



Università degli Studi di Salerno



Dipartimento di Ingegneria dell'Informazione ed Elettrica e
Matematica Applicata

Corso di Laurea in Ingegneria Informatica

Basi di Dati 2022/2023

Canale A-H

Project Work

**Traccia N.1 - Digital Marketing: Key Performance
Indicator**

Gruppo n. **08 – AH**

WP	Cognome e Nome	Matricola	e-mail	Responsabil e
1	Bruno Salvatore	0612705112	s.bruno35@studenti.unisa.it	X
2	Della Corte Mario	0612705151	m.dellacorte19@studenti.unisa.it	
3	Esposito Maurizio	0612705100	m.esposito343@studenti.unisa.it	
4	Ciaravola Giosuè	0612705043	g.ciaravola3@studenti.unisa.it	

Anno accademico 2022-2023

Sommario

1. Descrizione della realtà di interesse	4
1.1. Analisi della realtà di interesse	4
2. Analisi delle specifiche	7
2.1. Glossario dei termini	7
2.2. Strutturazione dei requisiti in frasi	9
2.2.1. Frasi di carattere generale	9
2.2.2. Frasi relative a Campagna	9
2.2.3. Frasi relative a Azienda	9
2.2.4. Frasi relative a Newsletter	9
2.2.5. Frasi relative a Utente	9
2.2.6. Frasi relative a Collezione	10
2.2.7. Frasi relative a Prodotto	10
2.2.8. Frasi relative a KPI	10
2.3. Identificazione delle operazioni principali	11
3. Progettazione Concettuale	12
3.1. Schema Concettuale	12
3.1.1. Note sullo schema E-R	13
3.2. Design Pattern	13
3.2.1. Pattern Storizzazione di un'entità per Campagna Pubblicitaria	13
3.2.2. Pattern Reificazione di relazione binaria per Valutazione	14
3.2.3. Pattern Reificazione di attributo per Categoria (di PRODOTTO)	15
3.2.4. Pattern Relazione di tipo Parte-di per Prodotto-Collezione	15
3.3. Dizionario dei Dati	16
3.4. Regole Aziendali	18
3.5. Analisi di qualità dello schema concettuale	19
4. Progettazione Logica	20
4.1. Ristrutturazione Schema Concettuale	20
4.1.1. Analisi delle Prestazioni	20
4.2. Analisi delle ridondanze	21
4.2.1. Analisi della ridondanza 1: Numero iscritti	21
4.3. Eliminazione delle generalizzazioni	26
4.3.1. Generalizzazione Campagna Pubblicitaria	26
4.4. Partizionamento/Accorpamento Entità e Associazioni	26
4.4.1. Eliminazione attributo multivalore Recapito Telefonico (di AZIENDA)	26
4.4.2. Eliminazione attributo multivalore Colori (di PRODOTTO)	27
4.4.3. Eliminazione attributo multivalore Taglie (di PRODOTTO)	27
4.4.4. Eliminazione attributo composto Sede Amministrativa (di AZIENDA)	27
4.5. Scelta degli identificatori principali	28
4.6. Schema ristrutturato finale	29
4.7. Schema logico	30
4.8. Documentazione dello schema logico	31
4.8.1. Vincoli della ristrutturazione	32

4.8.2. Vincoli della traduzione	32
5. Normalizzazione	33
5.1. CAMPAGNA_PUBBLICITARIA	33
5.1.1. Dipendenze funzionali	33
5.1.2. Forma normale raggiunta	33
5.2. AZIENDA	33
5.2.1. Dipendenze funzionali	33
5.2.2. Forma normale raggiunta	33
5.3. NUMERO_TELEFONICO	33
5.3.1. Dipendenze funzionali	33
5.3.2. Forma normale raggiunta	33
5.4. CATEGORIA	34
5.4.1. Dipendenze funzionali	34
5.4.2. Forma normale raggiunta	34
5.5. KPI	34
5.5.1. Dipendenze funzionali	34
5.5.2. Forma normale raggiunta	34
5.6. UTENTE	34
5.6.1. Dipendenze funzionali	34
5.6.2. Forma normale raggiunta	34
5.7. NEWSLETTER	35
5.7.1. Dipendenze funzionali	35
5.7.2. Forma normale raggiunta	35
5.8. VALUTAZIONE	35
5.8.1. Dipendenze funzionali	35
5.8.2. Forma normale raggiunta	35
5.9. COLLEZIONE	35
5.9.1. Dipendenze funzionali	35
5.9.2. Forma normale raggiunta	35
5.10. PRODOTTO	36
5.10.1. Dipendenze funzionali	36
5.10.2. Forma normale raggiunta	36
5.11. COLORI_ASSORTITI	36
5.11.1. Dipendenze funzionali	36
5.11.2. Forma normale raggiunta	36
5.12. COLORE	36
5.12.1. Dipendenze funzionali	36
5.12.2. Forma normale raggiunta	36
5.13. TAGLIE_ASSORTITE	36
5.13.1. Dipendenze funzionali	36
5.13.2. Forma normale raggiunta	36
5.14. TAGLIA	37
5.14.1. Dipendenze funzionali	37
5.14.2. Forma normale raggiunta	37

5.15. ISCRIZIONE	37
5.15.1. Dipendenze funzionali	37
5.15.2. Forma normale raggiunta	37
5.16. PUBBLICIZZA	37
5.16.1. Dipendenze funzionali	37
5.16.2. Forma normale raggiunta	37
6. Script Creazione e Popolamento Database	37
6.1. Generated Always As Identity	38
6.2. Note sul popolamento	38
6.3. Script di creazione e popolamento	39
7. Query SQL	40
7.1. Query con operatore di aggregazione e join: Tasso di conversione medio per ogni campagna pubblicitaria	40
7.2. Query nidificata complessa: Campagne in corso che pubblicizzano almeno un prodotto disponibile di categoria “maglie”	41
7.3. Query insiemistica: Collezioni pubblicizzate da Zalando ma non fornite da Etam	42
7.4. Eventuali Altre query	43
7.4.1. Profilazione	43
8. Viste	44
8.1. Vista: VistaAziendaNewsletterMediaValutazioni	44
8.1.1. Query con Vista: MigliorFeedbackCampagna	45
9. Trigger	46
9.1. Trigger Inizializzazione	46
9.1.1. Trigger 1: AlmenoUnPubblicizza()	46
9.1.2. Trigger2: aggiornaRidondanza()	48
9.1.3. Trigger3: proteggiRidondanza()	49
9.2. Trigger per vincoli aziendali	50
9.2.1. Trigger1: vietatscrizioniDisiscrizioni()	50
9.2.2. Trigger2: controlloCampagna()	51

1. Descrizione della realtà di interesse

Titolo: **A-H - Digital Marketing: Key Performance Indicator**

L'azienda Zalando richiede la progettazione di una base di dati per supportare le sue campagne di advertising online basate su newsletter. Una newsletter è una forma di comunicazione periodica inviata via email agli utenti che si sono registrati per riceverla. Solitamente, una newsletter contiene informazioni, notizie, aggiornamenti o promozioni su uno o più prodotti da parte di una specifica azienda, organizzazione o individuo. Sarà dunque necessario memorizzare informazioni sugli utenti, sui prodotti e sui vari KPI relativi all'andamento delle campagne stesse. In particolare:

- La **newsletter** sarà utilizzata per mantenere i clienti aggiornati sulle novità dei prodotti e sulla loro disponibilità effettiva.
- Ogni newsletter è associata ad una **campagna di advertising** (circa 20 attive mensilmente, ma in costante aumento, contando quelle terminate, circa 50 per stagione), identificata da un *codice* (numero progressivo), e per la quale è utile memorizzare *nome*, *budget*, *data di inizio* e *data di fine* (solo per le campagne concluse).
- La singola **newsletter** è identificata dalla campagna a cui è associata. Inoltre è utile memorizzare il *numero di iscritti* (inizializzato a zero) e la *cadenza* (espressa come numero di giorni che intercorrono tra due invii).
- Per ogni newsletter, c'è una lista di **utenti** iscritti (circa 100.000) identificati da *nickname* o dall'*e-mail*, di cui è utile memorizzare il *nome*, il *cognome*, il *sex*, la *data di nascita* e la *data di iscrizione* ad ogni newsletter.
- Lo scopo della campagna è pubblicizzare una o più **collezioni** (circa 3 per azienda, quindi 150). In particolare questa è identificata dal *nome*, dall'*anno* e dall'*azienda* che la fornisce; Risulta inoltre utile memorizzare una *descrizione* che ne illustri le caratteristiche.
- Una collezione è composta da uno o più **prodotti** (circa 10 per collezione, quindi 1500), identificati da *nome* (modello) e *collezione* di appartenenza. Inoltre, del prodotto, è utile memorizzare *colori* e *taglie assortiti*, la *categoria*, la *sezione* di appartenenza, la *disponibilità* (del prodotto in generale) e il *voto medio*.
- Ogni collezione pubblicizzata da una campagna è fornita da un'**azienda** che collabora con Zalando (circa 50), le quali comunicheranno l'effettiva *disponibilità* dei prodotti in vendita. L'azienda è identificata dalla *partita IVA* e di essa è utile memorizzare il *nome*, la *sede amministrativa* (opzionale, formata da nazione, città), e uno o più *recapiti telefonici*.
- Per ogni newsletter sono memorizzati diversi **KPI** (circa 9), ossia indicatori di performance tecnica per valutare il suo andamento. I KPI sono identificati dal *nome* e di essi è utile memorizzare la *descrizione*, la *formula*, il *valore minimo* (opzionale) e il *valore massimo* (opzionale); per ogni newsletter i KPI assumono un valore che viene inserito in un certo istante di tempo, in quanto i valori dei KPI vengono continuamente aggiornati fin quando la campagna non termina.

1.1. Analisi della realtà di interesse

Il progetto consiste nella creazione di un database per memorizzare le informazioni relative alle campagne di advertising di Zalando. Le diverse **campagne** potranno essere indette da Zalando stesso, al fine di mantenere gli utenti aggiornati sulla disponibilità dei prodotti, o commissionate da un'azienda esterna in collaborazione con Zalando, in modo da promuovere i propri prodotti. Per

poter confrontare campagne passate e correnti risulta necessario mantenere uno storico delle campagne pubblicitarie (quindi delle newsletter e KPI associati).

Quando viene indetta una campagna, viene creata la newsletter ad essa associata ed i KPI inizialmente non contengono valori, essi infatti saranno calcolati a partire dalla prima cadenza della newsletter, inoltre vengono memorizzate le collezioni da sponsorizzare, le aziende che le forniscono e i prodotti da cui sono composte; per questo motivo, un'azienda può essere inserita nel momento in cui indice una campagna (e quindi fornisce i suoi prodotti) oppure quando fornisce uno o più prodotti per una campagna di Zalando.

Inoltre, se l'azienda che intende commissionare una campagna è Zalando allora potrà pubblicizzare solo collezioni con lo stesso nome e lo stesso anno (esempio: tutte le collezioni primavera 2023 fornite da varie aziende); tutte le altre aziende, invece, potranno pubblicizzare una sola collezione che loro stesse forniscono.

Le **aziende** forniscono le loro informazioni anagrafiche, oltre a comunicare l'effettiva disponibilità dei loro prodotti, per garantire una corretta gestione delle collaborazioni. Anche Zalando è registrata nella base di dati, pubblicizza ma non fornisce collezioni.

Ogni **collezione** pubblicizzata sarà fornita da un'unica azienda e conterrà le informazioni su quest'ultima, un nome ed un anno relativo alla tematica o un periodo di prodotti che intende pubblicizzare (sport 2023, elegante 2023, inverno 2022, primavera 2023, ecc..) accompagnati da una opportuna descrizione. È possibile che una collezione possa essere pubblicizzata da più campagne.

Ogni **prodotto** che compone quella stessa collezione, sarà caratterizzato dalle informazioni ad esso relative (ad esempio nome, colori e taglie assortiti, categoria, sezione, la propria disponibilità e un voto medio in termini di recensione), utili per le analisi di mercato. Per categoria si intende la tipologia di abbigliamento (calzature, giacche, maglieria, pantaloni, ecc..) ed ogni prodotto appartiene ad una categoria. Per sezione si intende il target a cui è rivolto (Uomo, Donna, Bambino). Il voto medio del prodotto è espresso come numero con una cifra dopo la virgola che va da 0 a 5. Con disponibilità non si intende il numero di pezzi disponibili, ma solo se quel prodotto è disponibili o meno.

La **newsletter** sarà dunque lo strumento principale per tenere aggiornati gli utenti iscritti, seguendo la cadenza prefissata;

Gli **utenti** iscritti ad una newsletter (fornendo la propria anagrafica, utile a fini di profilazione), una volta conclusa l'iscrizione saranno correttamente registrati e riceveranno periodici aggiornamenti all'indirizzo e-mail fornito. Un utente può iscriversi e disiscriversi a qualunque newsletter a patto che la campagna associata non sia conclusa.

I **KPI** sono espressi come numeri con due cifre dopo la virgola e possono avere (non necessariamente) un valore minimo e un valore massimo. Per esempio, nel caso dei KPI normalizzati, il valore minimo è 0 e il valore massimo è 1. Inoltre, per tener traccia di come i KPI variano durante il corso della singola campagna, i loro valori non vengono sovrascritti ma ad ogni aggiornamento viene inserito un nuovo valore con l'istante di inserimento associato.

Per ogni newsletter sono memorizzati diversi KPI, ossia indicatori di performance tecnica atti a valutare l'andamento della campagna. I KPI che considereremo sono i seguenti.

- **Tasso di apertura:** Percentuale di utenti che aprono la newsletter rispetto al numero totale di destinatari. Formula normalizzata: $\#Aperture / (\#Email\ inviate - \#Email\ non\ ricevute)$;
- **Tasso di clic:** Percentuale di utenti che cliccano sui link presenti nella newsletter. Formula normalizzata: $\#Clicks / (\#Email\ inviate - \#Email\ non\ ricevute)$;

- **Tasso di conversione:** Percentuale di utenti che completano l'acquisto di un prodotto tramite la newsletter rispetto al numero totale di clic effettuati. Formula normalizzata: $\#Conversioni / \#Clicks$;
- **Tasso di disiscrizione:** Percentuale di utenti che si disiscrivono dalla newsletter rispetto al numero totale di destinatari. Formula normalizzata: $\#Disiscrizioni / (\#Email\ inviate - \#Email\ non\ ricevute)$;
- **Tasso di forward e condivisione:** Percentuale di utenti che inoltrano la newsletter ad altre persone o la condividono sui social network rispetto al numero totale di destinatari. Formula Normalizzata: $\#Condivisioni / (\#Email\ inviate - \#Email\ non\ ricevute)$;
- **Feedback:** Giudizio degli utenti espresso tramite valutazioni, normalizzato. Formula Normalizzata: $Voto\ Medio\ Newsletter / 5$;
- **Spaccato per device:** Percentuale di utenti che accedono alla newsletter tramite dispositivi mobili. Formula normalizzata: $\#Aperture\ Su\ Mobile / \#Aperture\ Totali$;
- **Tasso di risposta:** Percentuale di utenti che rispondono alla newsletter tramite feedback o contatti diretti con Zalando. Formula normalizzata: $\#Utenti\ Che\ Rispondono / \#Utenti\ Iscritti$;
- **Tasso di crescita:** Tasso di crescita della lista degli iscritti. Formula: $((\#Iscritti\ Attuali - \#Iscritti\ Precedenti) / \#Iscritti\ Precedenti)$. È importante specificare che questo viene calcolato usando come numero di iscritti precedenti quello risalente all'ultimo resoconto dei KPI (3 volte a settimana); è l'unico KPI non normalizzabile, quindi non ha valore minimo o valore massimo.

Dato che un utente deve aprire l'e-mail prima di cliccare un link in essa inclusa, si assume che il tasso di apertura è maggiore o uguale del tasso di clic; analogamente un utente deve cliccare un link incluso nell'e-mail prima di comprare il prodotto pubblicizzato (conversione), quindi si assume che il tasso di clic è maggiore o uguale del tasso di conversione.

In definitiva, il focus del progetto sarà sulla definizione di una lista di KPI tecnici (numero di clic, conversioni, dispositivi utilizzati, numero di condivisioni, crescita del reach della newsletter, ecc.). Non verranno, invece, presi in considerazione indici di carattere economico, come il fatturato generato dalle conversioni, i costi sostenuti, il ritorno sull'investimento (ROI), ecc.

