuntiva richiesta dall'utente finale, ovvero il calcolo dell'effort equivalente. L'effort equivalente rappresenta il numero di giorni stimati per una data attività tenuto conto della percentuale di attendibilità riferita alla stessa. Attraverso le funzionalità integrate nel report interattivo per la personalizzazione del layout, sono state effettuate delle modifiche volte a focalizzare l'attenzione sulle informazioni fondamentali per la pianificazione delle attività. In particolare, sono state nascoste alcune colonne non necessarie all'utente, come l'id della tabella T_MACROPIANIFICAZIONE, denominato ID_MACROPIANIFICAZIONE. Inoltre, si è scelto di evidenziare le celle della colonna *Giorni Da Allocare*, in base a una specifica condizione. Tramite il bottone *Actions*, gli utenti hanno la possibilità di personalizzare la visualizzazione del report cliccando su *Columns* per modificare le colonne visualizzate o su *Format*, seguito da *Highlight*, per mettere in rilievo particolari informazioni. Nello specifico, è stata scelta l'opzione di evidenziare soltanto le celle della colonna *Giorni Da Allocare* che contengono un numero di giorni diverso da zero. Tale approccio ha lo scopo di facilitare la rapida identificazione delle attività già allocate ad una risorsa, contrapposte a quelle che necessitano ancora di assegnazione.

The image shows a configuration window for highlighting data. At the top, there's a header with a play button, a star icon, and the text 'Alert'. Below this is a title bar 'Highlight' with a close button. The main area is divided into several sections: 'Sequence' (input: 10), 'Name' (input: Alert), 'Highlight Type' (dropdown: Cell), and 'Enabled' (checkbox: checked). The 'Highlight Style' section has 'Background Color' (input: #ffd6d2), 'Text Color' (input: empty), and a 'Preview' box showing 'Aa' on a pink background. The 'Highlight Condition' section has 'Column' (dropdown: Giorni Da Allocare), 'Operator' (dropdown: !=), and 'Expression' (input: 0). At the bottom right are three buttons: 'Cancel', 'Delete', and 'Apply'.

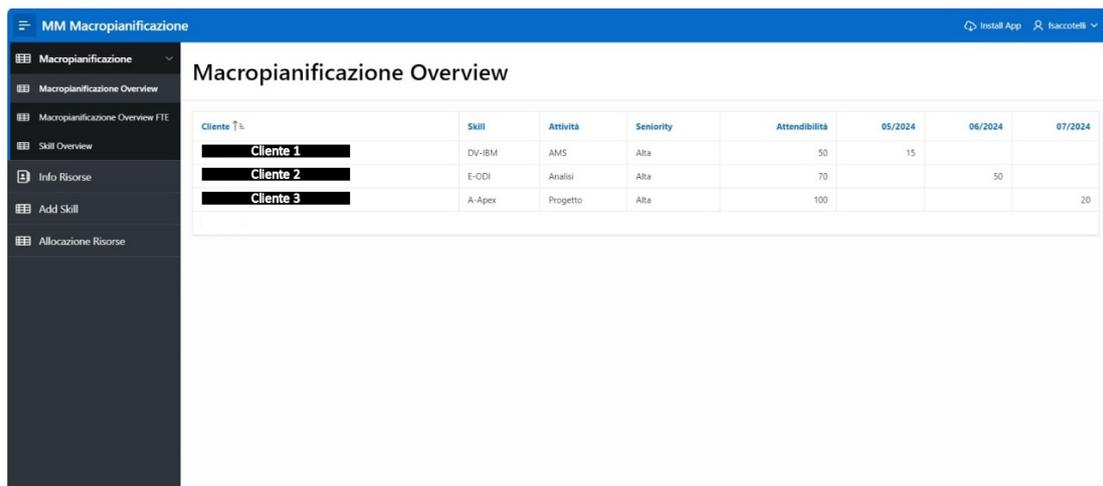
Figura 3.8. Highlight Giorni Da Allocare.

3.4.2 Macropianificazione Overviews

Una volta create le attività con i rispettivi giorni stimati per completarle, è possibile visualizzare tali informazioni in tre report riassuntivi. Tali report consentono di riportare ciascuna attività (nel caso delle pagine *Macropianificazione Overview* e *Macropianificazione Overview FTE*) o tecnologia (nel caso della pagina *Skill Overview*) in un'unica riga, al fine di visualizzarne la distribuzione dei giorni stimati nel tempo.

Ogni pagina è stata realizzata mediante l'uso di un *Classic Report*. In modo analogo agli *Interactive Reports*, il *Classic Report* consente di presentare i dati in forma tabellare, tuttavia senza offrire all'utente finale la possibilità di personalizzazione avanzata (ad eccezione dell'ordinamento e di semplici filtri). Anche la creazione di un *Classic Report* richiede la definizione della tabella su cui costruire il report, oppure l'implementazione di un'istruzione SQL personalizzata o ancora di una funzione PL/SQL che restituisca l'istruzione SQL necessaria. In questa specifica implementazione, è stata scelta la terza opzione menzionata, in quanto si è reso necessario un approccio dinamico per la generazione dei report.

Macropianificazione Overview



| Cliente ↑ | Skill | Attività | Seniority | Attendibilità | 05/2024 | 06/2024 | 07/2024 |
|-----------|--------|----------|-----------|---------------|---------|---------|---------|
| Cliente 1 | DV-IBM | AMS | Alta | 50 | 15 | | |
| Cliente 2 | E-ODI | Analisi | Alta | 70 | | 50 | |
| Cliente 3 | A-Apex | Progetto | Alta | 100 | | | 20 |

Figura 3.9. Macropianificazione Overview.

Il primo report è stato progettato con l'obiettivo di fornire una visione chiara e dettagliata delle attività pianificate, al fine di semplificare la comprensione e l'analisi dei dati.

Di seguito si riporta la funzione PL/SQL implementata per la creazione di tale report.

```

CREATE OR REPLACE FUNCTION get_query_macropianificazione_pivot
RETURN VARCHAR2 IS
  l_alternative VARCHAR2(4000);
  l_sqlquery CLOB;
  l_select_list VARCHAR2(4000);
  l_pivot_cols VARCHAR2(4000);
BEGIN
  SELECT '**** || LISTAGG(m.mese_id, ''', ''') WITHIN GROUP (ORDER BY m.mese_id) || ****'
  INTO l_alternative
  FROM t_macropianificazione m
  WHERE m.flag_cancelled = 'N';
  l_select_list := '';
  l_pivot_cols := '';
  FOR line IN (SELECT DISTINCT m.mese_id
               FROM t_macropianificazione m
               WHERE m.mese_id IN ( SELECT TO_NUMBER(TO_CHAR(ADD_MONTHS(TO_CHAR(SYSDATE, 'YYYYMM'), 'YYYYMM'), LEVEL - 1), 'YYYYMM') AS MeseAnno
                                   FROM dual
                                   CONNECT BY LEVEL <= 9)
               AND m.flag_cancelled = 'N'
               ORDER BY m.mese_id) LOOP
    l_select_list := l_select_list || TO_CHAR(TO_DATE(line.mese_id, 'YYYYMM'), 'MM/YYYY') || ', ';
    l_pivot_cols := l_pivot_cols || line.mese_id || ' AS ' || TO_CHAR(TO_DATE(line.mese_id, 'YYYYMM'), 'MM/YYYY') || ', ';
  END LOOP;
  l_pivot_cols := RTRIM(l_pivot_cols, ', ');
  l_sqlquery :=
  SELECT '**** || ' || l_select_list || '
  FROM (SELECT cl.cli_des AS Cliente, t.tipologia_tecnologia AS Skill, m.tipo_attivita AS Attività, m.seniority AS Seniority,
        aa.percentuale AS Attendibilità, m.mese_id, m.giorni_effort
        FROM t_macropianificazione m, t_a_attendibilita aa, t_a_tecnologie t, tabella_clienti cl
        WHERE m.cli_cod = cl.cli_cod AND m.id_attendibilita = aa.id_attendibilita AND m.id_tecnologia = t.id_tecnologia AND m.flag_cancelled = 'N')
        ) PIVOT (
        SUM(giorni_effort)
        FOR mese_id IN (' || l_pivot_cols || ')
        ) ORDER BY 1 ASC';
  RETURN l_sqlquery;
END;

```

Figura 3.10. Funzione `get_query_macropianificazione_pivot()`.

Si precisa che `tabella_clienti` è una tabella aziendale già presente nel database, contenente le informazioni relative ai clienti.

