



Università degli Studi di Salerno



Dipartimento di Ingegneria dell'Informazione ed Elettrica e
Matematica Applicata

Corso di Laurea in Ingegneria Informatica

Basi di Dati 2021/2022 Canale I-Z

Project Work

Traccia N. 5 – Sentiment Analysis: la gestione delle Opinion

Gruppo n. **08 – IZ**

| WP | Cognome e Nome | Matricola | e-mail | Responsabile |
|----|----------------------|------------|-----------------------------------|--------------|
| 1 | Postiglione Matteo | 0612705214 | m.postiglione10@studenti.unisa.it | |
| 2 | Relmi Emanuele | 0612705274 | e.relmi@studenti.unisa.com | X |
| 3 | Raffone Riccardo | 0612705256 | r.raffone3@studenti.unisa.it | |
| 4 | Quagliuolo Francesco | 0612705239 | f.quagliuolo@studenti.unisa.it | |

Anno accademico 2021-2022

Sommario

| | |
|---|-----------|
| 1. Descrizione della realtà di interesse | 3 |
| 1.1. Analisi della realtà di interesse | 4 |
| 2. Analisi delle specifiche | 5 |
| 2.1. Glossario dei termini | 5 |
| 2.2. Strutturazione dei requisiti in frasi..... | 6 |
| 2.2.1. Frasi di carattere generale | 6 |
| 2.2.2. Frasi relative a Utente..... | 6 |
| 2.2.3. Frasi relative a Sito..... | 6 |
| 2.2.4. Frasi relative ad Articolo | 6 |
| 2.2.5. Frasi relative ai Cookie | 6 |
| 2.2.6. Frasi relative a Commento..... | 6 |
| 2.3. Identificazione delle operazioni principali | 7 |
| 3. Progettazione Concettuale..... | 8 |
| 3.1. Schema Concettuale..... | 8 |
| 3.1.1. Note sullo schema E-R | 9 |
| 3.1.2. Design Pattern | 9 |
| 3.2. Dizionario dei Dati | 9 |
| 3.3. Regole Aziendali | 12 |
| 4. Progettazione Logica | 13 |
| 4.1. Ristrutturazione Schema Concettuale | 13 |
| 4.1.1. Analisi delle Prestazioni | 13 |
| 4.2. Analisi delle ridondanze | 14 |
| 4.2.1. Analisi della ridondanza 1 | 14 |
| 4.3. Eliminazione delle generalizzazioni..... | 15 |
| 4.3.1. Generalizzazione <i>Utente</i> | 15 |
| 4.3.2. Generalizzazione <i>Cookie</i> | 16 |
| 4.4. Partizionamento/Accorpamento Entità e Associazioni | 16 |
| 4.5. Scelta degli identificatori principali..... | 16 |
| 4.6. Schema ristrutturato finale | 17 |
| 4.7. Schema logico..... | 17 |
| 5. Normalizzazione | 19 |
| 5.1. Normalizzazione di “Articolo” | 19 |
| 6. Script Creazione e Popolamento Database | 20 |
| 6.1. Script Creazione | 20 |
| 6.2. Script Popolamento | 25 |
| 7. Query SQL..... | 28 |

| | | |
|--------|--|----|
| 7.1. | Query con operatore di aggregazione e join: “Estrarre il numero di utenti registrati che accedono ad un sito con uno specifico indirizzo IP” | 28 |
| 7.2. | Query nidificata complessa: “Stampare i commenti con un ranking superiore alla media” | 28 |
| 7.3. | Query insiemistica: “Estrarre quanti utenti di sesso femminile hanno visitato un determinato sito con un'età superiore ai 16 anni” | 28 |
| 7.4. | Eventuali Altre query | 29 |
| 7.4.1. | “Estrarre i cookie di sessione che denotano l'utilizzo dello stesso Sistema Operativo da parte degli Utenti Cookieless” | 29 |
| 8. | Viste | 30 |
| 8.1. | Vista Lettura Registrata | 30 |
| 8.1.1. | Query con Vista: “Estrarre il titolo dell'articolo più letto” | 30 |
| 9. | Trigger | 30 |
| 9.1. | Trigger inizializzazione | 30 |
| 9.1.1. | Trigger1: Cardinalità minima Articolo-Sito..... | 30 |
| 9.1.2. | Trigger2: Controllo Utente Cookieless | 31 |
| 9.2. | Trigger per vincoli aziendali | 31 |
| 9.2.1. | Trigger1: Limite Commenti | 32 |
| 9.2.2. | Trigger2: Aggiorna Numero Commenti..... | 32 |
| 9.2.3. | Trigger3: Controllo Ranking | 33 |
| 9.2.4. | Trigger4: Storico Cookie di Sessione | 33 |