2. Analisi delle specifiche

Workpackage	Task	Responsabile
WPO	Analisi delle specifiche	Intero Gruppo

2.1. Glossario dei termini

	Termine	Descrizione	Sinonimi	Collegamenti
1	Newsletter	Comunicazione periodica via email con contenuti informativi e promozionali inviati da Zalando ai suoi utenti iscritti.	Periodico via e-mail	Campagna di Advertising, Utente, KPI
2	Campagna di Advertising	Attività di promozione di prodotti o servizi attraverso la diffusione di messaggi pubblicitari su canali di comunicazione online	Campagna pubblicitaria, Campagna	Newsletter, Azienda, Collezione
3	KPI	Indicatori chiave di performance utilizzati per valutare l'efficacia di una campagna di newsletter e le sue performance tecniche	Indice di performance	Newsletter
4	Anagrafica azienda	Insieme di dati relativi all'identificazione di un'azienda	Identificativo azienda	Azienda
5	Utente	Persona iscritta alla newsletter di Zalando	Fruitore	Newsletter
6	Prodotto	Bene venduto da Zalando e pubblicizzato tramite la newsletter	Articolo, bene, merce, oggetto	Collezione
7	Collezione	Insieme di prodotti simili o correlati per tematica o periodo, che vengono messi in vendita o pubblicizzati contemporaneamente (sport, elegante, inverno 2022-23, primavera 2023, ecc..)	Linea di prodotti	Campagna di Advertising, Prodotto, Azienda
8	Categoria	Tipologia di abbigliamento (calzature, giacche, maglieria, pantaloni, ecc..)	Tipo, Tipologia	Prodotto
9	Sezione	Pubblico target del prodotto (uomo, donna, bambino)	-	Prodotto
10	Azienda	Società che collabora con Zalando fornendo prodotti per la vendita sul sito	Società, Impresa	Campagna di Advertising, Collezione
11	Tasso di Apertura	Percentuale di utenti che aprono la newsletter rispetto al numero totale di destinatari.	-	KPI

12	Tasso di Clic	Percentuale di utenti che cliccano sui link presenti nella newsletter rispetto al numero totale di destinatari.	-	KPI
13	Tasso di conversione	Percentuale di utenti che completano l'acquisto di un prodotto tramite la newsletter rispetto al numero totale di clic effettuati.	-	KPI
14	Tasso di Disiscrizione	Percentuale di utenti che si disiscrivono dalla newsletter rispetto al numero totale di destinatari.	-	KPI
15	Tasso di Forward e condivisione	Percentuale di utenti che inoltrano la newsletter ad altre persone o la condividono sui social network rispetto al numero totale di destinatari.	-	KPI
16	Feedback dei lettori	Giudizio degli utenti espresso tramite valutazioni, normalizzato.	-	KPI
17	Spaccato per Device	Percentuale di utenti che accedono alla newsletter tramite dispositivi mobili.	-	KPI
18	Tasso di risposta	Percentuale di utenti che rispondono alla newsletter tramite feedback o contatti diretti con Zalando.	-	KPI
19	Tasso di crescita	Tasso di crescita della lista degli iscritti	-	KPI

Tabella 1. Glossario dei Termini

2.2. Strutturazione dei requisiti in frasi

2.2.1. Frasi di carattere generale

Il progetto consiste nella creazione di un database per memorizzare le informazioni relative alle campagne di advertising basate su Newsletter di Zalando.

2.2.2. Frasi relative a Campagna

Ogni newsletter è associata ad una campagna di advertising (circa 20 attive mensilmente, ma in costante aumento contando quelle terminate), identificata da un codice (numero progressivo), e di cui è utile memorizzare nome, budget, data di inizio e data di fine (solo per le campagne concluse). Il progetto prevede la possibilità di indire campagne pubblicitarie da parte di Zalando o di aziende esterne che vi collaborano, con l'obiettivo di mantenere gli utenti aggiornati sulla disponibilità dei prodotti o di promuovere prodotti specifici.

Per tracciare il progresso delle campagne, viene mantenuto uno storico delle newsletter e dei KPI associati, con i valori dei KPI che vengono registrati ad ogni aggiornamento e non sovrascritti.

Quando viene indetta una campagna, vengono memorizzate le collezioni da sponsorizzare, le aziende che le forniscono e i prodotti da cui sono composte. Un'azienda può essere inserita sia al momento dell'indizione di una campagna (se fornisce i suoi prodotti) sia quando fornisce uno o più prodotti per una campagna di Zalando. Nel caso in cui l'azienda che commissiona una campagna sia Zalando, può pubblicizzare solo collezioni con lo stesso nome e anno, mentre tutte le altre aziende possono pubblicizzare una sola collezione da loro stesse fornita.

2.2.3. Frasi relative a Azienda

Ogni collezione pubblicizzata da una campagna è fornita da un'azienda che collabora con Zalando (circa 50), le quali comunicheranno l'effettiva disponibilità dei prodotti in vendita. L'azienda è identificata dalla partita IVA e di essa è utile memorizzare il nome, la sede amministrativa (opzionale, formata da nazione, città), e uno o più recapiti telefonici.

Le aziende forniscono le loro informazioni anagrafiche, oltre a comunicare l'effettiva disponibilità dei loro prodotti, per garantire una corretta gestione delle collaborazioni. Anche Zalando è registrata nella base di dati, ma non fornisce collezioni.

Un'azienda può essere inserita sia al momento dell'indizione di una campagna (se fornisce i suoi prodotti) sia quando fornisce uno o più prodotti per una campagna di Zalando. Nel caso in cui l'azienda che commissiona una campagna sia Zalando, può pubblicizzare solo collezioni con lo stesso nome e anno, mentre tutte le altre aziende possono pubblicizzare una sola collezione da loro stesse fornita.

2.2.4. Frasi relative a Newsletter

La newsletter sarà utilizzata per mantenere i clienti aggiornati sulle novità dei prodotti e sulla loro disponibilità effettiva.

La singola newsletter è identificata dalla campagna a cui è associata. Inoltre è utile memorizzare il numero di iscritti (inizializzato a zero) , la cadenza (espressa come numero di giorni che intercorrono tra due invii)

2.2.5. Frasi relative a Utente

Per ogni newsletter, è prevista una lista di utenti (circa 100.000) iscritti identificati da nickname o e-mail, con i dati anagrafici memorizzati, tra cui nome, cognome, sesso, data di nascita e data di iscrizione.

Gli utenti iscritti forniranno le proprie informazioni anagrafiche per fini di profilazione e, una volta conclusa l'iscrizione, saranno registrati correttamente e riceveranno aggiornamenti periodici all'indirizzo e-mail fornito.

Un utente può iscriversi e disiscriversi a qualunque newsletter a patto che la campagna associata non sia conclusa.

2.2.6. Frasi relative a Collezione

L'obiettivo delle campagne è quello di pubblicizzare una o più collezioni (circa 3 per azienda, per un totale di 150), ciascuna identificata dal nome, dall'anno e dall'azienda che la fornisce, oltre ad essere dotata di una descrizione.

Ogni collezione pubblicizzata è fornita da un'unica azienda e contiene informazioni su di essa, come il nome e l'anno relativi alla tematica o al periodo di prodotti che intende promuovere (ad esempio, sport 2022, elegante 2023, inverno 2022, primavera 2023, ecc.), accompagnati da una descrizione adeguata. È possibile che una collezione possa essere pubblicizzata da più di una campagna.

2.2.7. Frasi relative a Prodotto

Una collezione è composta da uno o più prodotti (circa 10 per collezione, quindi 1500), identificati da nome (modello) e collezione di appartenenza. Inoltre, del prodotto, è utile memorizzare colori e taglie assortiti, la categoria, la sezione di appartenenza, la disponibilità (del prodotto in generale) e il voto medio.

Per categoria si intende la tipologia di abbigliamento (calzature, giacche, maglieria, pantaloni, ecc..) ed ogni prodotto potrà appartenere a nessuna o ad una categoria.

Per sezione si intende il target a cui è rivolto (Uomo, Donna, Bambino).

Il voto medio del prodotto è espresso come numero con una cifra dopo la virgola che va da 0 a 5.

Con disponibilità non si intende il numero di pezzi disponibili, ma solo se quel prodotto è disponibili o meno.

2.2.8. Frasi relative a KPI

Per ogni newsletter sono memorizzati diversi KPI, ossia indicatori di performance tecnica per valutare il suo andamento. I KPI sono identificati dal nome e di essi è utile memorizzare la descrizione, la formula, il valore minimo (opzionale) e il valore massimo (opzionale).

I KPI che considereremo sono: il tasso di apertura (quante persone hanno aperto l'email), il tasso di clic (quante persone hanno cliccato su un link nell'email), il tasso di conversione (quante persone hanno effettivamente acquistato il prodotto), il tasso di disiscrizione (quante persone si sono disiscritte dalla newsletter), il tasso di forward e condivisione (quante persone hanno inoltrato o condiviso l'email), il feedback dei lettori, lo spaccato per device (quanti utenti hanno aperto l'email da computer, smartphone, ecc.), il tasso di risposta (quante persone hanno risposto alla newsletter) e il tasso di crescita della lista iscritti (quante nuove persone si sono iscritte alla newsletter).

I KPI sono espressi come numeri con due cifre dopo la virgola e possono avere (non necessariamente) un valore minimo e un valore massimo. Per esempio, nel caso dei KPI normalizzati, il valore minimo è 0 e il valore massimo è 1.

Inoltre, per tener traccia di come i KPI variano durante il corso della singola campagna, i loro valori non vengono sovrascritti ma ad ogni aggiornamento viene inserito un nuovo valore con l'istante di inserimento associato.

È importante notare che il tasso di apertura è maggiore o uguale al tasso di clic, poiché gli utenti devono aprire l'email prima di cliccare su un link, e che il tasso di clic è maggiore o uguale al tasso

di conversione, poiché gli utenti devono cliccare su un link per acquistare un prodotto pubblicizzato.

2.3. Identificazione delle operazioni principali

Operazione 1: inserisci i dati relativi alla nuova campagna, alla collezione pubblicizzata e prodotti da cui è composta se non già presenti (in media 1 volta alla settimana)

Operazione 2: inserisci i dati relativi a una nuova newsletter associata a una campagna (in media 1 volta a settimana)

Operazione 3: inserisci una nuova iscrizione di un utente a una newsletter (in media 200 volte al giorno)

Operazione 4: inserisci l'anagrafica di una nuova azienda che collabora con Zalando (in media 2 volte al mese)

Operazione 5: aggiorna i valori dei KPI di ogni newsletter (in media 2 volte al giorno)

Operazione 6: stampa un resoconto dei KPI di una newsletter, includendo le informazioni dei prodotti pubblicizzati, della newsletter (incluso il numero di iscritti) e della campagna di appartenenza (in media 3 volte a settimana)

Operazione 7: Identifica le newsletter che hanno maggiore successo in base a tasso di conversione, tasso di clic, tasso di apertura, e che hanno maggiore copertura in base a soglia di condivisione e forward, e che presentano un buon tasso di risposta e valutazione positiva (in media due volte alla settimana).

Operazione 8: stampa un report dello storico delle campagne terminate o ancora in corso nella stagione appena conclusa (informazioni sulla campagna, newsletter con numero di iscritti finale, e resoconto KPI) (in media 4 volte all'anno).

3. Progettazione Concettuale

Workpackage	Task	Responsabile
WP1	Progettazione Concettuale	Bruno Salvatore

3.1. Schema Concettuale

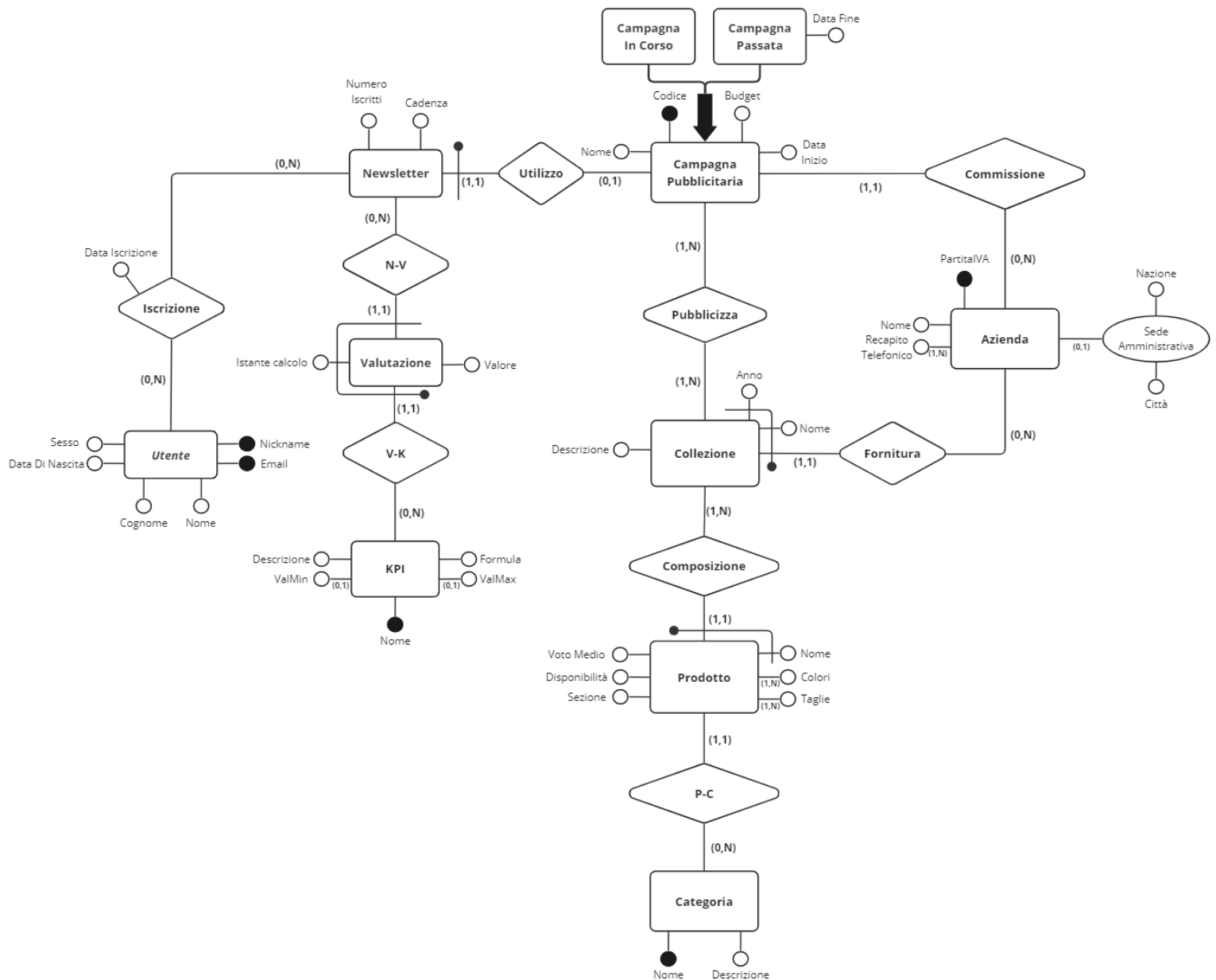


Figura 1. Schema E-R

3.1.1. Note sullo schema E-R

Lo schema concettuale è stato costruito utilizzando una strategia **top-down**, ossia partendo da uno scheletro composto dalle entità principali che è stato raffinato mediante opportune trasformazioni che aumentano il dettaglio dei vari concetti presenti; per esempio aggiungendo prima le associazioni, poi le cardinalità, e infine gli attributi.

3.2. Design Pattern

3.2.1. Pattern Storicizzazione di un'entità per Campagna Pubblicitaria

Nelle specifiche è espresso il bisogno di mantenere nella base di dati tutte le informazioni sulle campagne pubblicitarie (inclusando quelle sulle newsletter e KPI associati) anche in seguito alla loro terminazione. Per ottenere il risultato richiesto l'entità CAMPAGNA PUBBLICITARIA è stata specializzata in due entità figlie, ossia CAMPAGNA IN CORSO e CAMPAGNA PASSATA; quest'ultima aggiunge l'attributo *Data Fine*.