La funzione `get_query_macropianificazione_pivot()` organizza in una singola riga le informazioni relative ad un'attività, ovvero il cliente coinvolto, la tecnologia utilizzata, il tipo di attività da svolgere, la seniority della risorsa richiesta e l'attendibilità dei giorni stimati. Inoltre, utilizzando la clausola PIVOT all'interno della query SQL generata, la funzione aggiunge a quella specifica attività i mesi come colonne nel report. All'interno di queste colonne vengono forniti i giorni stimati per ciascun mese, consentendo all'utente di visualizzare come sono distribuiti nel tempo e, conseguentemente, di comprendere i livelli di impegno delle attività nel corso dei mesi.

La funzione è stata anche progettata per gestire dinamicamente l'integrazione di nuovi mesi ai dati di pianificazione, senza richiedere modifiche manuali alla query o al report stesso. Per una data attività, vengono, infatti, mostrati i mesi progressivamente inseriti nella tabella T_MACROPIANIFICAZIONE, garantendo una visualizzazione sempre aggiornata delle informazioni disponibili. Il fatto di non conoscere a priori il numero di colonne del report è stato gestito spuntando il flag *Use Generic Columns* presente nel Property Editor della regione *Macropianificazione view* e impostando un numero massimo di colonne generiche per il report. Una volta abilitata tale opzione, Oracle APEX genera in automatico le colonne generiche per il report, basandosi sulle colonne disponibili nella query SQL sottostante. Le colonne della query sono state opportunamente rinominate, al fine di utilizzare come intestazioni delle colonne del report il nome delle colonne stesse.

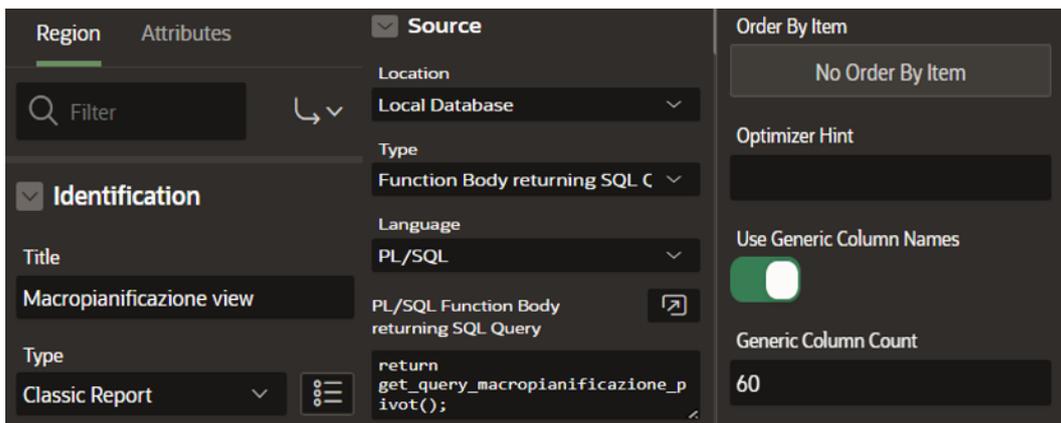


Figura 3.11. Impostazioni per la gestione dinamica dei mesi.

Al fine di rispettare i requisiti stabiliti in fase di progettazione e migliorare la leggibilità delle informazioni, la presentazione dei mesi ha inizio dal mese in corso e si estende fino ai successivi nove mesi.

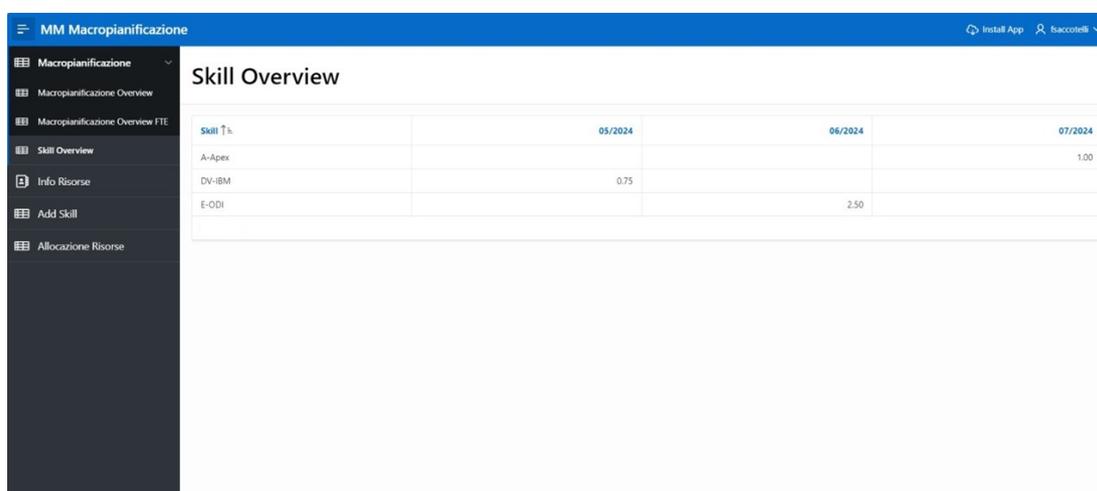
Macropianificazione Overview FTE

Il secondo report è stato implementato con l'intento di presentare le medesime informazioni contenute nella pagina Macropianificazione Overview, ma adottando un'unità di misura alternativa per esprimere i giorni. In questa configurazione, invece di fare diretto riferimento ai giorni lavorati, si utilizza l'unità di misura nota

come Full Time Equivalent (FTE). L'FTE è calcolato dividendo le ore lavorate in media da un dipendente con le ore di lavoro di un dipendente a tempo pieno. Di conseguenza, una risorsa impegnata a tempo pieno è rappresentata come 1 FTE. [28] Nello specifico, in tale contesto 1 FTE corrisponde a 20 giorni lavorativi.

La funzione `get_query_macropianificazione_fte_pivot()` si distingue dalla funzione precedentemente discussa per un unico aspetto: la conversione dei giorni in FTE. Questa conversione viene effettuata dividendo il numero di giorni mensili per 20, rappresentante il numero di giorni lavorativi associati a un FTE.

Skill Overview



| Skill ↑% | 05/2024 | 06/2024 | 07/2024 |
|----------|---------|---------|---------|
| A-Apex | | | 1.00 |
| DV-IBM | 0.75 | | |
| E-ODI | | 2.50 | |

Figura 3.12. Skill Overview.

Il terzo report, invece, riporta un riassunto dei giorni stimati, espressi in FTE, per ciascuna tecnologia richiesta nei mesi. La funzione `get_query_skill_pivot()` segue lo schema delle funzioni precedentemente esaminate, raccogliendo come unica informazione la tecnologia attribuita alle attività.

```

CREATE OR REPLACE FUNCTION get_query_skill_pivot
RETURN VARCHAR2 IS
  l_alternative VARCHAR2(4000);
  l_sqlquery CLOB;
  l_select_list VARCHAR2(4000);
  l_pivot_cols VARCHAR2(4000);
BEGIN
  SELECT '|| LISTAGG(m.mese_id, ''||') WITHIN GROUP (ORDER BY m.mese_id) || ''
  INTO l_alternative
  FROM t_macropianificazione m
  WHERE m.flag_cancelled = 'N';
  l_select_list := '*';
  l_pivot_cols := '';
  FOR line IN (SELECT DISTINCT m.mese_id
               FROM t_macropianificazione m
               WHERE m.mese_id IN ( SELECT TO_NUMBER(TO_CHAR(ADD_MONTHS(TO_DATE(TO_CHAR(SYSDATE, 'YYYYMM'), 'YYYYMM'), LEVEL - 1), 'YYYYMM')) AS MeseAnno
                                   FROM dual
                                   CONNECT BY LEVEL <= 9 )
               AND m.flag_cancelled = 'N'
               ORDER BY m.mese_id) LOOP
    l_select_list := l_select_list || line.mese_id || ' AS ' || TO_CHAR(TO_DATE(line.mese_id, 'YYYYMM'), 'MM/YYYY') || ', ';
  END LOOP;
  l_pivot_cols := RTRIM(l_pivot_cols, ', ');
  l_sqlquery := 'SELECT ' || l_select_list || '
               FROM ( SELECT t.tipologia_tecnologia AS Skill, m.mese_id, m.giorni_effort
                       FROM t_macropianificazione m, t_a_tecnologie t
                       WHERE m.id_tecnologia = t.id_tecnologia AND m.flag_cancelled = ''N'' ) PIVOT ( SUM(giorni_effort/20)
                                                       FOR mese_id IN (' || l_pivot_cols || ')
                                                       ORDER BY 1 ASC);
  RETURN l_sqlquery;
END;

```

Figura 3.13. Funzione `get_query_skill_pivot()`.