1. Descrizione della realtà di interesse

Titolo: **Profilazione degli utenti per la gestione delle pubblicità**

Si vuole realizzare una base di dati per una società che fornisce servizi pubblicitari, allo scopo di profilare gli utenti che visitano siti web a carattere giornalistico e di monitorare le loro interazioni con essi, al fine di raccogliere informazioni utili per la profilazione pubblicitaria.

In particolare, si vogliono rappresentare:

- i diversi **Utenti**, caratterizzati da un indirizzo IP.
 - Se gli utenti sono **Registrati**, vogliamo memorizzare anche il *Nome*, il *Cognome*, l'*Età*, l'*E-mail*, lo Username, il *Sesso* e il *Numero di Commenti*;
- i **Siti**, a cui gli utenti accedono, caratterizzati da un URL e dal *Nome del sito* scelto, inoltre memorizzano informazioni inerenti all'utente tramite i cookie;
- gli **Articoli** vengono identificati da un Link e un Titolo e contengono informazioni riguardo all'*Autore*, al *Corpo* e al *Timestamp di Creazione*, inoltre sono classificati tramite dei *Tag* e hanno un *Tempo di Lettura Stimato*;
- i **Cookie**, che sono in grado di tenere traccia dell'utente, sono caratterizzati da un ID univoco, da un *Nome*, una *Descrizione* e dalla *Data di Creazione*. I cookie possono essere specializzati in:
 - **Cookie di Sessione**, che memorizzano il *Sistema Operativo*, preferenze sulla *Lingua*, risoluzione dello *Schermo* e il *Browser* utilizzati dall'utente;
 - **Cookie Persistenti** che memorizzano preferenze sul *Tema* e *Geolocalizzazione* e sono caratterizzati da una *Data di Scadenza*;
- i **Commenti** rilasciati dagli utenti sono identificati dalla *Data di Scrittura*, da un *Ranking* e dal *Testo* del commento.

1.1. Analisi della realtà di interesse

L'obiettivo del progetto è realizzare un database in grado di profilare gli utenti che accedono ai vari siti, al fine di promuovere materiale pubblicitario mirato, basandosi sulle preferenze tracciate tramite la navigazione.

L'utente che visita il sito è identificato dall'indirizzo IP (IPv4) composto al più da 16 cifre. L'indirizzo IP è un identificatore univoco associato al dispositivo utilizzato dall'utente per accedere al sito web. Una volta collegato al sito, quest'ultimo verifica che l'utente sia registrato o meno e, in base alla situazione, assegna all'utente in questione uno o più cookie formati da un identificativo univoco. È inoltre possibile, per l'utente, leggere più articoli da diversi siti, ma ha anche la possibilità di leggere un articolo tramite un link diretto senza accedere al sito specifico.

I cookie per la profilazione dell'utente sono divisi in cookie di sessione e cookie persistenti. Riguardo ai primi, essi sono un tipo di cookie, la cui durata è limitata alla sessione di navigazione al termine della quale vengono eliminati. I cookie persistenti, invece, sono utilizzati dai siti web per memorizzare le informazioni sul dispositivo dell'utente per un periodo di tempo più lungo rispetto ai cookie accennati precedentemente, venendo salvati sul disco rigido del dispositivo dell'utente e rimanendovi immagazzinati anche dopo la chiusura del browser web.

Un utente che vuole registrarsi deve accettare obbligatoriamente i cookie persistenti, pena l'annullamento della registrazione. In caso di non registrazione, all'utente vengono assegnati solo i cookie di sessione.

Nel caso proposto, poniamo attenzione sulle interazioni che l'utente ha con gli articoli, andando a memorizzare uno o più eventuali commenti rilasciati.

Si specifica che un utente può rilasciare al più un commento per articolo solo se ha effettuato il login al sito.

Si precisa che nel database viene tralasciata l'analisi delle singole frasi che compongono il contenuto dei commenti rimanendo ad un livello più elevato (*document level*).