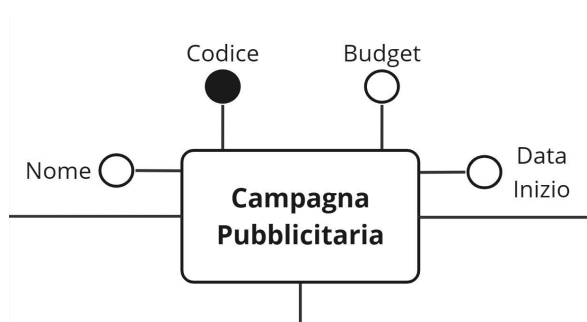


Figura 2. Schema precedente all'applicazione del Pattern Storicizzazione di un'entità per Campagna Pubblicitaria.

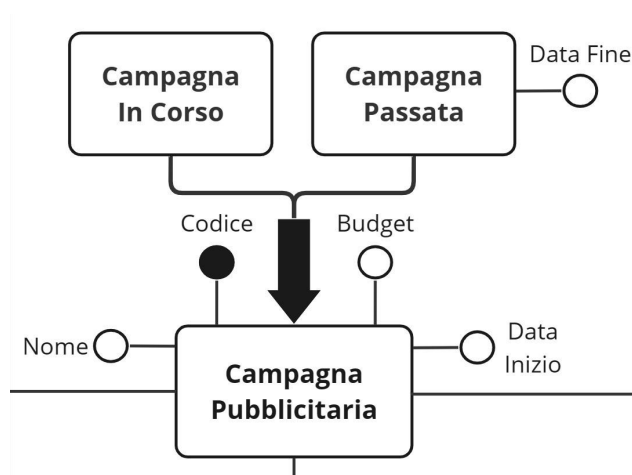


Figura 3. Schema successivo all'applicazione del Pattern Storicizzazione di un'entità per Campagna Pubblicitaria.

L'applicazione di questo pattern implica l'aggiunta di vincoli riguardanti l'impossibilità di aggiornare i dati di campagna, newsletter e KPI associati una volta che la campagna diventa terminata.

3.2.2. Pattern Reificazione di relazione binaria per Valutazione

Nelle specifiche si evidenzia che l'aggiornamento dei KPI non avviene sovrascrivendo il valore precedente ma aggiungendo un nuovo valore corredato di istante di inserimento. La soluzione precedente all'utilizzo del pattern non permette, per definizione, di rappresentare più occorrenze dell'associazione VALUTAZIONE che lega la stessa coppia di occorrenze di NEWSLETTER e KPI. Il problema non può essere risolto tramite l'utilizzo dell'attributo *Istante Calcolo* perché le associazioni non possono avere attributi identificativi.

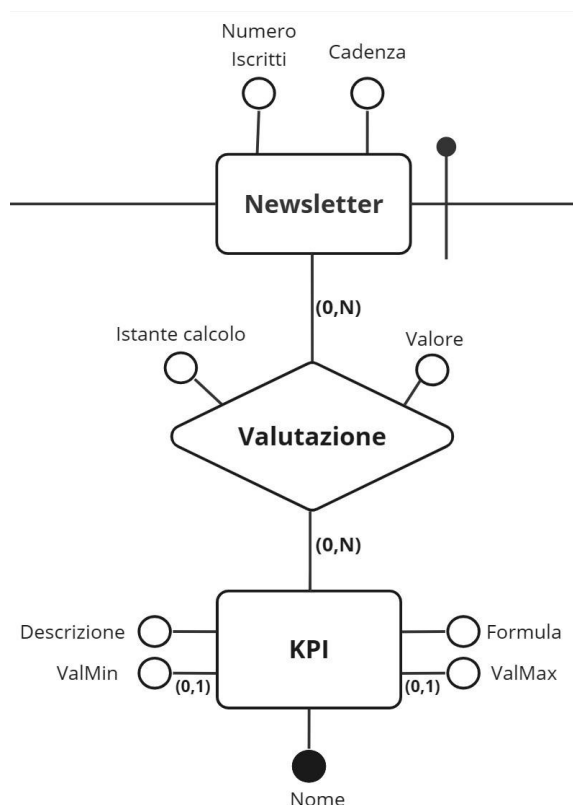


Figura 4. Schema precedente all'applicazione del Pattern Reificazione di relazione binaria per Valutazione.

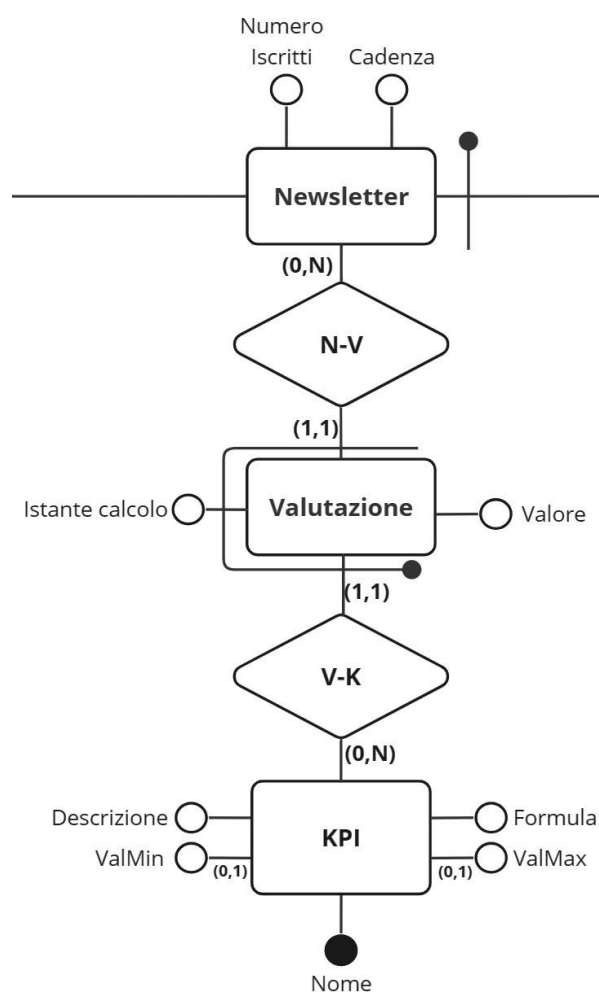


Figura 5. Schema successivo all'applicazione del Pattern Reificazione di relazione binaria per Valutazione.

Con la soluzione successiva all'utilizzo del pattern l'associazione VALUTAZIONE viene reificata. Per fare in modo che sia possibile rappresentare più occorrenze di Valutazione per la stessa coppia di occorrenze NEWSLETTER-KPI, l'identificatore dell'entità VALUTAZIONE deve includere, oltre a KPI e Newsletter (esterni), anche l'attributo *Istante Calcolo*, ottenendo un identificatore misto.

3.2.3. Pattern Reificazione di attributo per Categoria (di PRODOTTO)

Da una successiva analisi è emerso che l'attributo *Categoria* assume un set preciso di valori (che possono evolversi nel tempo), composti da *Nome* e *Descrizione*. Per questo motivo è risultato necessario reificare l'attributo *Categoria* in entità composta dagli attributi *Nome* e *Descrizione* e associarla a PRODOTTO tramite la relazione P-C.

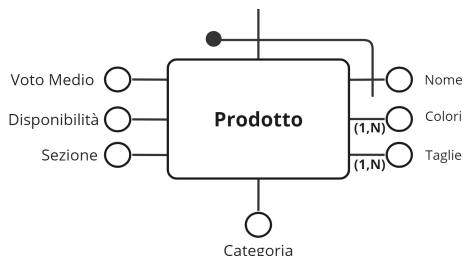


Figura 6. Schema precedente all'applicazione del Pattern Reificazione di attributo per Categoria (di Prodotto).

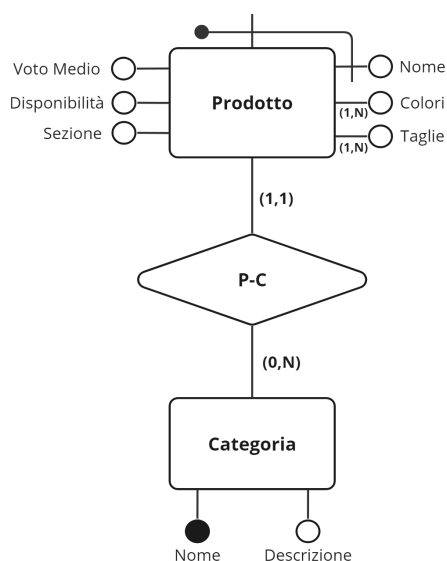


Figura 7. Schema successivo all'applicazione del Pattern Reificazione di attributo per Categoria (di Prodotto).

3.2.4. Pattern Relazione di tipo Parte-di per Prodotto-Collezione

Nelle specifiche si evidenzia che la collezione è composta da un insieme di prodotti, quindi è evidente la presenza del pattern **Parte-di**.

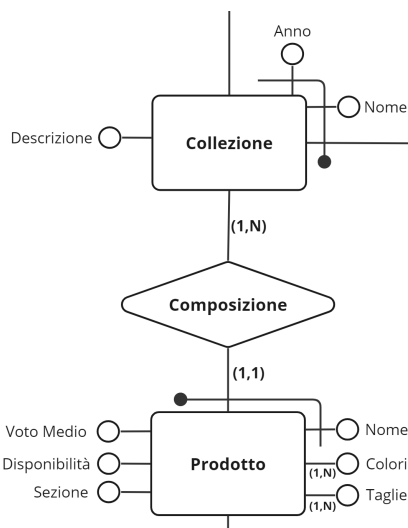


Figura 8. Schema successivo all'applicazione del Pattern Relazione di tipo Parte-di per Prodotto-Collezione.

3.3. Dizionario dei Dati

Entità	Descrizione	Attributi	Identificatore
Utente	Persona iscritta alla piattaforma di Zalando	Nome, Cognome, Nickname, Email, Sesso, Data Di Nascita	Nickname, Email
Newsletter	Comunicazione periodica via email con contenuti promozionali inviati da Zalando agli utenti iscritti	Cadenza, Numero Iscritti	CAMPAGNA PUBBLICITARIA
Valutazione	Valore (corredato di istante di inserimento) assunto da un KPI per una specifica Newsletter	Valore, Istante Calcolo	NEWSLETTER + Istante Calcolo + KPI
KPI	Identificatori di performance tecnica per la valutazione dell'efficacia della newsletter	Nome, Descrizione, Formula, Valore Min (0,1) , Valore Max (0,1)	Nome
Campagna Pubblicitaria	Attività di promozione dei prodotti attraverso la diffusione di messaggi pubblicitari su canali di comunicazione online	Codice, Nome, Budget, Data di Inizio	Codice
Campagna in Corso	Campagna pubblicitaria non ancora terminata	Codice, Nome, Budget, Data di Inizio	Codice
Campagna Passata	Campagna pubblicitaria già terminata con un'opportuna data di fine	Codice, Nome, Budget, Data di Inizio, Data di Fine	Codice
Azienda	Società che collabora con Zalando fornendo prodotti per la vendita sul sito	PartitaIVA, Nome, Recapito Telefonico (1,N) , Sede Amministrativa (Nazione , Città) (0,1)	PartitaIVA
Collezione	Insieme di prodotti simili o correlati per tematica o periodo, che vengono messi in vendita o pubblicizzati contemporaneamente (sport, elegante, inverno 2022, primavera 2023, ecc..)	Anno ,Nome, Descrizione	Anno + Nome + AZIENDA

Prodotto	Bene venduto da Zalando e pubblicizzato tramite la newsletter	Nome, Voto Medio, Colori, Taglie, Disponibilità, Sezione	Nome + COLLEZIONE
Categoria	Tipologia di abbigliamento (calzature, giacche, maglieria, pantaloni, ecc..)	Nome, Descrizione	Nome

Tabella 2. Dizionario dei dati – Entità

Relazioni	Descrizione	Entità Coinvolte	Attributi
Iscrizione	Associa l'utente alla newsletter a cui si è iscritto (se si disiscrive viene rimossa)	Utente (0,N), Newsletter (0,N)	Data Iscrizione
N-V	Associa la Newsletter alla propria valutazione	Newsletter (0,N), Valutazione (1,1)	-
V-K	Associa la Valutazione al KPI a cui si riferisce	Valutazione (1,1) , KPI (0,N)	-
Utilizzo	Relazione che lega la newsletter con la campagna pubblicitaria che utilizza	Newsletter (1,1), Campagna Pubblicitaria (0,1)	-
Commissione	Indica quale azienda ha commissionato quella campagna pubblicitaria	Campagna Pubblicitaria (1,1), Azienda (0,N)	-
Pubblicizza	Relaziona la campagna pubblicitaria alle collezioni che sta pubblicizzando	Campagna Pubblicitaria (1,N), Collezione (1,N)	-
Fornitura	Associa l'azienda alle collezioni che essa fornisce	Azienda (0,N), Collezione (1,1)	-
Composizione	Lega la collezione con i prodotti che la compongono	Collezione (1,N), Prodotto (1,1)	-
P-C	Associa il prodotto alla categoria a cui appartiene	Prodotto (1,1), Categoria (0,N)	-

Tabella 3. Dizionario dei dati - Relazioni

Workpackage	Task	Responsabile
WP4	Regole Aziendali	Ciaravola Giosuè

3.4. Regole Aziendali

Regole di Vincolo
<p>(RV1) UTENTE.Sesso DEVE essere un carattere tra uno di due valori possibili {M,F}.</p> <p>(RV2) Un UTENTE può iscriversi o disiscriversi da una newsletter a condizione che la campagna a cui è associata NON DEVE essere già conclusa.</p> <p>(RV3) UTENTE.Email DEVE essere formato da una stringa che deve essere conforme al seguente pattern: ‘^[a-zA-Z0-9]+@[a-zA-Z0-9]+\.[a-zA-Z]+\$’</p> <p>(RV3) ISCRIZIONE.DataDiIscrizione DEVE essere antecedente o uguale alla data attuale.</p> <p>(RV4) VALUTAZIONE.IstanteCalcolo DEVE corrispondere all’istante attuale nel quale viene inserito il valore del singolo KPI (formato Anno-Mese-Giorno-Ora-Minuti-Secondi).</p> <p>(RV5) VALUTAZIONE.Valore DEVE essere minore di KPI.ValMax corrispondente, se presente.</p> <p>(RV6) VALUTAZIONE.Valore DEVE essere maggiore di KPI.ValMin corrispondente, se presente.</p> <p>(RV7) CAMPAGNAPASSATA.DataDiFine DEVE essere antecedente o uguale alla data attuale.</p> <p>(RV8) CAMPAGNAPUBBLICITARIA.Codice DEVE essere un numero progressivo maggiore di 0.</p> <p>(RV9) Nel caso in cui AZIENDA.PartitaIVA sia uguale a “02986180210” (partitaIVA di Zalando) la campagna commissionata DEVE pubblicizzare collezioni che hanno lo stesso nome e lo stesso anno.</p> <p>(RV10) Nel caso in cui AZIENDA.PartitaIVA sia diversa da “02986180210” (partitaIVA di Zalando) la campagna commissionata DEVE pubblicizzare una collezione fornita da quell’azienda.</p> <p>(RV11) PRODOTTO.VotoMedio DEVE essere un numero con una cifra dopo la virgola compreso tra 0 e 5 (estremi compresi).</p> <p>(RV12) PRODOTTO.Sezione DEVE essere una stringa tra i tre valori possibili {Uomo,Donna,Bambino}.</p> <p>(RV13) Per ogni Newsletter, il valore di VALUTAZIONE.Valore associato a KPI.Nome quando Nome è ‘Tasso di Apertura’ DEVE essere maggiore o uguale di VALUTAZIONE.Valore associato a KPI.Nome quando Nome è ‘Tasso di Clic’.</p> <p>(RV14) Per ogni Newsletter, il valore di VALUTAZIONE.Valore associato a KPI.Nome quando Nome è ‘Tasso di Clic’ DEVE essere maggiore o uguale di VALUTAZIONE.Valore associato a KPI.Nome quando Nome è ‘Tasso di Conversione’.</p>

Tabella 4. Regole di vincolo

Regole di derivazione
<p>(RD1) NEWSLETTER.NumeroIscritti SI OTTIENE contando le occorrenze della relazione ISCRIZIONE associate alla NEWSLETTER corrispondente.</p>

Tabella 5. Regole di derivazione

3.5. Analisi di qualità dello schema concettuale

Al fine di garantire le proprietà generali che uno schema concettuale deve mantenere, possiamo ad analizzare le sue qualità.