3.4.3 Info Risorse

| Risorsa | Seniority | Skill | Priorità Skill Risorsa | Cliente | Priorità Cliente |
|-------------------|-----------|--------|------------------------|-----------|------------------|
| Risorsa 1 | Alta | DV-IBM | 1 | Cliente 1 | 1 |
| Risorsa 2 | Alta | DV-IBM | 1 | Cliente 1 | 1 |
| Risorsa 3 | Alta | DV-IBM | 1 | Cliente 1 | 1 |
| Risorsa 4 | Alta | E-ODI | 1 | Cliente 2 | 1 |
| Risorsa 5 | Alta | E-ODI | 1 | Cliente 2 | 1 |
| Risorsa 6 | Alta | A-Apex | 1 | Cliente 3 | 1 |
| Simone Morassutto | Alta | A-Apex | 1 | Cliente 3 | 1 |

Figura 3.14. Info Risorse.

La pagina denominata *Info Risorse* è la pagina dedicata alla gestione delle risorse. Analogamente alla pagina *Macropianificazione*, essa è stata implementata mediante un *Interactive Report* connesso ad un form.

Tale form, destinato a gestire l'inserimento, la modifica e la cancellazione logica delle voci presenti nel report, prevede i seguenti campi:

- **Risorsa**: il nome della risorsa aziendale;
- **Seniority**: il livello di seniority della risorsa;
- **Skill**: la tecnologia di competenza della risorsa;
- **Priorità Skill Risorsa**: il grado di priorità della risorsa per la tecnologia specificata (con il grado di priorità massimo inteso pari a 1);
- **Cliente**: il cliente associato ad una particolare risorsa;
- **Priorità cliente**: il grado di priorità del cliente rispetto a quella specifica risorsa (con il grado di priorità massimo inteso pari a 1).

In questo caso, le informazioni contenute nel form e nel report provengono dalle tabelle di anagrafica T_A_INFO_RISORSE e T_A_INFO_CLIENTI. Queste tabelle, riunite in una singola riga ai fini della gestione integrata delle informazioni, contengono rispettivamente i dati relativi alle risorse aziendali e ai clienti.

Per la costruzione del form e del report, si è reso dunque necessario implementare le seguenti query SQL di seguito riportare.

```

select distinct r.id_info_risorsa,
r.id_utente,
u.nome_utente,
r.id_tecnologia,
t.tipologia_tecnologia,
r.seniority,
r.priorita_risorsa,
cc.cli_cod,
cl.cli_des,
cc.priorita_cliente
from t_a_info_risorse r, t_a_info_clienti cc, tabella_clienti cl,
t_a_tecnologie t, tabella_risorse u
where r.id_utente = cc.id_utente and cc.cli_cod = cl.cli_cod and
t.id_tecnologia = r.id_tecnologia and u.id_utente = r.id_utente
and r.flag_cancelled = 'N'
and cc.flag_cancelled = 'N' and id_info_risorsa = id_info_cliente
order by u.nome_utente

select r.id_info_risorsa,
r.id_utente,
r.id_tecnologia,
r.seniority,
r.priorita_risorsa,
cc.id_info_cliente,
cc.cli_cod,
cc.priorita_cliente
from t_a_info_risorse r, t_a_info_clienti cc,
tabella_clienti cl, t_a_tecnologie t
where r.id_utente = cc.id_utente and cc.cli_cod = cl.cli_cod and
t.id_tecnologia = r.id_tecnologia and id_info_risorsa = id_info_cliente

```

Figura 3.15. Query Report e Form *Info Risorse*.

I campi *Risorsa*, *Skill* e *Cliente* del form sono configurati come liste di valori di tipo *Popup LOV*, mentre i campi *Priorità Skill Risorsa* e *Priorità Cliente* sono elenchi di valori di tipo *Select List* con *Static Values* (i cui tipi sono già stati precedentemente discussi).

L'inserimento, l'aggiornamento e la cancellazione logica delle informazioni sono operazioni che vengono effettuate coerentemente rispetto alle tabelle destinate a contenere i dati sulle risorse e sui clienti. A tale scopo, sono stati creati tre processi che regolano contemporaneamente le medesime operazioni per le due tabelle *T_A_INFO_RISORSE* e *T_A_INFO_CLIENTI*. Questi processi sono:

- ***Process Form Edit Info Risorse***: gestisce le operazioni di inserimento e si verifica alla pressione del bottone *Create*;

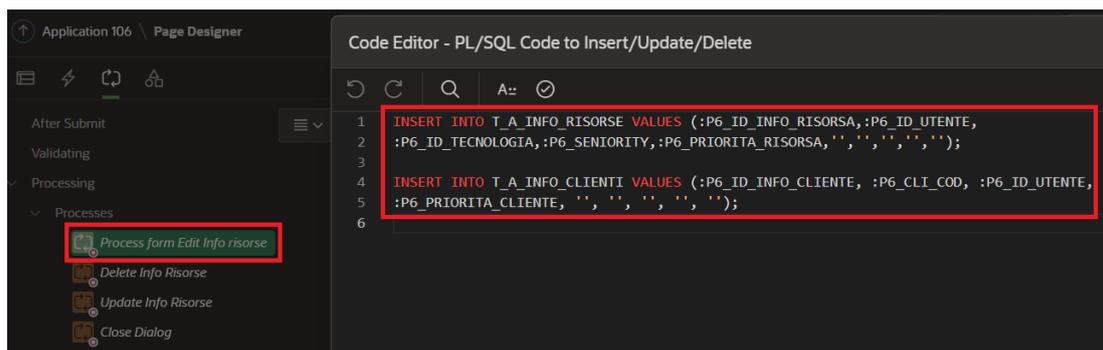


Figura 3.16. Process Form Edit Info Risorse.

- **Delete Info Risorse:** regola le operazioni di cancellazione logica e si verifica alla pressione del bottone *Delete*;

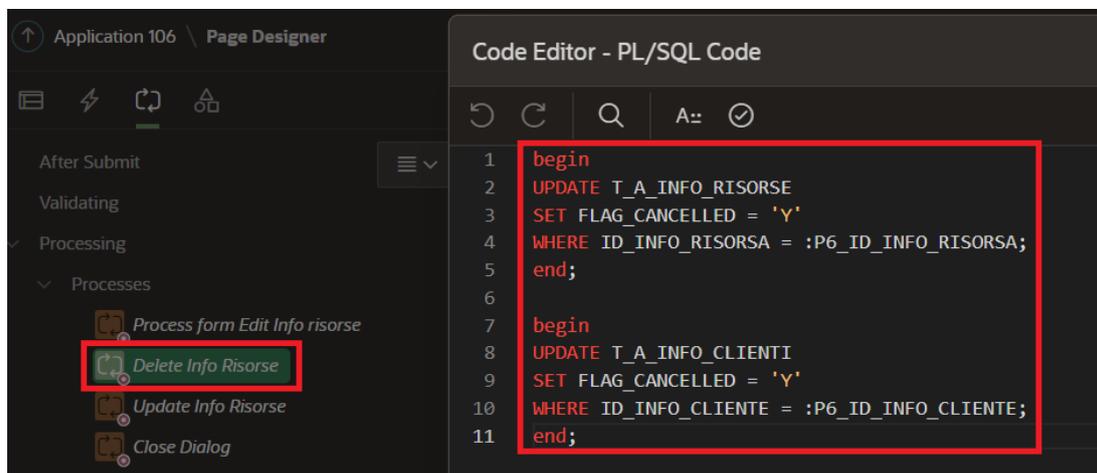


Figura 3.17. Delete Info Risorse.

- **Update Info Risorse:** si occupa delle operazioni di aggiornamento e si verifica alla pressione del bottone *Save*.

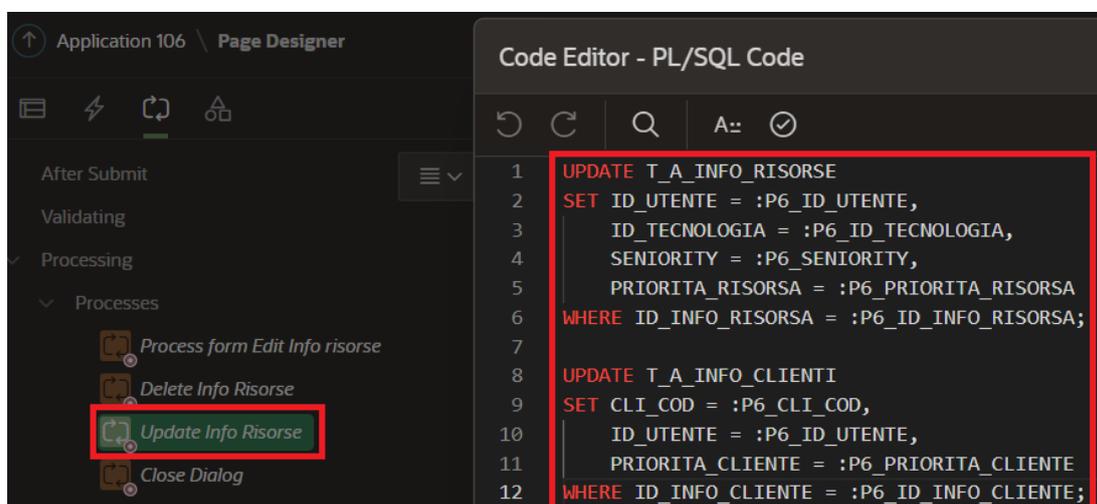


Figura 3.18. Update Info Risorse.

3.4.4 Add Skill

La pagina *Add Skill* è stata concepita con l'obiettivo di fornire all'utente finale la possibilità di aggiungere direttamente dall'interfaccia utente nuove tecnologie impiegate dall'azienda Mediamente Consulting per lo svolgimento delle proprie attività aziendali. Attraverso un *Interactive Report* è possibile visualizzare l'elenco delle tecnologie attualmente in uso, consentendo all'utente di ordinare, filtrare, evidenziare particolari tecnologie di interesse. Il form, incorporato nella stessa pagina, è invece adibito all'inserimento di nuove tecnologie, nonché alla modifica e alla cancellazione logica delle tecnologie già esistenti. Tale form, basato sulla tabella T_A_TECNOLOGIE, presenta infatti come unico campo *Skill*, che identifica il nome della tecnologia.

Il report è generato tramite l'esecuzione della seguente query SQL, la quale seleziona tutte le tecnologie disponibili in ordine alfabetico.

Query Add Skill

```
select ID_TECNOLOGIA, TIPOLOGIA_TECNOLOGIA
from T_A_TECNOLOGIE
where FLAG_CANCELLED = 'N'
order by TIPOLOGIA_TECNOLOGIA
```

3.5 Algoritmo di allocazione delle risorse

Nell'ambito della gestione aziendale, riveste un ruolo di fondamentale importanza un'allocazione efficiente delle risorse alle attività pianificate. Partendo dalla necessità di coordinare le attività presenti nella pagina *Macropianificazione* dedicata alla pianificazione e le risorse disponibili nella pagina *Info Risorse* correlata, è stato implementato un algoritmo, al fine di automatizzare questo processo.

Tale algoritmo ha come obiettivo principale quello di assicurare che le risorse aziendali vengano assegnate in maniera efficiente alle attività pianificate. Ciò è essenziale per massimizzare lo svolgimento delle attività e garantire il completamento tempestivo delle stesse. Nel caso specifico dell'azienda Mediamente Consulting, vista la complessità della gestione delle risorse, è stato realizzato un algoritmo che tiene conto di variabili cruciali come la priorità della risorsa su un determinato cliente, le tecnologie coinvolte e i vincoli di tempo. Per realizzare l'algoritmo si è reso necessario implementare una procedura PL/SQL, ovvero un sottoprogramma che esegue specifiche istruzioni [18], denominata *alloca_risorse*.

È possibile suddividere le istruzioni eseguite dall'algoritmo in tre fasi distinte:

1. Operazioni Generali

La prima fase dell'algoritmo prevede l'esecuzione di alcune istruzioni comuni. Tra queste troviamo:

- **Dichiarazione e inizializzazione delle variabili:** durante l'esecuzione dell'algoritmo, vengono eseguite delle operazioni matematiche, i cui risultati sono memorizzati in specifiche variabili. Queste ultime vengono dichiarate e inizializzate all'interno di tale fase preliminare;

```
create or replace PROCEDURE alloca_risorse AS
  v_giorni_da_allocare INT;
  v_numero_risorse_teoriche INT := 0; -- Contatore per il numero di risorse con stessa priorità
  v_giorni_da_allocare_teorici INT := 0;
  v_numero_risorse_effettive INT := 0;
  v_giorni_da_allocare_effettivi INT := 0;
  v_gg_rimanenti INT := 0;
  v_gg INT := 0;
```

Figura 3.19. Dichiarazione e inizializzazione delle variabili.

- **Aggiornamento:** ogni esecuzione dell'algoritmo di allocazione delle risorse comporta aggiornamenti alle tabelle T_MACROPIANIFICAZIONE e T_ALLOCAZIONE. Questi sono finalizzati a consentire la riallocazione, ovvero la riassegnazione delle risorse alle attività, tenendo conto di nuove attività inserite. Nello specifico, si rivela necessario reimpostare al valore iniziale il campo giorni_rimanenti, ovvero i giorni da allocare, della tabella T_MACROPIANIFICAZIONE ed effettuare la cancellazione logica delle righe presenti nella tabella T_ALLOCAZIONE, al fine di effettuare l'algoritmo su tutte le attività fino a quel momento inserite. Entrambi gli aggiornamenti vengono applicati esclusivamente alle righe relative ai mesi a partire da quello corrente;

```

BEGIN
UPDATE T_MACROPIANIFICAZIONE
SET giorni_rimanenti = giorni_effort
WHERE mese_id in (
SELECT TO_NUMBER(TO_CHAR(ADD_MONTHS(TO_DATE(TO_CHAR(SYSDATE, 'YYYYMM'), 'YYYYMM'), LEVEL - 1), 'YYYYMM')) AS MeseAnno
FROM dual
CONNECT BY LEVEL <= MONTHS_BETWEEN(TO_DATE('203012', 'YYYYMM'), TO_DATE(TO_CHAR(SYSDATE, 'YYYYMM'), 'YYYYMM')) + 1
);

UPDATE T_ALLOCAZIONE
SET flag_cancelled = 'Y'
WHERE mese_id in (
SELECT TO_NUMBER(TO_CHAR(ADD_MONTHS(TO_DATE(TO_CHAR(SYSDATE, 'YYYYMM'), 'YYYYMM'), LEVEL - 1), 'YYYYMM')) AS MeseAnno
FROM dual
CONNECT BY LEVEL <= MONTHS_BETWEEN(TO_DATE('203012', 'YYYYMM'), TO_DATE(TO_CHAR(SYSDATE, 'YYYYMM'), 'YYYYMM')) + 1
);

```

Figura 3.20. Aggiornamento.

- **Selezione delle attività e azzeramento dei contatori:** per allocare le risorse, viene effettuato un ciclo FOR sulle righe restituite da una query, eseguita sulla tabella T_MACROPIANIFICAZIONE. La query seleziona le attività a partire dal mese corrente e le ordina in base all'effort equivalente, in modo decrescente, e al mese, in modo crescente. Tale ordinamento è necessario per selezionare per prime le attività che richiedono un numero maggiore di giorni per essere completate. Una volta selezionate e ordinate le attività, il ciclo FOR itera su ciascuna riga nell'insieme risultante. All'interno del loop, vengono azzerati i contatori relativi al numero di risorse, viene assegnato alla

variabile `v_giorni_da_allocare` il numero di giorni dell'attività e, successivamente, su ciascuna riga vengono eseguite una serie di azioni, affrontate nei punti successivi;

```

-- Elenco attività ordinate per effort equivalente (decrescente) e mese (crescente)
FOR macro_row IN (
  SELECT m.cli_cod, m.id_tecnologia, m.tipo_attivita, m.seniority, m.mese_id, m.giorni_effort, m.giorni_rimanenti
  FROM t_macropianificazione m
  INNER JOIN t_a_attendibilita att ON m.id_attendibilita = att.id_attendibilita
  WHERE m.mese_id in (
    SELECT TO_NUMBER(TO_CHAR(ADD_MONTHS(TO_DATE(TO_CHAR(SYSDATE, 'YYYYMM'), 'YYYYMM'), LEVEL - 1), 'YYYYMM')) AS MeseAnno
    FROM dual
    CONNECT BY LEVEL <= MONTHS_BETWEEN(TO_DATE('203012', 'YYYYMM'), TO_DATE(TO_CHAR(SYSDATE, 'YYYYMM'), 'YYYYMM')) + 1
  )
  ORDER BY ((att.PERCENTUALE/100) * m.GIORNI_EFFORT) DESC, m.mese_id ASC
) LOOP
  v_giorni_da_allocare := macro_row.giorni_effort;
  v_numero_risorse_teoriche := 0; -- Azzerare il contatore all'inizio di ogni iterazione
  v_numero_risorse_effettive := 0; -- Azzerare il contatore all'inizio di ogni iterazione

```

Figura 3.21. Selezione delle attività e azzeramento dei contatori.