- **Correttezza:** Uno schema concettuale risulta essere corretto quando utilizza propriamente i costrutti messi a disposizione dal modello concettuale di riferimento, in particolare passiamo a fare un'analisi per *ispezione* confrontando i concetti presenti nello schema in via di costruzione con le specifiche e con le definizioni dei costrutti del modello concettuale usato. Lo schema rispetta la sintassi, per esempio è stato evitato l'errore di avere un attributo identificativo in un'associazione utilizzando il pattern di reificazione, e rispetta la semantica, per esempio utilizzando una generalizzazione e non un'associazione per indicare se la campagna pubblicitaria è passata o in corso.
- **Completezza:** Uno schema concettuale risulta essere completo quando rappresenta tutti i dati di interesse e quando tutte le operazioni possono essere eseguite a partire dai concetti descritti nello schema. Lo schema rappresenta esaustivamente la realtà descritta dalle specifiche; tutte le operazioni principali possono essere svolte a partire dai concetti presenti nello schema, inoltre tutti i concetti coinvolti in un'operazione presente nelle specifiche sono raggiungibili "navigando" attraverso lo schema .
- **Leggibilità:** Uno schema concettuale risulta essere leggibile quando rappresenta i requisiti in maniera naturale e facilmente comprensibile. Lo schema risulta essere autoesplicativo, in quanto i nomi sono scritti in modo da rappresentare al meglio i concetti (e il loro significato); la struttura del diagramma è molto lineare ed evita loop o altri collegamenti convoluti; i requisiti sono rappresentati in maniera naturale a partire dai concetti fondamentali e le associazioni che li legano.
- **Minimalità:** Uno schema concettuale risulta essere minimale quando tutte le specifiche sui dati risultano essere rappresentate una sola volta nello schema, come nel caso in analisi. In questo caso l'unica ridondanza mantenuta è il numero di iscritti alle newsletter, dato che può essere utile per ridurre il numero di accessi nelle operazioni principali, e sarà analizzata in seguito.

4. Progettazione Logica

Workpackage	Task	Responsabile
WP2	Progettazione Logica	Della Corte Mario

4.1. Ristrutturazione Schema Concettuale

4.1.1. Analisi delle Prestazioni

4.1.1.1. Tavola dei volumi

Concetto	Tipo	Volume
Campagna Pubblicitaria	E	50
Campagna In Corso	E	20
Campagna Passata	E	30
Newsletter	E	50
KPI	E	9
Valutazione	E	450
Utente	E	100.000
Azienda	E	50
Collezione	E	150
Prodotto	E	1500
Categoria	E	20
Commissione	R	50
Pubblicizza	R	200
Fornitura	R	150
Composizione	R	1500
P-C	R	1500
Utilizzo	R	50
N-V	R	450
V-K	R	450
Iscrizione	R	1.000.000

Tabella 6. Tavola dei volumi

4.1.1.2. Tavola delle operazioni

Operazione	Tipo	Frequenza
Operazione 3: inserisci una nuova iscrizione di un utente a una newsletter	I	200 al giorno
Operazione 6: stampa un resoconto dei KPI di una newsletter, includendo le informazioni dei prodotti pubblicizzati, della newsletter (incluso il numero di iscritti) e della campagna di appartenenza	B	3 a settimana
Operazione 8: stampa un report dello storico delle campagne terminate o ancora in corso nella stagione appena conclusa (informazioni sulla campagna, newsletter con numero di iscritti finale, e resoconto KPI)	B	4 all'anno

Tabella 7. Tavola delle operazioni

4.2. Analisi delle ridondanze

- **Ridondanza 1:** numero iscritti (NEWSLETTER). Il numero di iscritti ad una newsletter si ottiene contando il numero di occorrenze dell'associazione ISCRIZIONE che fanno riferimento a tale newsletter.

TIPO: Attributo Derivabile da conteggio di occorrenze

4.2.1. Analisi della ridondanza 1: Numero iscritti

- **Operazione 3:** inserisci una nuova iscrizione di un utente a una newsletter (effettuata 200 volte al giorno).

Con Ridondanza

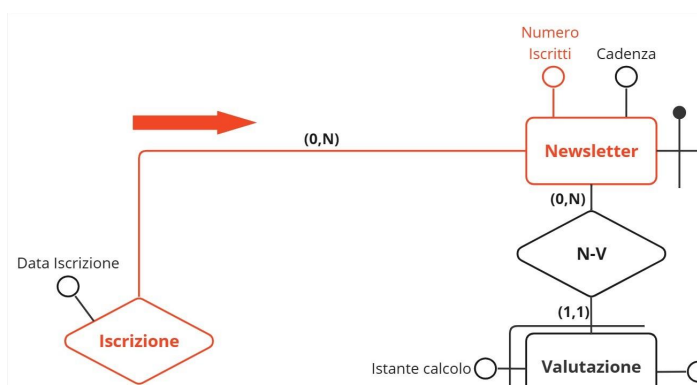


Figura 9. Schema di operazione Operazione 3 con ridondanza.

CONCETTO	CoSTRUTTO	ACCESSI	TIPO
ISCRIZIONE	R	1	S
NEWSLETTER	E	1	L
NEWSLETTER	E	1	S

Tabella 8. Tavola degli accessi Operazione 3 con ridondanza

Considerando il costo dell'accesso in scrittura il doppio di quello in lettura.

Costo = $2S + L = 5$ accessi

Costo Complessivo = $5 * 200 * 365 = 365.000$ accessi all'anno

Senza Ridondanza

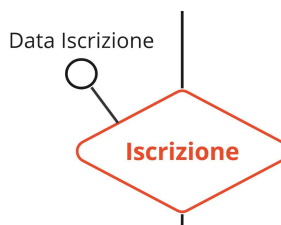


Figura 10. Schema di operazione Operazione 3 senza ridondanza.

CONCETTO	CoSTRUTTO	ACCESSI	TIPO
ISCRIZIONE	R	1	S

Tabella 10. Tavola degli accessi Operazione 3 senza ridondanza

Considerando il costo dell'accesso in scrittura il doppio di quello in lettura.

Costo = $S = 2$ accessi

Costo Complessivo = $2 * 200 * 365 = 146.000$ accessi all'anno

- **Operazione 6:** stampa un resoconto dei KPI di una newsletter, includendo le informazioni dei prodotti pubblicizzati, della newsletter (incluso il numero di iscritti) e della campagna di appartenenza (effettuata 3 volte a settimana)

Con Ridondanza

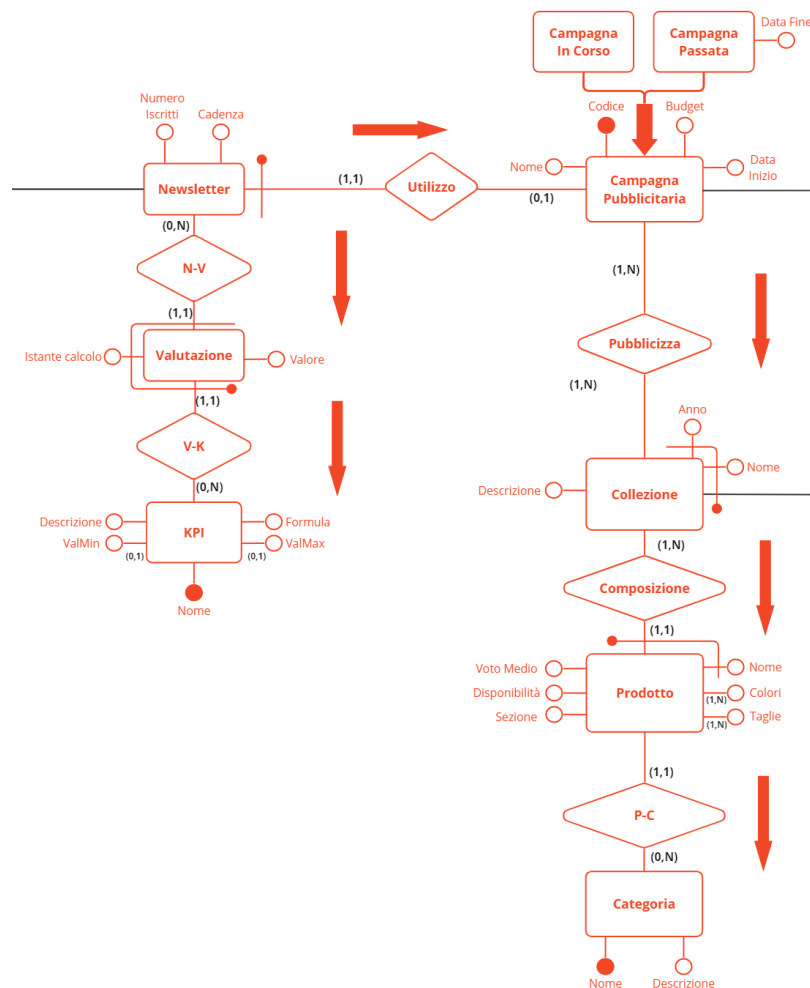


Figura 11. Schema di operazione Operazione 6 con ridondanza.

CONCETTO	COSTRUTTO	ACCESSI	TIPO
NEWSLETTER	E	1	L
N-V	R	9	L
VALUTAZIONE	E	9	L
V-K	R	9	L
KPI	E	9	L
UTILIZZO	R	1	L
CAMPAGNA PUBBLICITARIA	E	1	L
PUBBLICIZZA	R	1	L
COLLEZIONE	E	1	L
COMPOSIZIONE	R	10	L
PRODOTTO	E	10	L
P-C	R	10	L
CATEGORIA	E	10	L

Tabella 10. Tavola degli accessi Operazione 6 con ridondanza

Consideriamo che il numero di KPI è 9, e assumiamo che in media una collezione è composta da 10 prodotti, e che in media una campagna pubblicitaria 1 collezione.

Per semplicità è stata considerata una sola valutazione per ogni KPI di una newsletter.

Considerando il costo dell'accesso in scrittura il doppio di quello in lettura.

Costo = 81L = 81 accessi

Costo Complessivo = 81 * 3 * 54 = 13.122 accessi all'anno

Senza Ridondanza

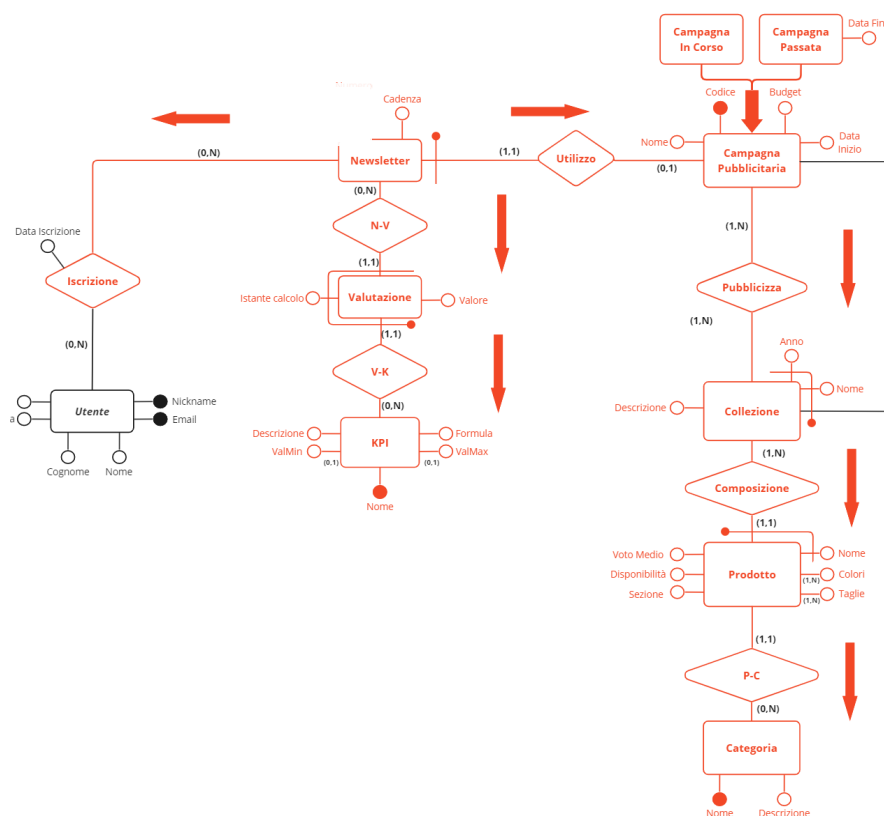


Figura 12. Schema di operazione Operazione 6 senza ridondanza.

CONCETTO	COSTRUTTO	ACCESSI	TIPO
NEWSLETTER	E	1	L
N-V	R	9	L
VALUTAZIONE	E	9	L
V-K	R	9	L
KPI	E	9	L
UTILIZZO	R	1	L
CAMPAGNA PUBBLICITARIA	E	1	L
PUBBLICIZZA	R	1	L
COLLEZIONE	E	1	L
COMPOSIZIONE	R	10	L
PRODOTTO	E	10	L
P-C	R	10	L
CATEGORIA	E	10	L
ISCRIZIONE	R	20.000	L

Tabella 11. Tavola degli accessi Operazione 6 senza ridondanza

Considerando il costo dell'accesso in scrittura il doppio di quello in lettura.

Costo = 20.081L = 20.081 accessi

Costo Complessivo = 20.081 * 3 * 54 = 3.253.122 accessi all'anno

- **Operazione 8:** stampa un report dello storico delle campagne terminate o ancora in corso nella stagione appena conclusa (informazioni sulla campagna, newsletter con numero di iscritti finale, e resoconto KPI) (effettuata 4 volte all'anno)

Con Ridondanza

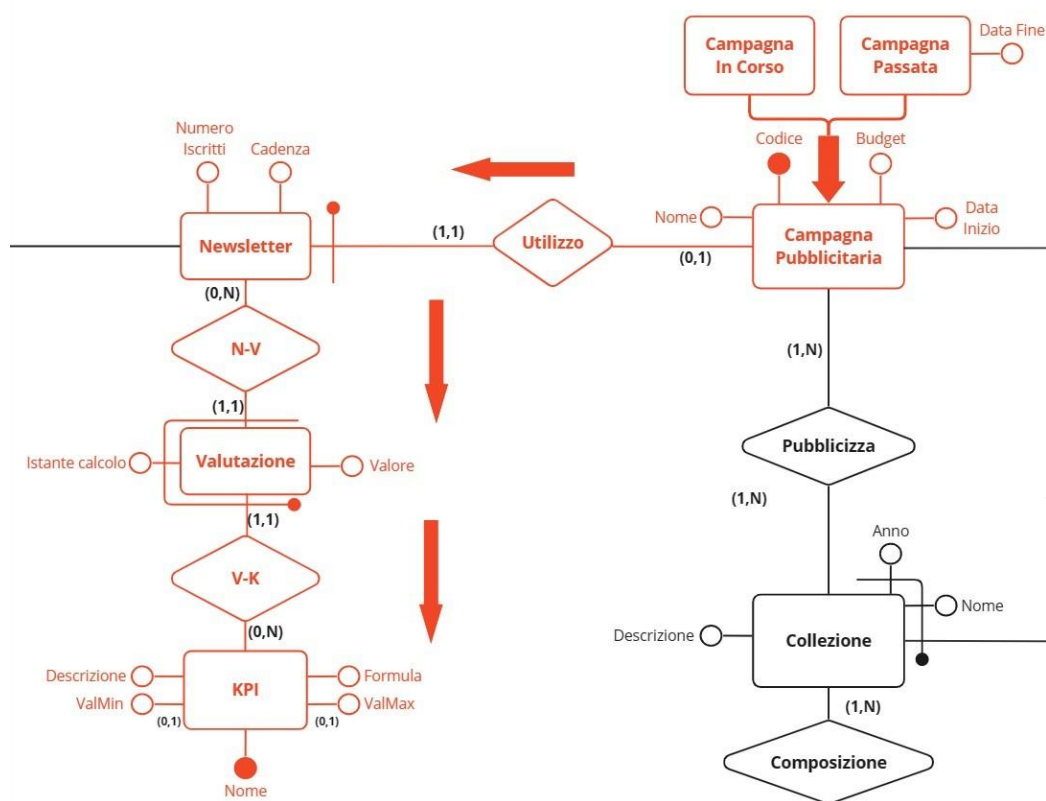


Figura 13. Schema di operazione Operazione 8 con ridondanza.

CONCETTO	COSTRUTTO	ACCESSI	TIPO
CAMPAGNA PUBBLICITARIA	E	50	L
UTILIZZO	R	50	L
NEWSLETTER	E	50	L
N-V	R	450	L
VALUTAZIONE	E	450	L
V-K	R	450	L
KPI	E	450	L

Tabella 12. Tavola degli accessi Operazione 8 con ridondanza

Considerando il costo dell'accesso in scrittura il doppio di quello in lettura.

Costo = 1.950L = 1.950 accessi

Costo Complessivo = 1.950 * 4 = 7.800 accessi all'anno

Senza Ridondanza

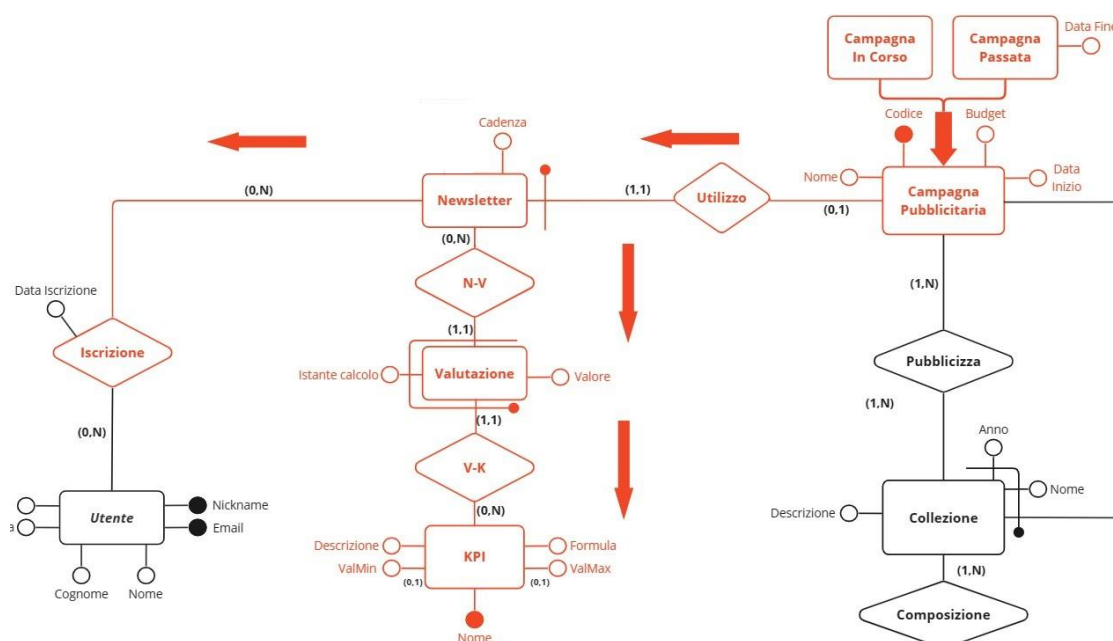


Figura 14. Schema di operazione Operazione 8 senza ridondanza.

CONCETTO	COSTRUTTO	ACCESSI	TIPO
CAMPAGNA PUBBLICITARIA	E	50	L
UTILIZZO	R	50	L
NEWSLETTER	E	50	L
N-V	R	450	L
VALUTAZIONE	E	450	L
V-K	R	450	L
KPI	E	450	L
ISCRIZIONE	R	1.000.000	L

Tabella 13. Tavola degli accessi Operazione 8 senza ridondanza

Considerando il costo dell'accesso in scrittura il doppio di quello in lettura.

Costo = 1.001.950L = 1.001.950 accessi

Costo Complessivo = 1.001.950 * 4 = 4.780.000 accessi all'anno

4.2.1.1. Valutazione della ridondanza 1

Dopo aver analizzato le operazioni che coinvolgono la ridondanza si osserva che, con il carico considerato:

- In presenza di ridondanza il costo delle operazioni è di circa 385.922 accessi annuali
- Trattandosi di un dato intero, per cui assumiamo che 4 byte siano più che sufficienti, l'occupazione di memoria è di circa 4*(Numero di occorrenze **NEWSLETTER**) byte, ovvero 200 byte (circa un 1/5 di kilobyte, trascurabile)
- In assenza di ridondanza il costo delle operazioni è di 8.179.122 accessi annuali

Pertanto, poiché riteniamo trascurabile l'occupazione di memoria aggiuntiva e, inoltre, gli accessi in assenza di ridondanza sono maggiori, si decide di mantenere la ridondanza in quanto riduce sostanzialmente il numero di accessi.

4.3. Eliminazione delle generalizzazioni

4.3.1. Generalizzazione *Campagna Pubblicitaria*

Dato che la generalizzazione è totale, è possibile intraprendere tutte e tre le strade possibili della ristrutturazione di questo costrutto: accorpamento delle entità figlie nell'entità padre; accorpamento dell'entità padre nelle entità figlie; sostituzione della generalizzazione con associazioni.

Tenendo in considerazione che non ci sono operazioni principali che fanno distinzione tra le diverse entità figlie, e che non ci sono associazioni che coinvolgono solo una delle entità figlie, anche se abbiamo uno spreco di memoria dovuto ai valori nulli, la nostra scelta ricade sulla prima opzione. Ciò ci assicura anche un numero minore di accessi rispetto alle altre scelte nelle quali le occorrenze e gli attributi sono distribuiti tra le varie entità.

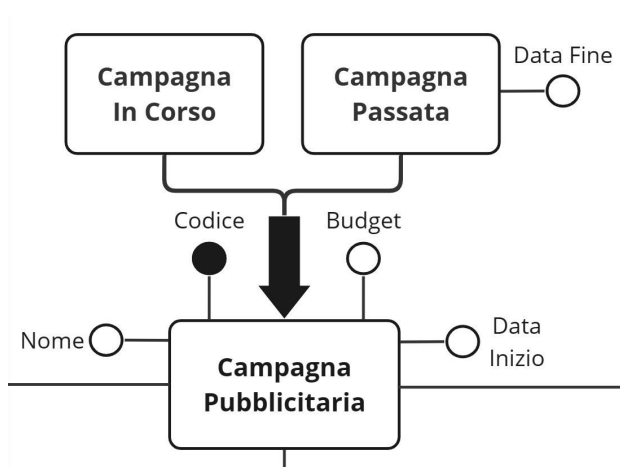


Figura 15. Prima della ristrutturazione della generalizzazione.

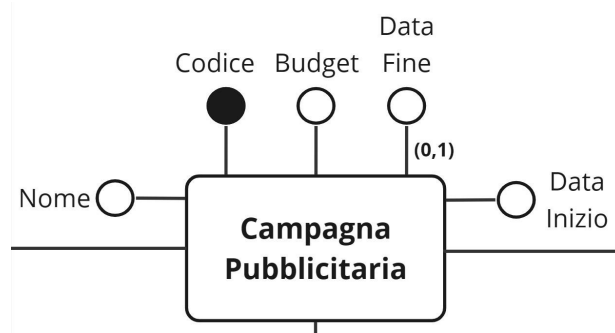


Figura 16. Dopo la ristrutturazione della generalizzazione

Si è implementata la prima opzione utilizzando direttamente l'attributo *Data Fine* per stabilire se la campagna è terminata: se non è valorizzato allora la campagna è in corso, se è valorizzato allora la campagna è passata.

4.4. Partizionamento/Accorpamento Entità e Associazioni

4.4.1. Eliminazione attributo multivalore Recapito Telefonico (di AZIENDA)

Il primo attributo multivalore lo troviamo all'interno dell'entità AZIENDA.

Nella ristrutturazione, l'attributo multivalore viene reificato in una entità NUMERO TELEFONICO che si collega all'entità AZIENDA tramite l'associazione RECAPITO.

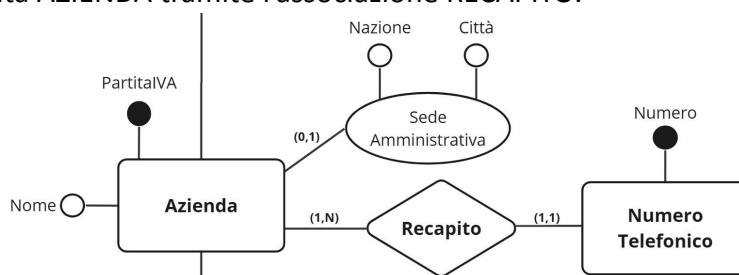


Figura 17. Dopo l'eliminazione dell'attributo multivalore Recapito Telefonico (di AZIENDA)

4.4.2. Eliminazione attributo multivalore Colori (di PRODOTTO)

Il secondo attributo multivalore lo troviamo all'interno dell'entità PRODOTTO.

Nella ristrutturazione, l'attributo multivalore viene reificato in una entità COLORE che si collega all'entità PRODOTTO tramite l'associazione *Colori assortiti*.

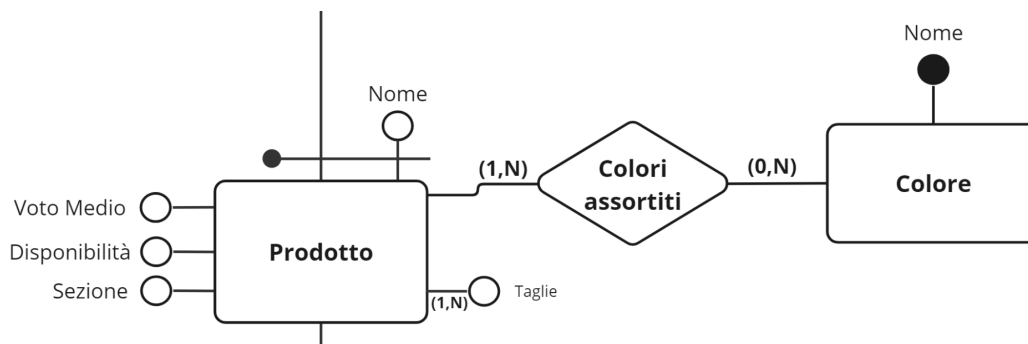


Figura 18. Dopo l'eliminazione dell'attributo multivalore Colori (di PRODOTTO)

4.4.3. Eliminazione attributo multivalore Taglie (di PRODOTTO)

Il terzo attributo multivalore lo troviamo all'interno dell'entità PRODOTTO.

Nella ristrutturazione, l'attributo multivalore viene reificato in una entità TAGLIA che si collega all'entità PRODOTTO tramite l'associazione *TAGLIE ASSORTITE*.

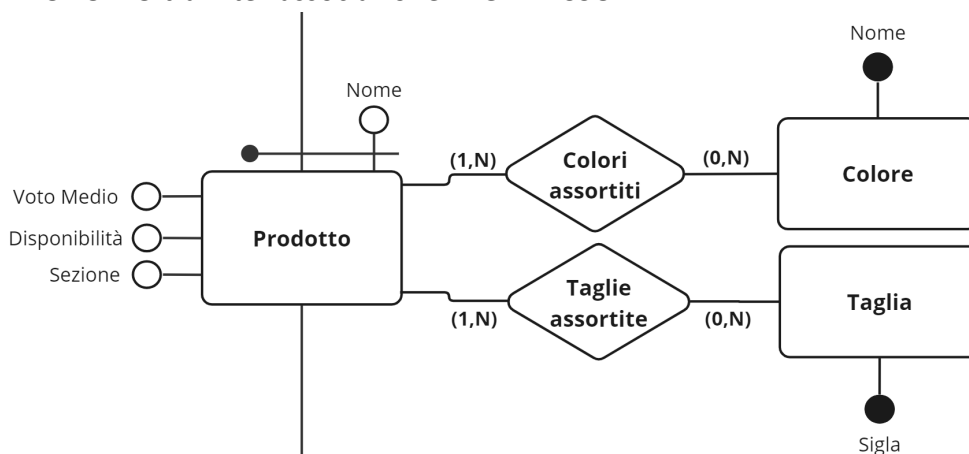


Figura 19. Dopo l'eliminazione dell'attributo multivalore Taglie (di PRODOTTO)

4.4.4. Eliminazione attributo composto Sede Amministrativa (di AZIENDA)

Nel nostro schema concettuale troviamo un solo caso di attributo composto, all'interno dell'entità AZIENDA.

La soluzione applicata è stata quella di scindere l'attributo composto in attributi semplici, inserendo il vincolo per il quale o sono specificati entrambi, o nessuno di loro è specificato.

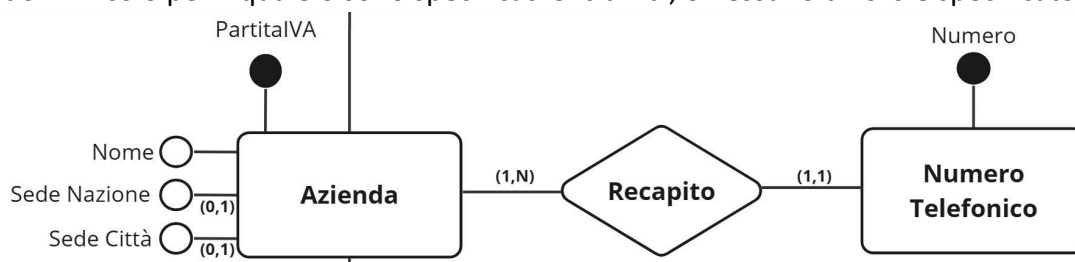


Figura 20. Dopo l'eliminazione dell'attributo composto Sede Amministrativa (di AZIENDA)

4.5. Scelta degli identificatori principali

- CAMPAGNA PUBBLICITARIA: Codice

- NEWSLETTER: CAMPAGNA PUBBLICITARIA

- VALUTAZIONE: Id

In questo caso, a causa dell'identificatore misto, composto da più attributi, si è ritenuto utile introdurre un id numerico (Id).

- KPI: Nome

- UTENTE: Nickname

In questo caso, vista la presenza di più identificatori, si è scelto di promuovere il Nickname come principale, visto che in media ha una stringa più corta rispetto alla Email.

- AZIENDA: PartitaIVA

- NUMERO TELEFONICO: Numero

- COLLEZIONE: Id

Si è introdotto un Id per il motivo analogo al caso della VALUTAZIONE.

- PRODOTTO: Id

Si è introdotto un Id per il motivo analogo al caso della VALUTAZIONE.

- COLORE: Nome

- TAGLIA: Sigla

- CATEGORIA: Nome

Nei casi in cui si è aggiunto l'Id, o in cui si è scelto, tra due, l'identificatore principale, sono stati lasciati anche gli altri come identificatori alternativi.

Nello schema ristrutturato, gli identificatori principali sono evidenziati in **grassetto**.