- **Calcolo del numero di risorse:** la prima azione da eseguire su ogni attività è capire quante siano le risorse su cui dividere il numero di giorni necessari al completamento dell'attività. Tale calcolo viene effettuato due volte. Nel primo caso, il conteggio viene effettuato sulle risorse che rispettano i requisiti di tecnologia, seniority e cliente imposti dall'attività selezionata e che possiedono priorità massima sulla stessa. Nello specifico, il livello di priorità riguarda sia quello che la risorsa possiede nei confronti del cliente, sia quello sulla tecnologia coinvolta. Nel secondo caso, invece, viene calcolato il numero di risorse che, oltre a soddisfare gli stessi requisiti del primo calcolo, hanno anche una disponibilità tale da completare i giorni a loro assegnati.

```

IF macro_row.giorni_rimanenti > 0 THEN
  -- Calcola il numero di risorse con stessa priorità
  SELECT COUNT(*)
  INTO v_numero_risorse_teoriche
  FROM T_A_INFO_RISORSE r
  JOIN T_A_INFO_CLIENTI cc ON r.id_utente = cc.id_utente
  WHERE cc.id_info_cliente = r.id_info_risorsa AND cc.cli_cod = macro_row.cli_cod
  AND id_info_risorsa = id_info_cliente AND r.id_tecnologia = macro_row.id_tecnologia
  AND r.seniority = macro_row.seniority
  AND (cc.priorita_cliente, r.priorita_risorsa) = (
    SELECT MIN(cc_inner.priorita_cliente), MIN(r_inner.priorita_risorsa)
    FROM T_A_INFO_RISORSE r_inner
    JOIN T_A_INFO_CLIENTI cc_inner ON r_inner.id_utente = cc_inner.id_utente
    WHERE cc_inner.id_info_cliente = r_inner.id_info_risorsa AND cc_inner.cli_cod = macro_row.cli_cod
    AND id_info_risorsa = id_info_cliente AND r_inner.id_tecnologia = macro_row.id_tecnologia
    AND r_inner.seniority = macro_row.seniority);
  v_giorni_da_allocare_teorici := v_giorni_da_allocare/v_numero_risorse_teoriche;
  SELECT COUNT(*)
  INTO v_numero_risorse_effettive
  FROM T_A_INFO_RISORSE r
  JOIN T_A_INFO_CLIENTI cc ON r.id_utente = cc.id_utente
  WHERE cc.id_info_cliente = r.id_info_risorsa AND cc.cli_cod = macro_row.cli_cod
  AND id_info_risorsa = id_info_cliente
  AND r.id_tecnologia = macro_row.id_tecnologia
  AND r.seniority = macro_row.seniority
  AND (cc.priorita_cliente, r.priorita_risorsa) = (
    SELECT MIN(cc_inner.priorita_cliente), MIN(r_inner.priorita_risorsa)
    FROM T_A_INFO_RISORSE r_inner
    JOIN T_A_INFO_CLIENTI cc_inner ON r_inner.id_utente = cc_inner.id_utente
    WHERE cc_inner.id_info_cliente = r_inner.id_info_risorsa AND cc_inner.cli_cod = macro_row.cli_cod
    AND id_info_risorsa = id_info_cliente
    AND r_inner.id_tecnologia = macro_row.id_tecnologia AND r_inner.seniority = macro_row.seniority)
  AND r.id_utente IN (
    SELECT id_utente
    FROM T_ALLOCAZIONE
    WHERE mese_id = macro_row.mese_id
    GROUP BY id_utente
    HAVING SUM(giorni_effort) <= 20 - v_giorni_da_allocare_teorici
  UNION
  SELECT id_utente
  FROM T_A_INFO_RISORSE
);

```

Figura 3.22. Calcolo del numero di risorse.

Dopo l'esecuzione dell'ultima operazione, si procede alla gestione di due scenari distinti. Tali scenari comprendono: una discrepanza nel conteggio dei risultati ottenuti dalla precedente operazione, oppure una loro corrispondenza. In base allo scenario rilevato, verrà adottato un procedimento specifico che condurrà all'allocazione delle risorse. Queste procedure saranno trattate rispettivamente nella fase 2 e nella fase 3.

2. Allocazione con carico di lavoro non uniforme

Nel caso in cui i due conteggi restituiscano due risultati diversi, questo significa che non tutte le risorse, che soddisfano i requisiti richiesti dall'attività in analisi, hanno giorni a sufficienza per assorbire il carico di lavoro proveniente da un'allocazione omogenea tra tutte le risorse con pari priorità. Questo rappresenta il caso in cui l'allocazione ad un'attività viene effettuata in base alla disponibilità. In particolare, viene inizialmente effettuato un ciclo FOR sulle risorse che soddisfano i requisiti dell'attività da allocare, ordinate in base a quelle che, in quel mese, hanno un maggior carico di lavoro già assegnato.

Per ogni risorsa ottenuta da questa query, viene determinato il numero di giorni che possono essere assegnati ad essa e viene, quindi, inserita l'allocazione della risorsa per l'attività nella tabella T_ALLOCAZIONE.

Dopo aver allocato le risorse, i giorni a loro assegnati vengono decrementati dai giorni da allocare, relativi alla specifica attività nella tabella T_MACROPIANIFICAZIONE. In questa circostanza, si inserisce anche il caso in cui una singola risorsa non possa sostenere completamente il carico di lavoro e riceva solo i giorni rimanenti disponibili per l'allocazione.

```

IF v_numero_risorse_effettive <> v_numero_risorse_teoriche THEN
v_gg_rimanenti := macro_row.giorni_effort;
FOR ris_row IN (
SELECT DISTINCT r.id_utente, r.seniority, r.id_tecnologia, r.priorita_risorsa, cc.cli_cod, cc.priorita_cliente, SUM(aa.giorni_effort) AS gg
FROM T_A_INFO_RISORSE r
JOIN T_A_INFO_CLIENTI cc ON r.id_utente = cc.id_utente
JOIN T_ALLOCAZIONE aa ON r.id_utente = aa.id_utente
WHERE cc.id_info_cliente = r.id_info_risorsa AND cc.cli_cod = macro_row.cli_cod
AND id_info_risorsa = id_info_cliente
AND r.id_tecnologia = macro_row.id_tecnologia
AND r.seniority = macro_row.seniority
AND aa.mese_id = macro_row.mese_id
AND (cc.priorita_cliente, r.priorita_risorsa) = (
SELECT MIN(cc_inner.priorita_cliente), MIN(r_inner.priorita_risorsa)
FROM T_A_INFO_RISORSE r_inner
JOIN T_A_INFO_CLIENTI cc_inner ON r_inner.id_utente = cc_inner.id_utente
WHERE cc_inner.id_info_cliente = r_inner.id_info_risorsa AND cc_inner.cli_cod = macro_row.cli_cod
AND id_info_risorsa = id_info_cliente
AND r_inner.id_tecnologia = macro_row.id_tecnologia
AND r_inner.seniority = macro_row.seniority)
GROUP BY r.id_utente, r.seniority, r.id_tecnologia, r.priorita_risorsa, cc.cli_cod, cc.priorita_cliente
HAVING SUM(aa.giorni_effort) < 20
ORDER BY SUM(aa.giorni_effort) DESC
) LOOP
v_gg := LEAST(20 - ris_row.gg, v_gg_rimanenti);
INSERT INTO t_allocazione (id_allocazione, cli_cod, id_tecnologia, tipo_attivita, seniority, mese_id,
id_utente, giorni_effort, flag_cancelled, creation_by, creation_date, last_update_by, last_update_date)
VALUES ('', macro_row.cli_cod, macro_row.id_tecnologia, macro_row.tipo_attivita, macro_row.seniority, macro_row.mese_id,
ris_row.id_utente, v_gg, '', '', '', '', '');
v_gg_rimanenti := v_gg_rimanenti - v_gg;
UPDATE t_macropianificazione
SET giorni_rimanenti = v_gg_rimanenti
WHERE cli_cod = macro_row.cli_cod
AND id_tecnologia = macro_row.id_tecnologia
AND seniority = macro_row.seniority
AND mese_id = macro_row.mese_id;
END LOOP;

```

Figura 3.23. Allocazione con carico di lavoro non uniforme.