4.6. Schema ristrutturato finale

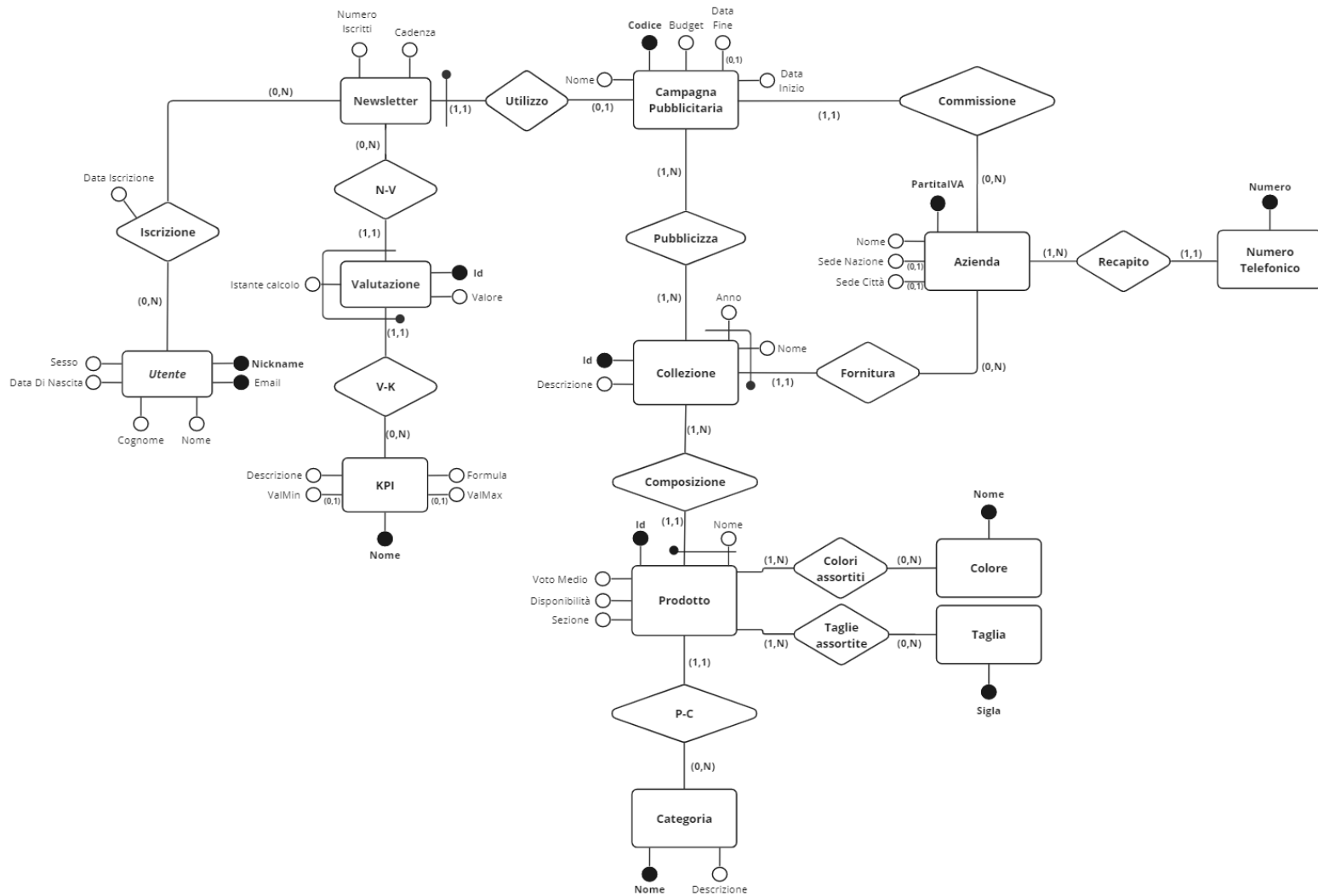


Figura 21. Schema ristrutturato finale

4.7. Schema logico

CAMPAGNA_PUBBLICITARIA (codice, nome, budget, data_inizio, data_fine*, azienda_committente)

azienda_committente → AZIENDA.partita_IVA

AZIENDA (partita_IVA, nome, sede_nazione*, sede_citta*)

NUMERO_TELEFONICO (numero, azienda)

azienda → AZIENDA.partita_IVA

CATEGORIA (nome, descrizione)

KPI (nome, descrizione, formula, val_min*, val_max*)

UTENTE (nickname, email, nome, cognome, sesso, data_di_nascita)

NEWSLETTER (campagna, cadenza, numero_iscritti)

campagna → CAMPAGNA_PUBBLICITARIA.codice

VALUTAZIONE (id, newsletter, kpi, istante_calcolo, valore)

newsletter → NEWSLETTER.campagna

kpi → KPI.nome

COLLEZIONE (id, azienda_fornitrice, nome, anno, descrizione)

azienda_fornitrice → AZIENDA.partita_IVA

PRODOTTO (id, collezione, nome, voto_medio, disponibilita, sezione, categoria)

collezione → COLLEZIONE.id

categoria → CATEGORIA.nome

COLORI_ASSORTITI (prodotto, colore)

prodotto → PRODOTTO.id

colore → COLORE.nome

COLORE (nome)

TAGLIE_ASSORTITE (prodotto, taglia)

prodotto → PRODOTTO.id

taglia → TAGLIA.sigla

TAGLIA (sigla)

ISCRIZIONE (newsletter, utente, data_iscrizione)

newsletter → NEWSLETTER.campagna

utente → UTENTE.nickname

PUBBLICIZZA (campagna, collezione)

campagna → CAMPAGNA_PUBBLICITARIA.codice

collezione → COLLEZIONE.id

La chiave primaria è indicata con grassetto e sottolineatura, chiavi alternative (primarie in senso relazionale) sono solo sottolineate.

[AttributoRelazioneInterna → RelazioneEsterna.Attributo] indica un vincolo di integrità referenziale.

Gli attributi seguiti da * sono attributi opzionali

4.8. Documentazione dello schema logico

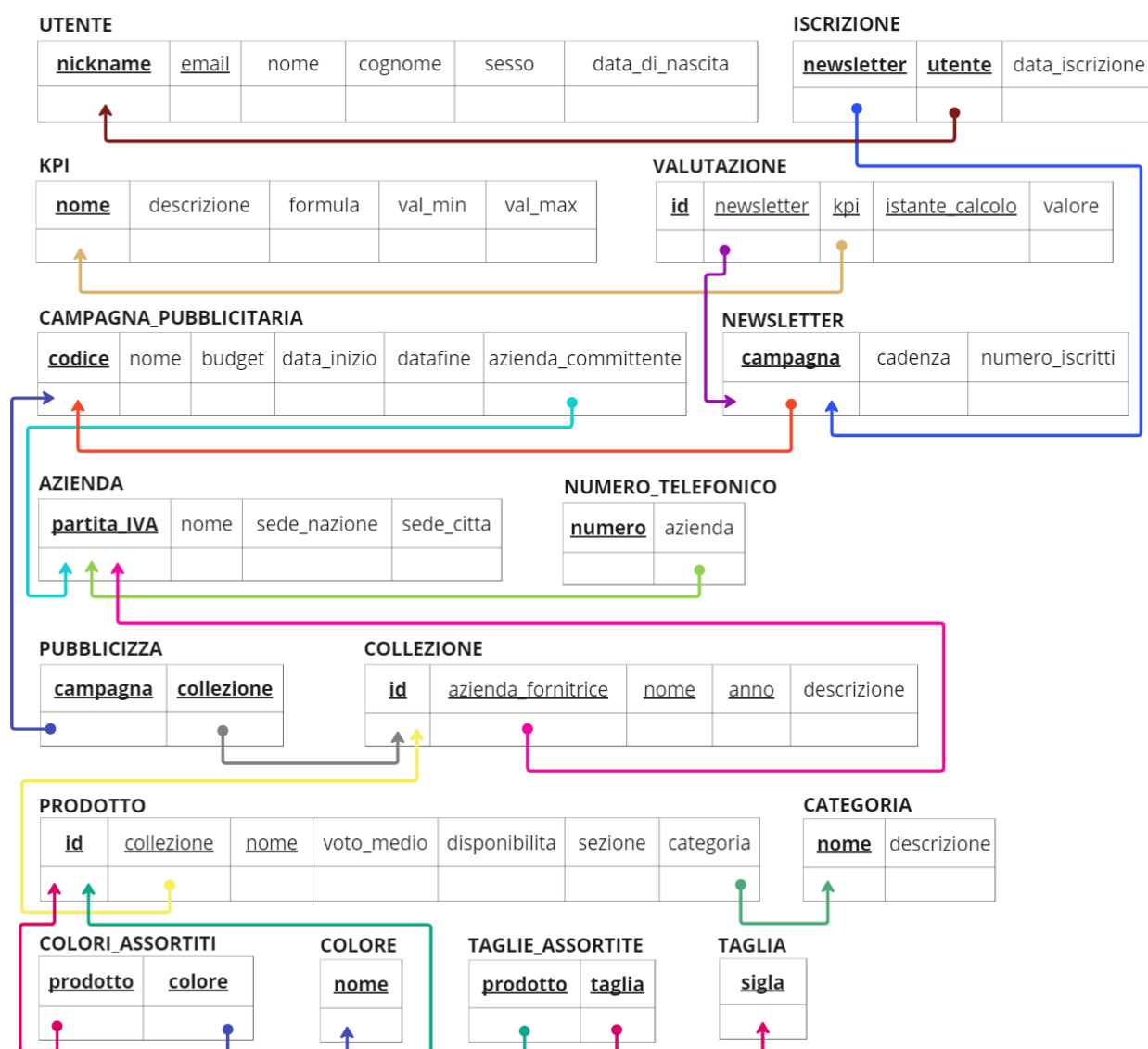


Figura 22. Schema logico con vincoli di integrità referenziale

4.8.1. Vincoli della ristrutturazione

Regole di Ristrutturazione
(VR1) Gli attributi AZIENDA.sede_nazione e AZIENDA.sede_citta DEVONO essere entrambi valorizzati o entrambi null.

Tabella 14. Regole di ristrutturazione

4.8.2. Vincoli della traduzione

Regole di Traduzione
(VT1) Ogni tupla di CAMPAGNA_PUBBLICITARIA DEVE essere referenziata da almeno una tupla di PUBBLICIZZA .
(VT2) Ogni tupla di COLLEZIONE DEVE essere referenziata da almeno una tupla di PUBBLICIZZA .
(VT3) Ogni tupla di AZIENDA DEVE essere referenziata da almeno una tupla di NUMERO_TELEFONICO .
(VT4) Ogni tupla di COLLEZIONE DEVE essere referenziata da almeno una tupla di PRODOTTO .
(VT5) Ogni tupla di PRODOTTO DEVE essere referenziata da almeno una tupla di COLORI_ASSORTITI .
(VT6) Ogni tupla di PRODOTTO DEVE essere referenziata da almeno una tupla di TAGLIE_ASSORTITE .

Tabella 15. Regole di traduzione

5. Normalizzazione

Workpackage	Task	Responsabile
WP3	Normalizzazione	Esposito Maurizio

Ai fini della normalizzazione saranno ovviamente considerate solo le dipendenze funzionali minimali e non banali.

5.1. CAMPAGNA_PUBBLICITARIA

5.1.1. Dipendenze funzionali

codice → nome codice → budget codice → data_inizio

codice → data_fine codice →
azienda_committente

5.1.2. Forma normale raggiunta

Visto che:

- tutti gli attributi sono atomici (1NF);
- per ogni dipendenza funzionale (non banale) la parte sinistra è una superchiave della relazione (in questo caso la chiave stessa);

si può affermare che la relazione è in **forma normale di Boyce e Codd**.

5.2. AZIENDA

5.2.1. Dipendenze funzionali

partita_IVA → nome partita_IVA → sede_nazione partita_IVA → sede_citta

5.2.2. Forma normale raggiunta

Visto che:

- tutti gli attributi sono atomici (1NF);
- per ogni dipendenza funzionale (non banale) la parte sinistra è una superchiave della relazione (in questo caso la chiave stessa);

si può affermare che la relazione è in **forma normale di Boyce e Codd**.

5.3. NUMERO_TELEFONICO

5.3.1. Dipendenze funzionali

Essendo una relazione composta da un solo attributo (che ovviamente è chiave primaria), non ci sono dipendenze funzionali non banali.

5.3.2. Forma normale raggiunta

Per lo stesso motivo la relazione è sicuramente in **forma normale di Boyce e Codd**.

5.4. CATEGORIA

5.4.1. Dipendenze funzionali

nome → descrizione

5.4.2. Forma normale raggiunta

Visto che:

- tutti gli attributi sono atomici (1NF);
- per ogni dipendenza funzionale (non banale) la parte sinistra è una superchiave della relazione (in questo caso la chiave stessa);

si può affermare che la relazione è in **forma normale di Boyce e Codd**.

5.5. KPI

5.5.1. Dipendenze funzionali

nome → descrizione

nome → formula

nome → val_min

nome → val_max

5.5.2. Forma normale raggiunta

Visto che:

- tutti gli attributi sono atomici (1NF);
- per ogni dipendenza funzionale (non banale) la parte sinistra è una superchiave della relazione (in questo caso la chiave stessa);

si può affermare che la relazione è in **forma normale di Boyce e Codd**.

5.6. UTENTE

5.6.1. Dipendenze funzionali

nickname → email

nickname → nome

nickname → cognome

nickname → sesso

nickname → data_di_nascita

email → nickname

email → nome

email → cognome

email → sesso

email → data_di_nascita

5.6.2. Forma normale raggiunta

Visto che:

- tutti gli attributi sono atomici (1NF);
- per ogni dipendenza funzionale (non banale) la parte sinistra è una superchiave della relazione (in questo caso la chiave stessa);

si può affermare che la relazione è in **forma normale di Boyce e Codd**.

5.7. NEWSLETTER

5.7.1. Dipendenze funzionali

campagna → cadenza

campagna → numero_iscritti

5.7.2. Forma normale raggiunta

Visto che:

- tutti gli attributi sono atomici (1NF);
- per ogni dipendenza funzionale (non banale) la parte sinistra è una superchiave della relazione (in questo caso la chiave stessa);

si può affermare che la relazione è in **forma normale di Boyce e Codd**.

5.8. VALUTAZIONE

5.8.1. Dipendenze funzionali

id → newsletter

id → kpi

id → istante_calcolo

id → valore

newsletter, kpi, istante_calcolo → id

newsletter, kpi, istante_calcolo → valore

5.8.2. Forma normale raggiunta

Visto che:

- tutti gli attributi sono atomici (1NF);
- per ogni dipendenza funzionale (non banale) la parte sinistra è una superchiave della relazione (in questo caso la chiave stessa);

si può affermare che la relazione è in **forma normale di Boyce e Codd**.

5.9. COLLEZIONE

5.9.1. Dipendenze funzionali

id → nome

id → anno

id → azienda_fornitrice

id → descrizione

azienda_fornitrice, nome, anno → id

azienda_fornitrice, nome, anno → descrizione

5.9.2. Forma normale raggiunta

Visto che:

- tutti gli attributi sono atomici (1NF);
- per ogni dipendenza funzionale (non banale) la parte sinistra è una superchiave della relazione (in questo caso la chiave stessa);

si può affermare che la relazione è in **forma normale di Boyce e Codd**.

5.10. PRODOTTO

5.10.1. Dipendenze funzionali

id → collezione

id → nome

id → voto_medio

id → disponibilita

id → sezione

id → categoria

collezione, nome → id

collezione, nome → voto_medio

collezione, nome → disponibilita

collezione, nome → sezione

collezione, nome → categoria

5.10.2. Forma normale raggiunta

Visto che:

- tutti gli attributi sono atomici (1NF);
- per ogni dipendenza funzionale (non banale) la parte sinistra è una superchiave della relazione (in questo caso la chiave stessa);

si può affermare che la relazione è in **forma normale di Boyce e Codd**.

5.11. COLORI_ASSORTITI

5.11.1. Dipendenze funzionali

Essendo una relazione composta da una coppia di attributi che forma la chiave primaria, non ci sono dipendenze funzionali non banali.

5.11.2. Forma normale raggiunta

Per lo stesso motivo la relazione è sicuramente in **forma normale di Boyce e Codd**.

5.12. COLORE

5.12.1. Dipendenze funzionali

Essendo una relazione composta da un singolo attributo che è anche la chiave primaria, non ci sono dipendenze funzionali non banali.

5.12.2. Forma normale raggiunta

Per lo stesso motivo la relazione è sicuramente in **forma normale di Boyce e Codd**.

5.13. TAGLIE_ASSORTITE

5.13.1. Dipendenze funzionali

Essendo una relazione composta da una coppia di attributi che forma la chiave primaria, non ci sono dipendenze funzionali non banali.

5.13.2. Forma normale raggiunta

Per lo stesso motivo la relazione è sicuramente in **forma normale di Boyce e Codd**.

5.14. TAGLIA

5.14.1. Dipendenze funzionali

Essendo una relazione composta da un singolo attributo che è anche la chiave primaria, non ci sono dipendenze funzionali non banali.

5.14.2. Forma normale raggiunta

Per lo stesso motivo la relazione è sicuramente in **forma normale di Boyce e Codd**.

5.15. ISCRIZIONE

5.15.1. Dipendenze funzionali

newsletter, utente → data_iscrizione

5.15.2. Forma normale raggiunta

Visto che:

- tutti gli attributi sono atomici (1NF);
- per ogni dipendenza funzionale (non banale) la parte sinistra è una superchiave della relazione (in questo caso la chiave stessa);

si può affermare che la relazione è in **forma normale di Boyce e Codd**.

5.16. PUBBLICIZZA

5.16.1. Dipendenze funzionali

Essendo una relazione composta da una coppia di attributi che forma la chiave primaria, non ci sono dipendenze funzionali non banali.

5.16.2. Forma normale raggiunta

Per lo stesso motivo la relazione è sicuramente in **forma normale di Boyce e Codd**.

6. Script Creazione e Popolamento Database

Workpackage	Task	Responsabile
WP2	SQL: Script creazione e popolamento	Della Corte Mario

6.1. Generated Always As Identity

In fase di creazione del database è stata utilizzata la clausola **“GENERATED ALWAYS AS IDENTITY”** per quegli attributi che rappresentano un codice o id numerico che parte da 1 e va incrementato dopo ogni inserimento (codice della campagna, id di collezione, prodotto e valutazione).