3. Allocazione con carico di lavoro uniforme

Nel caso in cui, invece, i due conteggi ottenuti alla fine della fase 1 dell'algoritmo risultino equivalenti, ciò implica che tutte le risorse che soddisfano i requisiti dell'attività hanno la capacità di sopportare il carico di lavoro distribuito in maniera uniforme tra loro.

Questa situazione rappresenta il caso più semplice da gestire, poiché per effettuare l'allocazione è sufficiente iterare tra le risorse che soddisfano i requisiti dell'attività e che hanno a loro già assegnati non più di 20 giorni nel mese. Tale scenario comprende anche il caso di singola risorsa con massima priorità, in grado di sopportare tutto il carico di lavoro. Ad ogni iterazione di questo ciclo più interno, viene eseguita l'allocazione della risorsa attraverso l'inserimento di una nuova riga nella tabella T_ALLOCAZIONE e l'aggiornamento dei giorni da allocare nella tabella T_MACROPIANIFICAZIONE.

```

ELSE
v_giorni_da_allocaire_effettivi := v_giorni_da_allocaire_teorici;
FOR risorse_row IN (
SELECT DISTINCT r.id_utente, r.seniority, r.id_tecnologia, r.priorita_risorsa, cc.cli_cod, cc.priorita_cliente
FROM T_A_INFO_RISORSE r
JOIN T_A_INFO_CLIENTI cc ON r.id_utente = cc.id_utente
WHERE cc.id_info_cliente = r.id_info_risorsa AND cc.cli_cod = macro_row.cli_cod
AND id_info_risorsa = id_info_cliente
AND r.id_tecnologia = macro_row.id_tecnologia
AND r.seniority = macro_row.seniority
AND (cc.priorita_cliente, r.priorita_risorsa) = (
SELECT MIN(cc_inner.priorita_cliente), MIN(r_inner.priorita_risorsa)
FROM T_A_INFO_RISORSE r_inner
JOIN T_A_INFO_CLIENTI cc_inner ON r_inner.id_utente = cc_inner.id_utente
WHERE cc_inner.id_info_cliente = r_inner.id_info_risorsa AND cc_inner.cli_cod = macro_row.cli_cod
AND id_info_risorsa = id_info_cliente
AND r_inner.id_tecnologia = macro_row.id_tecnologia
AND r_inner.seniority = macro_row.seniority)
AND r.id_utente IN (
SELECT id_utente
FROM T_ALLOCAZIONE
WHERE mese_id = macro_row.mese_id
GROUP BY id_utente
HAVING SUM(giorni_effort) <= 20 - v_giorni_da_allocaire_effettivi
UNION
SELECT id_utente
FROM T_A_INFO_RISORSE)
) LOOP
INSERT INTO t_allocazione (id_allocazione, cli_cod, id_tecnologia, tipo_attivita, seniority, mese_id,
id_utente, giorni_effort, flag_cancelled, creation_by, creation_date, last_update_by, last_update_date)
VALUES ('', macro_row.cli_cod, macro_row.id_tecnologia, macro_row.tipo_attivita, macro_row.seniority, macro_row.mese_id,
risorse_row.id_utente, v_giorni_da_allocaire_effettivi, '', '', '', '', '');
UPDATE t_macropianificazione
SET giorni_rimanenti = giorni_rimanenti - v_giorni_da_allocaire_effettivi
WHERE cli_cod = macro_row.cli_cod
AND id_tecnologia = macro_row.id_tecnologia
AND seniority = macro_row.seniority AND mese_id = macro_row.mese_id;
END LOOP;
END IF;
END IF;
END LOOP;
END alloca_risorse;

```

Figura 3.24. Allocazione con carico di lavoro uniforme.

3.5.1 Allocazione Risorse

| Cliente | Skill Attivita | Seniority Mese/Anno | Risorsa | Giorni Assegnati |
|-----------|-------------------|-----------------------|-------------------|------------------|
| Cliente 1 | DV-IBM AMS | Alta 05/2024 | Risorsa 1 | 5 |
| Cliente 1 | DV-IBM AMS | Alta 05/2024 | Risorsa 2 | 5 |
| Cliente 1 | DV-IBM AMS | Alta 05/2024 | Risorsa 3 | 5 |
| Cliente 2 | E-ODI Analisi | Alta 06/2024 | Risorsa 4 | 20 |
| Cliente 2 | E-ODI Analisi | Alta 06/2024 | Risorsa 5 | 20 |
| Cliente 3 | A-Apex Progetto | Alta 07/2024 | Risorsa 6 | 10 |
| Cliente 3 | A-Apex Progetto | Alta 07/2024 | Simone Morassutto | 10 |

Figura 3.25. Allocazione Risorse.

La pagina in cui è possibile procedere con l’allocazione delle risorse e visualizzare i risultati è *Allocazione Risorse*. Data la necessità di garantire all’utente finale eventuali personalizzazioni e la possibilità di effettuare degli aggiustamenti manuali sui giorni allocati, si è reso necessario implementare un *Interactive Report* accoppiato ad un *form*. In questo contesto, il form è stato progettato per il solo aggiornamento delle righe del report, non essendo necessari l’inserimento e la cancellazione logica, già gestiti dalla procedura implementata.

La pagina *Allocazione Risorse* presenta alla destra della barra di ricerca dell’Interactive Report un bottone denominato *Alloca Risorse*, mediante il quale è possibile allocare le risorse disponibili alle attività pianificate. Tale bottone è stato creato attraverso l’aggiunta di un elemento di tipo *Button* nella sezione *Right of Interactive Report Search Bar* nel pannello di sinistra del Page Designer. Successivamente, è stata associata a questo bottone una *Dynamic Action* per gestire le funzionalità dinamiche. In particolare, è stato necessario dapprima configurare l’evento che attiva la *Dynamic Action*, ovvero il ‘click’ del bottone. In seguito, sono state definite tre azioni che vengono eseguite quando l’evento viene attivato. Nello specifico, queste sono:

- ***Un’azione di conferma.*** Questa prima azione assicura che l’utente confermi esplicitamente la sua intenzione di procedere con l’allocazione delle risorse;
- ***L’esecuzione della procedura PL/SQL lato server.*** La seconda azione consiste nell’esecuzione della procedura PL/SQL denominata *alloca_risorse()*, la quale è responsabile dell’effettiva assegnazione delle risorse alle attività pianificate. Questa operazione avviene solo dopo che l’utente ha fornito la sua conferma;
- ***Un’azione di invio della pagina.*** La terza azione si occupa dell’invio della pagina dopo che la procedura PL/SQL è stata eseguita con successo. Questo processo assicura che la pagina si aggiorni adeguatamente, riflettendo eventuali cambiamenti derivanti dall’allocazione delle risorse.

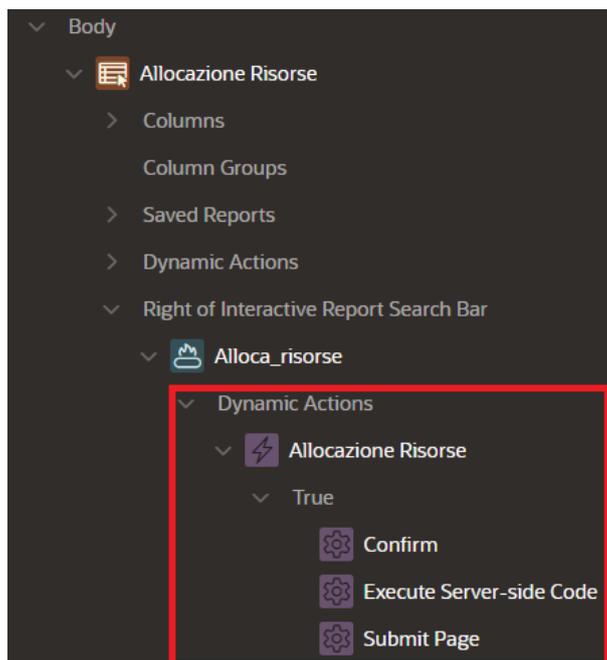


Figura 3.26. Dynamic Actions.

Secondo quanto descritto, dopo aver cliccato il bottone *Alloca Risorse* e dato conferma, l'utente visualizzerà nella pagina un report, nel quale, in ciascuna riga, verranno mostrate le informazioni relative ad una specifica attività (Cliente, Skill | Attività, Seniority | Mese/Anno) con l'aggiunta della risorsa che si occuperà interamente o in parte di svolgerla e dei giorni assegnati a quella risorsa per completarla. Di seguito si riporta la query implementata sulla quale è costruito l'*Interactive Report*.