La clausola rende la colonna a cui è applicata una “identity column”, ossia una colonna a cui è associata una sequenza numerica implicita che assegna valori progressivi per ogni tupla inserita. In questo modo le insert non devono specificare il codice/id. Inoltre la keyword **“ALWAYS”** non permette all’utente di specificare, nella insert, un valore per quella colonna (se non tramite un override), e non permette nemmeno di modificarne il valore (tranne se il valore è **“default”**) in un secondo momento tramite una update.

La clausola è conforme allo standard SQL (al contrario del tipo SERIAL, che serve allo stesso scopo ma viene sconsigliato) ed è supportata da PostgreSQL dalla versione 10.

6.2. Note sul popolamento

Per una questione di comodità gli inserimenti sono stati divisi per tabelle; per soddisfare i trigger di cardinalità minima lo script va eseguito nella sua interezza.

In una situazione di utilizzo comune, però, per rispettare le cardinalità minime, per esempio, un'azienda va inserita nella stessa transazione insieme ad almeno un suo numero telefonico; Per inserire una campagna vanno inseriti anche l'azienda committente, almeno una collezione, l'azienda fornitrice, almeno un prodotto con una categoria, taglia e colore assortita (se non già presenti).

E’ mostrato in seguito un esempio di transazione per l’inserimento di una azienda e il suo numero telefonico:

BEGIN

```
INSERT INTO AZIENDA (partita_IVA, nome, sede_nazione, sede_citta)
VALUES ('12345678901', 'Ferrari', 'Italia', 'Maranello')
```

```
INSERT INTO NUMERO_TELEFONICO (numero, azienda)
VALUES ('+391234567890', '12345678901'),
        ('+391234567891', '12345678901')
```

COMMIT

6.3. Script di creazione e popolamento

Data la lunghezza del file abbiamo deciso non includere lo script nella relazione, ma di fornirlo come allegato. In particolare:

- Lo script di creazione e popolamento tabelle è "CreazionePopolamento.sql".

NB: Si precisa che lo script di creazione dei trigger che seguiranno dovrà essere eseguito successivamente al popolamento, per i motivi spiegati nel *Paragrafo 9.2.1*

7. Query SQL

Workpackage	Task	Responsabile
WP3	SQL: Query	Esposito Maurizio

Le seguenti query sono anche allegate nel file "Query.sql"

7.1. Query con operatore di aggregazione e join: Tasso di conversione medio per ogni campagna pubblicitaria

Questa query ha come risultato una tabella che associa al codice e al nome di ogni campagna la media dei valori registrati nel tempo del KPI "Tasso di Conversione", associati alla newsletter della campagna stessa. Le tuple ottenute sono in ordine decrescente in base alla media calcolata.

```
SELECT CP.codice AS codice_campagna, CP.nome AS nome_campagna,
ROUND(AVG(VA.valore)::numeric, 4) AS media_tasso_conversione
FROM CAMPAGNA_PUBBLICITARIA CP
JOIN NEWSLETTER NS ON CP.codice = NS.campagna
JOIN VALUTAZIONE VA ON NS.campagna = VA.newsletter
JOIN KPI K ON VA.kpi = K.nome
WHERE K.nome = 'Tasso di conversione'
GROUP BY CP.codice, CP.nome
ORDER BY media_tasso_conversione DESC;
```

L'output restituito dalla query in seguito al popolamento è il seguente.

	codice_campagna integer	nome_campagna character varying	media_tasso_conversione numeric
1	3	Campagna Dior 2023	0.5500
2	6	Campagna Zalando 2022	0.5100
3	2	Campagna Ferrari 2022	0.5000
4	1	Campagna Ferrari 2023	0.1050

7.2. Query nidificata complessa: Campagne in corso che pubblicizzano almeno un prodotto disponibile di categoria “maglie”

In SQL è possibile scrivere interrogazioni che presentano al loro interno altre interrogazioni, ed in questo caso si parla appunto di “query nidificate”. La nidificazione può avvenire sia nella clausola *SELECT*, sia nella clausola *FROM* sia nella clausola *WHERE*. Nel caso in questione la query sottostante restituisce le informazioni sulle campagne pubblicitarie attive (dunque che hanno il valore dell’attributo *data_fine* settato a *null*) che promuovono almeno un prodotto della categoria “Maglie” che risulti disponibile, ed ordina i risultati in base alla data di inizio della campagna pubblicitaria.

```
SELECT CP.codice, CP.nome, CP.data_inizio, CP.budget,
CP.azienda_committente
FROM CAMPAGNA_PUBBLICITARIA CP
  WHERE CP.data_fine IS NULL AND EXISTS (
    SELECT *
    FROM PUBBLICIZZA P
    JOIN PRODOTTO PR ON P.collezione = PR.collezione
    JOIN CATEGORIA C ON PR.categoria = C.nome
    WHERE P.campagna = CP.codice
      AND PR.disponibilita = TRUE
      AND C.nome = 'Maglie'
  )
ORDER BY CP.data_inizio;
```

L’output restituito dalla query in seguito al popolamento è il seguente:

	codice [PK] integer	nome character varying (100)	data_inizio date	budget bigint	azienda_committente character varying (11)
1	7	Campagna Zara 2023	2021-07-01	800000	45678901234
2	1	Campagna Ferrari 2023	2023-01-01	500000	12345678901
3	17	Campagna Mango 2023	2023-04-10	500000	90123456789
4	15	Campagna Tom Tailor 2023	2023-05-15	200000	89012345678

7.3. Query insiemistica: Collezioni pubblicizzate da Zalando ma non fornite da Etam

Ricordando che una query insiemistica combina più operazioni insiemistiche, come l'unione, l'intersezione e la differenza, per ottenere risultati basati sulla logica degli insiemi; si ricava la seguente query, la quale utilizza l'operatore *EXCEPT* per ottenere la differenza tra le collezioni pubblicizzate da Zalando e quelle prodotte da Nike. La prima parte della query seleziona tutte le collezioni pubblicizzate da Zalando, mentre la seconda parte seleziona le collezioni prodotte da Nike. L'operatore *EXCEPT* restituisce solo le collezioni che sono presenti nella prima parte ma non nella seconda, cioè le collezioni pubblicizzate da Zalando ma non prodotte da Etam. Le tuple ottenute sono in ordine crescente in base all'anno della collezione.

```
SELECT CO.id, CO.nome, CO.anno, CO.azienda_fornitrice, CO.descrizione
FROM COLLEZIONE AS CO
JOIN PUBBLICIZZA AS P ON CO.id = P.collezione
JOIN CAMPAGNA_PUBBLICITARIA AS CP ON P.campagna = CP.codice
JOIN AZIENDA AS A ON CP.azienda_committente = A.partita_IVA
WHERE A.partita_IVA = '02986180210' -- partita IVA di Zalando
EXCEPT
SELECT CO.id, CO.nome, CO.anno, CO.azienda_fornitrice, CO.descrizione
FROM COLLEZIONE AS CO
JOIN AZIENDA AS A ON CO.azienda_fornitrice = A.partita_IVA
WHERE A.partita_IVA = '78901234567' -- partita IVA di Etam
ORDER BY anno;
```

L'output restituito dalla query in seguito al popolamento è il seguente:

	id integer	nome character varying (50)	anno numeric (4)	azienda_fornitrice character varying (11)	descrizione character varying (255)
1	3	Collezione Autunno 2022	2022	01236667890	Collezione autunnale per Piazza Italia
2	13	Collezione Primavera 2023	2023	01236667890	Collezione primaverile per Piazza Italia

7.4. Eventuali Altre query

7.4.1. Profilazione

la seguente query estrae informazioni sugli utenti che si sono iscritti a newsletter che pubblicizzano una collezione che contiene la parola 'Estate' nel nome. Vengono quindi conteggiate le iscrizioni di ciascun utente e vengono inclusi all'interno dell'output solo quelli con più di una iscrizione. I risultati sono ordinati per cognome. La query è utile a scopi di profilazione dell'utente.

```
SELECT U.cognome, U.nome, U.email, COUNT(I.newsletter) AS numero_iscrizioni
UTENTE UTENTE AS U
JOIN ISCRIZIONE AS I ON U.nickname = I.utente
JOIN NEWSLETTER AS N ON I.newsletter = N.campagna
JOIN CAMPAGNA_PUBBLICITARIA AS CP ON N.campagna = CP.codice
JOIN PUBBLICIZZA AS P ON CP.codice = P.campagna
JOIN COLLEZIONE AS C ON P.collezione = C.id
JOIN C.nome LIKE '%Estate%'
GROUP BY U.nome, U.cognome, U.email
HAVING COUNT(I.newsletter) > 1
ORDER BY U.cognome;
```

L'output restituito dalla query in seguito al popolamento è il seguente:

	cognome character varying	nome character varying	email character varying	numero_iscrizioni bigint
1	Brown	Emily	user4@example.com	2
2	Gaeta	Matteo	user9@example.com	2
3	Johnson	Michael	user3@example.com	2
4	Smith	Jane	user2@example.com	2
5	Thomas	Jessica	user8@example.com	2
6	White	Mia	user10@example.com	2

8. Viste

Workpackage	Task	Responsabile
WP4	Viste	Ciaravola Giosuè

Le viste vengono definite in SQL associando un nome ed una lista di attributi al risultato dell'esecuzione di un'interrogazione. Ricordiamo che nel particolare l'ordine degli attributi nella *select* deve corrispondere all'ordine degli attributi nello schema.

Le seguenti viste e query sono anche allegate nel file "Query.sql"

8.1. Vista: VistaAziendaNewsletterMediaValutazioni

La query sottostante crea una vista che riassume le informazioni delle campagne pubblicitarie di Ferrari, includendo il codice della campagna, il nome della campagna, il budget, le date di inizio e fine, la cadenza della newsletter, il numero di iscritti, i KPI e la media delle valutazioni.

```

DROP VIEW IF EXISTS VistaAziendaNewsletterMediaValutazioni;

CREATE VIEW VistaAziendaNewsletterMediaValutazioni AS
SELECT CP.codice AS codice_campagna, CP.nome AS nome_campagna, CP.budget,
       CP.data_inizio, CP.data_fine,
       N.cadenza AS cadenza_newsletter, N.numero_iscritti AS
num_iscritti_newsletter,
       V.kpi, ROUND(AVG(V.valore)::numeric,4) AS media_valutazione
FROM CAMPAGNA_PUBBLICITARIA CP
JOIN AZIENDA A ON CP.azienda_committente = A.partita_IVA
JOIN NEWSLETTER N ON CP.codice = N.campagna
JOIN VALUTAZIONE V ON N.campagna = V.newsletter
WHERE A.partita_IVA = '12345678901' -- partita IVA di Ferrari
GROUP BY CP.codice, CP.nome, CP.budget, CP.data_inizio, CP.data_fine,
         N.campagna, N.cadenza, N.numero_iscritti, V.kpi
ORDER BY CP.codice;

```

In particolare l'output fornito successivamente all'esecuzione è il seguente:

	codice_campagna integer	nome_campagna character varying	budget bigint	data_inizio date	data_fine date	cadenza_newsletter integer	num_iscritti_newsletter integer	kpi character varying	media_valutazione numeric
1	1	Campagna Ferrari 2023	500000	2023-01-01	[null]	3		4 Feedback	0.8650
2	1	Campagna Ferrari 2023	500000	2023-01-01	[null]	3		4 Spaccato per device	0.7000
3	1	Campagna Ferrari 2023	500000	2023-01-01	[null]	3		4 Tasso di apertura	0.7850
4	1	Campagna Ferrari 2023	500000	2023-01-01	[null]	3		4 Tasso di clic	0.4900
5	1	Campagna Ferrari 2023	500000	2023-01-01	[null]	3		4 Tasso di conversione	0.1050
6	1	Campagna Ferrari 2023	500000	2023-01-01	[null]	3		4 Tasso di crescita	0.0250
7	1	Campagna Ferrari 2023	500000	2023-01-01	[null]	3		4 Tasso di disiscrizione	0.0150
8	1	Campagna Ferrari 2023	500000	2023-01-01	[null]	3		4 Tasso di forward e condivisione	0.0650
9	1	Campagna Ferrari 2023	500000	2023-01-01	[null]	3		4 Tasso di risposta	0.0250
10	2	Campagna Ferrari 2022	750000	2022-01-01	2022-12-31	5		4 Feedback	0.8000
11	2	Campagna Ferrari 2022	750000	2022-01-01	2022-12-31	5		4 Spaccato per device	0.6000
12	2	Campagna Ferrari 2022	750000	2022-01-01	2022-12-31	5		4 Tasso di apertura	0.7500
13	2	Campagna Ferrari 2022	750000	2022-01-01	2022-12-31	5		4 Tasso di clic	0.6500
14	2	Campagna Ferrari 2022	750000	2022-01-01	2022-12-31	5		4 Tasso di conversione	0.5000
15	2	Campagna Ferrari 2022	750000	2022-01-01	2022-12-31	5		4 Tasso di crescita	0.0500
16	2	Campagna Ferrari 2022	750000	2022-01-01	2022-12-31	5		4 Tasso di disiscrizione	0.1000
17	2	Campagna Ferrari 2022	750000	2022-01-01	2022-12-31	5		4 Tasso di forward e condivisione	0.3000
18	2	Campagna Ferrari 2022	750000	2022-01-01	2022-12-31	5		4 Tasso di risposta	0.2000

8.1.1. Query con Vista: *MigliorFeedbackCampagna*

Considerando che un'azienda ha diverse campagne pubblicitarie in corso, abbiamo deciso di implementare una query che permette di ottenere informazioni più dettagliate sull'andamento di tali campagne. In particolare, questa query restituisce la campagna ancora in corso che riceve il feedback medio più positivo da parte degli utenti, indicando quindi quella maggiormente apprezzata.

```
SELECT *
FROM VistaAziendaNewsletterMediaValutazioni V
WHERE V.data_fine IS NULL -- Solo campagne pubblicitarie non ancora concluse
AND V.kpi = 'Feedback' -- Considera solo il KPI di feedback
AND V.media_valutazione = (SELECT MAX(media_valutazione)
FROM VistaAziendaNewsletterMediaValutazioni
WHERE kpi = 'Feedback');
```

In particolare l'output della query risulta essere il seguente:

	codice_campagna integer	nome_campagna character varying	budget bigint	data_inizio date	data_fine date	cadenza_newsletter integer	num_iscritti_newsletter integer	kpi character varying	media_valutazione numeric
1	1	Campagna Ferrari 2023	500000	2023-01-01	[null]	3	4	Feedback	0.8650

9. Trigger

9.1. Trigger Inizializzazione

<i>Workpackage</i>	<i>Task</i>	<i>Responsabile</i>
WP1	Trigger inizializzazione/popoloamento database	Bruno Salvatore

Il trigger è un costrutto estremamente potente che consente di rendere la base di dati in grado di reagire ad eventi definiti dall'amministratore tramite l'esecuzione di opportune azioni. Una base di dati con tale capacità si dice attiva e dispone di un sottosistema integrato per definire e gestire regole di produzione. Le regole seguono il paradigma: *Evento-Condizione-Azione*.

I seguenti Trigger sono anche allegati nel file "Trigger.sql"

9.1.1. Trigger 1: *AlmenoUnPubblicizza()*

Questo trigger definisce una funzione "*almenoUnPubblicizza()*" che viene eseguita ogniqualevolta che si inserisce una riga nella tabella *CAMPAGNA_PUBBLICITARIA* o nella tabella *COLLEZIONE*, o dopo una cancellazione o aggiornamento di una riga della tabella *PUBBLICIZZA*.

I due blocchi *IF* presenti all'interno della funzione controllano rispettivamente che non ci siano righe della tabella *CAMPAGNA_PUBBLICITARIA* che non sono referenziate in *PUBBLICIZZA* e che non ci siano righe della tabella *COLLEZIONE* che non sono referenziate nella tabella *PUBBLICIZZA*.

In sintesi, controlla che sia rispettata la cardinalità minima (1,N) dell'associazione *PUBBLICIZZA* lato *CAMPAGNA_PUBBLICITARIA* e analogamente lato *COLLEZIONE*. Quindi una campagna deve pubblicizzare almeno una collezione, e una collezione deve essere pubblicizzata da almeno una campagna.