Query Allocazione Risorse

```
select al.id_allocazione, cl.cli_des,
       t.tipologia_tecnologia || ' | ' || al.tipo_attivita
  AS Tecnologia_Activita,
       al.seniority || ' | ' || TO_CHAR(
       TO_DATE(al.mese_id, 'YYYYMM'), 'MM/YYYY')
  AS Seniority_Mese, u.nome_utente, al.giorni_effort
from t_allocazione al, t_a_tecnologie t,
     tabella_risorse u, tabella_clienti cl
where al.id_utente = u.id_utente
      AND al.cli_cod = cl.cli_cod
      AND al.id_tecnologia = t.id_tecnologia
      AND al.flag_cancelled = 'N'
```

Similmente alla tabella dei clienti, denominata *tabella_clienti*, anche *tabella_risorse* è una tabella aziendale preesistente, contenente le informazioni relative alle risorse dell'organizzazione.

La scelta di combinare le colonne "Skill | Attività" e "Seniority | Mese/Anno" è stata motivata dalla necessità di consentire la visualizzazione delle informazioni del report in due modalità distinte. Queste includono la presentazione sopra descritta fornita dal report interattivo, nonché un'altra modalità resa disponibile tramite una funzionalità interna del report stesso, ovvero la funzione pivot. Questa seconda visualizzazione, grazie alla sua capacità di eseguire un'aggregazione dei dati e presentarli in un formato più intuitivo, è stata progettata per ottimizzare l'analisi delle informazioni da parte dell'utente, offrendo la possibilità di effettuare analisi più efficaci.

In questo caso, a differenza delle pagine *Macropianificazione Overview*, *Macropianificazione Overview FTE* e *Skill Overview*, si è scelto di non implementare manualmente il pivot. Ciò è dovuto all'intento di rendere maggiormente flessibile l'implementazione della pagina *Allocazione Risorse*, che costituisce il nucleo centrale dell'applicazione MM Macropianificazione. L'uso della funzione pivot di un report interattivo consente, infatti, di sfruttare altre funzionalità dell'ambiente

APEX, come la gestione delle autorizzazioni, il controllo degli accessi e la personalizzazione dell'interfaccia utente, senza dover scrivere codice aggiuntivo e di rendere più semplici aggiornamenti futuri di questa pagina.

The screenshot shows the 'Allocazione Risorse' page in the MM Macropianificazione application. The interface includes a search bar, an 'Actions' menu with a 'Pivot' option selected, and a table with the following data:

| Cliente | Skill Attività | Seniority Mese/Anno | Risorsa 1 | Risorsa 2 | Risorsa 3 | Risorsa 4 | Risorsa 5 | Risorsa 6 | Simone Morassutto |
|-----------|-------------------|-----------------------|------------------|------------------|------------------|------------------|------------------|------------------|-------------------|
| | | | Giorni Assegnati | |
| Cliente 1 | DV-IBM AMS | Alta 05/2024 | 5 | 5 | 5 | | | | |
| Cliente 2 | E-ODI Analisi | Alta 06/2024 | | | | 20 | 20 | | |
| Cliente 3 | A-Apex Progetto | Alta 07/2024 | | | | | | 10 | 10 |

Figura 3.27. Pivot Allocazione Risorse.

Come già trattato, la funzione pivot consente di effettuare una trasposizione delle righe in colonne per generare risultati organizzati in un formato crosstab. [26] Per implementare un'operazione pivot in un Interactive Report è necessario cliccare sul bottone *Actions* (situato vicino alla barra di ricerca) e selezionare la voce *Pivot*. A quel punto si apre una finestra nella quale è possibile selezionare le colonne pivot da trasporre, le righe da visualizzare (fino ad un massimo di tre righe) e definire funzioni di aggregazione per le colonne trasposte. In figura sono mostrate le opzioni selezionate per il caso specifico.

Pivot ✕

Pivot Columns

1 Risorsa ▼

2 - Select Pivot Column - ▼

Add Pivot Column

Row Columns

1 Cliente ▼

2 Skill | Attività ▼

3 Seniority | Mese/Anno ▼

| | Functions | Column | Label | Format Mask | Sum |
|---|--|--|------------------|--|-------------------------------------|
| 1 | Sum ▼ | Giorni Assegnati ▼ | Giorni Assegnati | 999G999G999G999 ▼ | <input checked="" type="checkbox"/> |
| 2 | - Select Function - ▼ | - Select Column - ▼ | | ▼ | <input type="checkbox"/> |

Add Function

Cancel Delete **Apply**

Figura 3.28. Impostazioni Report Pivot.

Una volta creato il report pivot, accanto alla barra di ricerca compaiono due icone, *View Report* e *View Pivot*, con le quali si può rispettivamente passare dalla visualizzazione del report a quella del pivot.

Capitolo 4

Conclusioni e Sviluppi futuri

Nel presente elaborato di tesi, è stato trattato il mondo dello sviluppo low-code, soffermandosi in particolare su Oracle APEX come piattaforma di sviluppo. Sono state analizzate le caratteristiche chiave di tale tool, tra cui le sue capacità di sviluppo rapido di applicazioni web e la sua integrazione nativa con il database Oracle. Durante l'analisi di Oracle APEX, è stato affrontato come tale piattaforma fornisca strumenti potenti per l'implementazione di applicazioni di diversa complessità con ridotta codifica. Nello specifico, è stato esaminato come Oracle APEX consenta la creazione di pagine web dinamiche e interattive tramite l'utilizzo di componenti predefiniti e la personalizzazione mediante i linguaggi SQL e PL/SQL.

Successivamente, le conoscenze acquisite su tale piattaforma sono state utilizzate per sviluppare l'applicazione MM Macropianificazione, trattata nel Capitolo 3. Questa è stata progettata per gestire la pianificazione delle attività e l'allocazione delle risorse nell'azienda Mediamente Consulting, sede di tirocinio. Attraverso l'implementazione di pagine specifiche, è stato fornito all'utente finale uno strumento utile per la pianificazione delle attività aziendali, per la gestione delle risorse disponibili e per l'allocazione di tali risorse in base ad un algoritmo di allocazione definito.

L'algoritmo di allocazione è stato realizzato tenendo conto di diversi criteri, quali ad esempio le priorità delle risorse sui clienti e sulle tecnologie utilizzate, il livello di seniority delle risorse e vincoli temporali. Tale algoritmo è stato integrato nell'applicazione per automatizzare il processo di allocazione delle risorse, rendendo

maggiormente efficiente e precisa la pianificazione aziendale.

L'esperienza di sviluppo con Oracle APEX ha evidenziato le potenzialità che le piattaforme low-code offrono nella creazione di applicazioni web. Si tratta, infatti, di piattaforme che semplificano notevolmente il processo di sviluppo, anche per quanto riguarda le soluzioni più complesse, consentendo agli sviluppatori di realizzare rapidamente soluzioni robuste e funzionali, senza la necessità di scrivere un numero elevato di linee di codice.

L'implementazione dell'applicazione MM Macropianificazione ha fornito un esempio tangibile di come Oracle APEX possa essere utilizzato per affrontare problemi aziendali reali. L'applicazione è stata progettata per sostituire una serie di file Excel attualmente utilizzati dall'azienda riguardanti le informazioni relative alla pianificazione delle attività e alla gestione delle risorse, nonché per sostituire l'assegnazione manuale delle risorse alle attività. Uno dei principali casi d'uso di tale tool è, infatti, la sostituzione dei fogli di calcolo, in cui si è dimostrato come la possibilità di gestire i propri dati tramite la creazione di tabelle nel database Oracle, e successivamente la gestione di questi tramite un'applicazione, fornisca una struttura solida, che consente di centralizzare informazioni cruciali per il processo decisionale e ottimizzare operazioni rilevanti.

Tuttavia, è importante osservare che, mettendo a disposizione degli sviluppatori numerose funzionalità, l'apprendimento e la padronanza di Oracle APEX richiedono un investimento iniziale di tempo necessario a prendere confidenza con lo strumento e sfruttarne appieno le potenzialità. La curva di apprendimento è estremamente rapida, sebbene si ritengano necessarie conoscenze base di SQL, di database, dei suoi principali oggetti e, anche se in minima parte, dei linguaggi HTML, CSS e JavaScript. Inoltre, per affrontare algoritmi che richiedono una logica articolata, si considera essenziale possedere le principali nozioni di programmazione, come la logica dei cicli, le istruzioni IF-ELSE, ecc. Queste ultime sono state indispensabili per implementare la procedura PL/SQL che gestisce l'allocazione automatica delle risorse.