NB: nel caso si volesse per esempio cancellare una campagna (che per le politiche di cascade porterebbe all'eliminazione delle occorrenze di *PUBBLICIZZA* che la referenziano) il trigger potrebbe non permettere di cancellarla perchè poi la collezione associata potrebbe rimanere senza campagna. Per risolvere bisogna cancellare sia la campagna che la collezione associata con due *DELETE* all'interno di una transazione.

Per permettere l'inserimento o la cancellazione di occorrenze all'interno della stessa transazione in modo da soddisfare il trigger, questo deve essere invocato al termine della transazione e non all'esecuzione di ogni query. Per fare ciò il trigger va definito in fase di creazione come "*CONSTRAINT TRIGGER*" e va aggiunta la clausola "*DEFERRABLE INITIALLY DEFERRED*". PostgreSQL impone che i *CONSTRAINT TRIGGER* siano di tipo *AFTER* e *ROW LEVEL*, come in questo caso.

Nello script di creazione del database sono inclusi tutti i trigger che gestiscono tutte le cardinalità minime (1,N), che sono completamente analoghi al trigger appena mostrato (ne sono 4 senza contare il seguente).

Ricapitolando, il vincolo di cardinalità minima può essere violato in queste tre situazioni:

- inserimento nella tabella *CAMPAGNA_PUBBLICITARIA*
- inserimento nella tabella *COLLEZIONE*
- modifica o cancellazione nella tabella *PUBBLICIZZA*

Dato che un trigger può essere definito su un'unica tabella target, vanno creati tre trigger che eseguono la stessa trigger function.

Di seguito riportato il codice del trigger (solo del primo relativo alle cardinalità minime):

```
CREATE OR REPLACE FUNCTION almenoUnPubblicizza() RETURNS TRIGGER AS $$
BEGIN
IF (EXISTS (SELECT codice FROM campagna_pubblicitaria
            WHERE codice NOT IN (SELECT campagna FROM pubblicizza))) THEN
    RAISE EXCEPTION 'ERRORE CAMPAGNA (1,N) PUBBLICIZZA';
END IF;
IF (EXISTS (SELECT id FROM collezione
            WHERE id NOT IN (SELECT collezione FROM pubblicizza))) THEN
    RAISE EXCEPTION 'ERRORE COLLEZIONE (1,N) PUBBLICIZZA';
END IF;
RETURN NULL;
END $$ LANGUAGE plpgsql;

DROP TRIGGER IF EXISTS triggerAlmenoUnPubblicizza1 ON
campagna_pubblicitaria;
CREATE CONSTRAINT TRIGGER triggerAlmenoUnPubblicizza1
AFTER INSERT ON campagna_pubblicitaria
DEFERRABLE INITIALLY DEFERRED
FOR EACH ROW EXECUTE PROCEDURE almenoUnPubblicizza();

DROP TRIGGER IF EXISTS triggerAlmenoUnPubblicizza2 ON collezione;
CREATE CONSTRAINT TRIGGER triggerAlmenoUnPubblicizza2
AFTER INSERT ON collezione
DEFERRABLE INITIALLY DEFERRED
FOR EACH ROW EXECUTE PROCEDURE almenoUnPubblicizza();

DROP TRIGGER IF EXISTS triggerAlmenoUnPubblicizza ON pubblicizza;
CREATE CONSTRAINT TRIGGER triggerAlmenoUnPubblicizza
AFTER DELETE OR UPDATE ON pubblicizza
DEFERRABLE INITIALLY DEFERRED
FOR EACH ROW EXECUTE PROCEDURE almenoUnPubblicizza();
```


9.1.2. Trigger2: *aggiornaRidondanza()*

Questo trigger definisce una funzione “aggiornaRidondanza()” che viene invocata dopo l’inserimento, l’aggiornamento e la cancellazione della colonna NEWSLETTER nella tabella ISCRIZIONE. Il trigger viene eseguito per ogni riga interessata.

Anche in questo caso esso viene organizzato tramite due blocchi *IF*:

- Nel caso in cui la variabile *NEW* contiene un valore diverso da *NULL* (ossia in caso il trigger sia stato attivato da una *INSERT* o da una *UPDATE*, che forniscono una nuova riga), esegue un aggiornamento sulla tabella NEWSLETTER, impostando il valore della colonna “numero_iscritti” a 0 per tutte le righe in cui il valore della colonna “campagna” corrisponde al valore nella colonna “newsletter” della nuova riga;
- Nel caso in cui, invece, la variabile *OLD* contiene un valore diverso da *NULL* (ossia in caso il trigger sia stato attivato da una *DELETE* o da una *UPDATE*, che forniscono una vecchia riga) esegue un aggiornamento sulla tabella NEWSLETTER impostando il valore della colonna “numero_iscritti” a 0 per tutte le righe in cui il valore della colonna “campagna” corrisponde al valore nella colonna “newsletter” della vecchia riga.
- Si osserva che nel caso in cui il trigger è stato attivato da una *UPDATE*, vanno eseguiti entrambi i rami *IF* perchè va aggiornata la newsletter che ha perso un iscritto (la *OLD*) e la newsletter che ha guadagnato un iscritto (la *NEW*).

In sintesi, il trigger viene utilizzato per aggiornare il valore della colonna “numero_iscritti” nella tabella NEWSLETTER quando vengono effettuate modifiche nella colonna “newsletter” della tabella ISCRIZIONE.

Va specificato che la *UPDATE* che setta il numero di iscritti a zero serve semplicemente per invocare il trigger “proteggiRidondanza” (che verrà spiegato nel paragrafo successivo), dunque va bene qualunque tipo di modifica.

Di seguito riportato il codice del trigger:

```
CREATE OR REPLACE FUNCTION aggiornaRidondanza() RETURNS TRIGGER AS $$
BEGIN
  IF (NEW IS NOT NULL) THEN
    UPDATE newsletter SET numero_iscritti=0 WHERE
    campagna=NEW.newsletter;
  END IF;
  IF (OLD IS NOT NULL) THEN
    UPDATE newsletter SET numero_iscritti=0 WHERE
    campagna=OLD.newsletter;
  END IF;
  RETURN NULL;
END $$ LANGUAGE plpgsql;

DROP TRIGGER IF EXISTS TriggerAggiornaRidondanza ON newsletter;
CREATE TRIGGER TriggerAggiornaRidondanza
AFTER INSERT OR DELETE OR UPDATE OF newsletter ON iscrizione
FOR EACH ROW
EXECUTE PROCEDURE aggiornaRidondanza();
```

9.1.3. Trigger3: *proteggiRidondanza()*

Questo trigger definisce una funzione “*proteggiRidondanza()*” che viene invocata dopo l’inserimento o l’aggiornamento della colonna `NUMERO_ISCRITTI` nella tabella `NEWSLETTER`. Il trigger viene eseguito per ogni riga interessata.

All'interno della funzione, viene eseguita una query di selezione (`SELECT`) che conta il numero di righe nella tabella `ISCRIZIONE` in cui il valore della colonna “newsletter” corrisponde al valore della colonna “campagna” nella nuova riga. Il risultato di questa query viene assegnato alla colonna “numero_iscritti” nella nuova riga (keyword *INTO*). Infine la nostra funzione andrà a restituire questa nuova riga, che sostituirà la riga che inizialmente sarebbe stata inserita o aggiornata dalla query che ha attivato il trigger; per questo motivo, il trigger va necessariamente definito *BEFORE*.

In sintesi, il trigger viene utilizzato per proteggere la ridondanza del valore della colonna “numero_iscritti” nella tabella `NEWSLETTER`. Prima di effettuare l’inserimento o l’aggiornamento della colonna “numero_iscritti”, il trigger esegue una query per calcolare il numero di iscrizioni nella tabella `ISCRIZIONE` che corrispondono alla campagna specificata nella nuova riga. Questo valore viene quindi assegnato alla colonna “numero_iscritti” nella nuova riga.

Di seguito riportato il codice del trigger:

```
CREATE OR REPLACE FUNCTION proteggiRidondanza() RETURNS TRIGGER AS $$
BEGIN
SELECT COUNT(*) INTO NEW.numero_iscritti
FROM iscrizione WHERE newsletter = NEW.campagna;
RETURN NEW;
END $$ LANGUAGE plpgsql;

DROP TRIGGER IF EXISTS TriggerProteggiRidondanza ON newsletter;
CREATE TRIGGER TriggerProteggiRidondanza
BEFORE INSERT OR UPDATE OF numero_iscritti ON newsletter
FOR EACH ROW
EXECUTE PROCEDURE proteggiRidondanza();
```

9.2. Trigger per vincoli aziendali

Workpackage	Task	Responsabile
WP4	Trigger per vincoli aziendali	Ciaravola Giosuè

9.2.1. Trigger1: *vietaIscrizioniDisiscrizioni()*

Il seguente trigger fa riferimento al vincolo aziendale (**RV2**) relativo all'iscrizione o disiscrizione di un utente a una newsletter. Questo vincolo stabilisce che un utente non può né iscriversi né disiscriversi da una campagna già conclusa. Il trigger definisce una funzione chiamata "vietaIscrizioniDisiscrizioni()" che viene attivata nel momento in cui facciamo inserimenti ,modifiche o cancellazioni sulla tabella "ISCRIZIONE".

In sintesi, il trigger assicura che l'iscrizione o la disiscrizione da una campagna pubblicitaria sia consentita solo se la campagna non è ancora conclusa, altrimenti viene sollevata un'eccezione per impedire l'operazione che ha attivato il trigger.

NB: L'implementazione di questo trigger presuppone che una campagna venga inserita nel database come campagna in corso a cui poi vengono associati una newsletter con le sue iscrizioni e i valori di KPI corrispondenti, e solo in un momento successivo la campagna diventi conclusa con la modifica della data di fine.

Per la corretta esecuzione delle query, il popolamento inserisce sin dall'inizio delle campagne già concluse e le iscrizioni associate, che quindi attiverrebbero il trigger e non permetterebbero il popolamento. Per questo motivo, mentre tutti gli altri trigger possono essere creati dopo la creazione delle tabelle e prima del popolamento, questo è l'unico che va necessariamente creato dopo il popolamento.

Va osservato che a campagna conclusa le iscrizioni non sono gli unici elementi da rendere immutabili. Anche i dati relativi alla newsletter e valutazioni dei KPI associati andrebbero resi non modificabili tramite trigger analoghi.

Di seguito riportato lo script con il trigger:

```
CREATE OR REPLACE FUNCTION vietaIscrizioniDisiscrizioni() RETURNS TRIGGER
AS $$
BEGIN
IF (NEW IS NOT NULL) THEN
    IF ((SELECT CP.data_fine FROM campagna_pubblicitaria CP
        WHERE CP.codice = NEW.newsletter) IS NOT NULL) THEN
        RAISE EXCEPTION 'ERRORE ISCRIZIONE/DISISCRIZIONE A CAMPAGNA
CONCLUSA';
    END IF;
END IF;
IF (OLD IS NOT NULL) THEN
    IF ((SELECT CP.data_fine FROM campagna_pubblicitaria CP
        WHERE CP.codice = OLD.newsletter) IS NOT NULL) THEN
```

```
        RAISE EXCEPTION 'ERRORE ISCRIZIONE/DISISCRIZIONE A CAMPAGNA
CONCLUSA';
    END IF;
END IF;
RETURN NULL;
END $$ LANGUAGE plpgsql;

DROP TRIGGER IF EXISTS TriggerVietaIscrizioniDisiscrizioni ON iscrizione;
CREATE TRIGGER TriggerVietaIscrizioniDisiscrizioni
AFTER INSERT OR UPDATE OR DELETE ON iscrizione
FOR EACH ROW
EXECUTE PROCEDURE vietaIscrizioniDisiscrizioni();
```

9.2.2. Trigger2: *controlloCampagna()*

Il seguente trigger fa riferimento ai vincoli aziendali **(RV9)** e **(RV10)** che impongono alle campagne commissionate da Zalando di pubblicizzare una o più collezioni aventi lo stesso nome e lo stesso anno, mentre impone alle campagne non commissionate da Zalando di pubblicizzare una sola collezione in cui l'azienda fornitrice coincide con l'azienda committente.

Nel particolare:

1. Il primo IF verifica se l'associazione "pubblicizza" inserita/modificata (referenziata tramite la keyword NEW) fa riferimento ad una campagna commissionata da Zalando, verificandone la *partita IVA*
 - 1.1. Se il primo IF ha successo si passa ad un IF interno, il quale controlla se la nuova collezione associata alla campagna ha *nome* oppure *anno* diverso da almeno una di tutte le altre collezioni associate alla stessa campagna (e.g. Se una campagna con id=5 presenta una collezione che si chiama "Collezione Primavera" e ha anno "2023", tutte le altre collezioni appartenenti a quella campagna devono avere stesso nome ed anno). Se la condizione è soddisfatta viene sollevata un'eccezione con un messaggio di errore.
2. Il secondo IF verifica se l'associazione pubblicizza inserita/modificata (riferimento a NEW) fa riferimento a una campagna commissionata da un'azienda diversa da Zalando, verificandone la *partita IVA*. Se la condizione è verificata si passa a due IF in sequenza
 - 2.1. Viene controllato se la campagna è associata a più di una riga nella tabella "pubblicizza", cioè se risulta essere associata a più di una collezione. Se ciò è vero viene sollevata un'eccezione con un messaggio di errore.
 - 2.2. Successivamente, viene verificato che l'azienda committente della campagna e l'azienda fornitrice della collezione ad essa associata sono diverse, e in questo caso viene sollevata un'eccezione con un messaggio di errore.

NB: I vincoli aziendali **(RV9)** e **(RV10)** possono essere violati anche in altre due situazioni:

- Se viene modificato l'attributo azienda_committente di una campagna pubblicitaria
- Se vengono modificati almeno uno degli attributi azienda_fornitrice, nome, anno di una collezione

In entrambi i casi si tratta di una modifica di attributi costituenti di una campagna o di una collezione, che in linea di principio andrebbero resi non modificabili una volta avvenuto l'inserimento. Ciò può essere ottenuto modificando i permessi di modifica degli utenti che accedono al database, oppure tramite dei semplici trigger che sollevano un'eccezione quando viene tentata una modifica.

Di seguito riportato lo script con il trigger:

```
CREATE OR REPLACE FUNCTION controlloCampagna() RETURNS TRIGGER AS $$  
BEGIN  
IF (EXISTS (SELECT azienda_committente FROM campagna_pubblicitaria  
            WHERE codice = NEW.campagna  
            AND azienda_committente = '02986180210')) THEN  
    IF (EXISTS (SELECT C.nome, C.anno FROM publicizza AS P  
                JOIN collezione AS C ON P.collezione = C.id  
                WHERE NEW.campagna = P.campagna AND  
                (C.nome <>  
                 (SELECT nome FROM collezione WHERE id = NEW.collezione)  
                 OR C.anno <>  
                 (SELECT anno FROM collezione WHERE id = NEW.collezione)))) THEN  
        RAISE EXCEPTION 'ERRORE CAMPAGNE DI ZALANDO DEVONO  
        PUBBLICIZZARE CAMPAGNE CON NOME E ANNO UGUALI';  
    END IF;  
END IF;  
IF (EXISTS (SELECT azienda_committente FROM campagna_pubblicitaria  
            WHERE codice = NEW.campagna  
            AND azienda_committente <> '02986180210')) THEN  
    IF ((SELECT count(*) FROM publicizza  
        WHERE campagna = NEW.campagna) > 1) THEN  
        RAISE EXCEPTION 'ERRORE CAMPAGNE NON DI ZALANDO DEVONO  
        PUBBLICIZZARE UNA SOLA COLLEZIONE';  
    END IF;  
    IF ((SELECT azienda_committente FROM campagna_pubblicitaria  
        WHERE codice = NEW.campagna) <> (SELECT azienda_fornitrice  
        FROM collezione WHERE id = NEW.collezione)) THEN  
        RAISE EXCEPTION 'ERRORE CAMPAGNE NON DI ZALANDO DEVONO  
        PUBBLICIZZARE COLLEZIONI PROPRIE';  
    END IF;  
END IF;  
RETURN NULL;  
END $$ LANGUAGE plpgsql;
```



```
DROP TRIGGER IF EXISTS TriggerControlloCampagna1 ON publicizza;  
CREATE TRIGGER TriggerControlloCampagna1  
AFTER INSERT OR UPDATE ON publicizza  
FOR EACH ROW  
EXECUTE PROCEDURE controlloCampagna();
```