Si conclude, dunque, che l'approccio low-code supportato da Oracle APEX si è dimostrato un'opzione valida e potente, al fine di sviluppare il presente progetto di tesi. Infatti, la combinazione di un'interfaccia di sviluppo intuitiva, la flessibilità

delle funzionalità e la robustezza del database Oracle su cui si poggia sono in grado di rispondere a molteplici esigenze aziendali. Tuttavia, si ritiene che attualmente per sfruttare pienamente tale strumento sia fondamentale avere un certo grado di familiarità con i concetti di base della programmazione e del database, nonché monitorare l'evoluzione delle tecnologie low-code e le nuove release di APEX, al fine di garantire un utilizzo ottimale nel contesto aziendale.

4.1 Sviluppi futuri dell'applicazione

In quest'ultimo paragrafo, verranno delineate possibili integrazioni future per ampliare le funzionalità dell'applicazione MM Macropianificazione.

In primo luogo, potrebbe essere concessa la possibilità all'utente finale di caricare, visualizzare e gestire documenti o file relativi alle attività direttamente dall'applicazione. L'aggregazione di tutte le informazioni inerenti alle attività in un'unica piattaforma favorirebbe la coesione delle informazioni e la prontezza nell'accesso alle stesse.

Un'altra area da sviluppare in futuro potrebbe riguardare l'implementazione di un sistema di autorizzazioni avanzate. Attraverso Oracle APEX è, infatti, possibile definire ruoli e privilegi degli utenti, in modo tale che l'accesso a specifiche informazioni sia limitato alle sole persone autorizzate. Tale scenario estenderebbe l'utilizzo dell'applicazione non solo a chi si occupa della pianificazione, ma anche a coloro che svolgeranno tali attività, consentendo loro di consultare e, eventualmente, modificare le informazioni in base ai propri privilegi e alle unità operative di riferimento.

La pagina dedicata all'allocazione delle risorse potrebbe essere potenziata tramite l'impiego del componente Calendar di Oracle APEX. Tale approccio consentirebbe la visualizzazione e la gestione dei progetti direttamente all'interno di un calendario, offrendo agli utenti un'interfaccia intuitiva e facilmente comprensibile. Questo permetterebbe di avere una panoramica immediata delle attività assegnate alle risorse nel corso dei mesi, con la possibilità di specificare non solo i giorni previsti per il loro svolgimento, ma anche le settimane di riferimento. Inoltre, per migliorare ulteriormente la produttività delle risorse aziendali, l'applicazione potrebbe essere

integrata con noti servizi di calendario, come Outlook Calendar o Google Calendar. Questo consentirebbe alle risorse dell'organizzazione di visualizzare direttamente all'interno dei propri calendari le attività loro assegnate e di evidenziare le relative scadenze.

Infine, l'applicazione potrebbe gestire l'invio di notifiche via e-mail riguardanti le allocazioni, definite dall'algoritmo, alle risorse coinvolte. A tal fine, si potrebbe offrire al responsabile delle attività la possibilità di confermare le allocazioni effettuate, ad esempio tramite l'utilizzo di un apposito bottone, in modo tale da procedere con l'invio della notifica alla risorsa interessata.

Bibliografia

- [1] P. Staniszewski, “2023 gartner magic quadrant for enterprise lcacp: Low-code keeps getting more relevant,” 2023. <https://pretius.com/blog/gartner-quadrant-low-code/>.
- [2] ORACLE, “Application express architecture administration,” <https://view.officeapps.live.com/op/view.aspx?src=https%3A%2F%2Fwww.oracle.com%2Ftechnetwork%2Fdeveloper-tools%2Fapex%2Foverview%2Fapex-dba-1867077.pptx&wdOrigin=BROWSELINK>.
- [3] ORACLE, “Database 2 day + application express developer’s guide,” https://docs.oracle.com/cd/E14373_01/appdev.32/e13367/intro_app.htm#CEGFCAGD.
- [4] F. L. Trofa, “Sviluppo low code: cos’è e quali le migliori piattaforme,” 2022. <https://universeit.blog/sviluppo-low-code/>.
- [5] IBM, “Che cos’è il low-code,” 2021. <https://www.ibm.com/it-it/topics/low-code#:~:text=Le%20piattaforme%20low-code%20rispondono%20a%20questa%20esigenza%2C%20in,raggiungendo%20i%2013%2C8%20miliardi%20di%20dollari%20nel%202021>.
- [6] IBM, “Low-code vs. no-code: What’s the difference?,” <https://www.ibm.com/blog/low-code-vs-no-code/>.
- [7] T. Halaye, “Mercato della piattaforma di sviluppo low-code per applicazione (basata sul web, basata su dispositivi mobili, basata su desktop e server), uso finale (bfsi, settore automobilistico e manifatturiero, vendita al dettaglio, istruzione, it e telecomunicazioni, trasporti e logistica, altri) e regione, tendenze globali e previsioni dal 2023 al 2030,” 2023. <https://exactitudeconsultancy.com/it/reports/33949/>

- low-code-development-platform-market/#description.
- [8] Wikipedia, “Gartner,” <https://it.wikipedia.org/wiki/Gartner>.
- [9] S. U. Foppa, “Il futuro del low code: verso strategie di business “componibili,”” 2023. <https://www.zerounoweb.it/cio-innovation/metodologie/il-futuro-del-low-code-verso-strategie-di-business-componibili/>.
- [10] AppMaster, “Come il low-code sta sconvolgendo lo sviluppo tradizionale del software,” 2023. <https://appmaster.io/it/blog/rivoluzione-a-basso-codice-sviluppo-software#sfide-e-limiti-del-low-code>.
- [11] S. Menghini, “Oracle apex: lo sviluppo applicativo low-code visto dagli utenti,” 2021. <https://www.oracle.com/it/news/announcement/oracle-apex-low-code-application-development-2021-05-04/>.
- [12] C. Costa, “Oracle apex: il low code per applicazioni enterprise scalabili e sicure,” 2021. <https://www.bigdata4innovation.it/big-data/oracle-apex-il-low-code-per-applicazioni-enterprise-scalabili-e-sicure/>.
- [13] S. Lingua, “Oracle apex: piattaforma low-code come nuovo standard,” 2021. <https://blog.mediamenteconsulting.it/2021/03/18/mm039/>.
- [14] A. Mohindroo, “Oracle apex wins infoworld technology of the year 2023 award for software development platform,” 2023. <https://blogs.oracle.com/apex/post/oracle-apex-wins-infoworld-technology-of-the-year-2023-award>.
- [15] IBM, “Cos’è l’architettura three-tier,” <https://www.ibm.com/it-it/topics/three-tier-architecture>.
- [16] ORACLE, “Architettura,” <https://apex.oracle.com/it/platform/architecture/>.
- [17] ORACLE, “Oracle apex – app builder user’s guide,” 2024. <https://docs.oracle.com/en/database/oracle/apex/23.2/htmldb/oracle-apex-app-builder-users-guide.pdf>.
- [18] ORACLE, “Sql workshop guide,” 2024. <https://docs.oracle.com/en/database/oracle/apex/23.2/aeut1/index.html#Oracle%C2%AE-APEX>.
- [19] ORACLE, “Welcome to team development,” <https://apex.oracle.com/pls/apex/r/apex/team-development/welcome-page?clear=10&session=>

- 117270332756963.
- [20] ORACLE, “Why oracle apex?,” <https://apex.oracle.com/en/platform/why-oracle-apex/>.
 - [21] ORACLE, “Powered by oracle,” <https://apex.oracle.com/it/platform/powered-by-oracle/>.
 - [22] ORACLE, “Casi d’uso,” <https://apex.oracle.com/it/solutions/use-cases/>.
 - [23] ORACLE, “Roadmap,” <https://apex.oracle.com/en/learn/resources/roadmap/>.
 - [24] M. Consulting, “Media-mente consulting,” <https://www.mediamenteconsulting.it/>.
 - [25] ORACLE, “Database pl/sql language reference,” 2023. <https://docs.oracle.com/en/database/oracle/oracle-database/21/lnpls/plsql-triggers.html#GUID-217E8B13-29EF-45F3-8D0F-2384F9F1D231>.
 - [26] ORACLE, “End user’s guide,” 2022. <https://docs.oracle.com/en/database/oracle/apex/22.1/aeug/about-interactive-reports.html#GUID-9877F66F-A980-4604-91E8-0CD7B9A3ADDE>.
 - [27] M. Learn, “Vincoli di chiavi primarie ed esterne,” <https://learn.microsoft.com/it-it/sql/relational-databases/tables/primary-and-foreign-key-constraints?view=sql-server-ver16>.
 - [28] Wikipedia, “Equivalentente a tempo pieno,” https://it.wikipedia.org/wiki/Equivalentente_a_tempo_pieno.