Nel caso in analisi, il visitatore del sito potrebbe non aver creato un account e potrebbe anche aver scelto di non accettare i cookie. Ciò significa che abbiamo solo alcune informazioni sull'utente, in quanto alcuni cookie di sessione vengono utilizzati per personalizzare l'esperienza di navigazione.

Tuttavia, questo tipo di cookie è automaticamente eliminato quando l'utente chiude il browser o termina la sessione di navigazione. Pertanto, non possiamo essere completamente certi dell'identità dell'utente che sceglie di non accettare i cookie, ma possiamo effettuarne una stima utilizzando le informazioni limitate disponibili.

2. Analisi delle specifiche

| Workpackage | Task | Responsabile |
|--------------------|--------------------------|---------------------|
| WP0 | Analisi delle specifiche | Intero Gruppo |

2.1. Glossario dei termini

| | Termine | Descrizione | Sinonimi | Collegamenti |
|----------|----------------|---|-----------------------|-----------------------------|
| 1 | Utente | Persona che utilizza un dispositivo per navigare sul sito e ricercare gli articoli | Visitatore | Sito, Articolo |
| 2 | Sito | Pagina web visitabile da un utente | - | Articolo, Utente, Cookie |
| 3 | Articolo | Articolo giornalistico leggibile da un utente | - | Sito, Commento |
| 4 | Commento | Commento effettuato da un utente inerente all'articolo o l'argomento da esso trattato | Opinione, Interazione | Utente registrato, Articolo |
| 5 | Cookie | Piccolo file di testo che viene inviato da un sito web al browser dell'utente e memorizzato sul dispositivo dell'utente | - | Sito |
| 6 | URL | L'URL (Uniform Resource Locator) è una stringa di caratteri che identifica univocamente un indirizzo di risorsa su Internet | Link | Sito, Articolo |
| 7 | Indirizzo IP | Un indirizzo IP (Internet Protocol address) è una serie univoca di numeri che identifica un dispositivo connesso a una rete informatica che utilizza il protocollo Internet (IP). | - | Utente |

Tabella 1. Glossario dei Termini

2.2. Strutturazione dei requisiti in frasi

2.2.1. Frasi di carattere generale

Si vuole realizzare una base di dati per una società che fornisce servizi pubblicitari, allo scopo di profilare gli utenti che visitano siti web a carattere giornalistico e di monitorare le loro interazioni con essi, al fine di raccogliere informazioni utili per la profilazione pubblicitaria.

2.2.2. Frasi relative a Utente

Si vogliono rappresentare i diversi **Utenti**, caratterizzati da un *indirizzo IP*. Se gli utenti sono **Registrati**, vogliamo memorizzare anche il *Nome*, il *Cognome*, l'*Età*, l'*E-mail*, lo *Username*, il *Sesso* e il *Numero di Commenti*;

2.2.3. Frasi relative a Sito

Il **Sito**, a cui gli utenti accedono, è caratterizzato da un *URL* e dal *Nome del Sito* scelto, e inoltre memorizza informazioni inerenti all'utente tramite i cookie.

2.2.4. Frasi relative ad Articolo

Gli **Articoli** vengono identificati da un *Link* e un *Titolo*, inoltre contengono informazioni riguardo all'*Autore*, al *Corpo* e al *Timestamp di Creazione*, sono classificati tramite dei *Tag* e hanno un *Tempo di Lettura Stimato*.

2.2.5. Frasi relative ai Cookie

I **Cookie**, che sono in grado di tenere traccia dell'utente, sono caratterizzati da un **ID** univoco, da un *Nome*, una *Descrizione* e dalla *Data di Creazione*. I cookie possono essere specializzati in: **Cookie di Sessione**, che memorizzano il *Sistema Operativo*, preferenze sulla *Lingua*, risoluzione dello *Schermo* e il *Browser* utilizzati dall'utente, e **Cookie Persistenti** che memorizzano preferenze sul *Tema* e *Geolocalizzazione* e sono caratterizzati da una *Data di Scadenza*.

2.2.6. Frasi relative a Commento

I **Commenti** rilasciati dagli utenti sono identificati dalla *Data di Scrittura*, da un *Ranking* e dal *Testo* del commento.

2.3. Identificazione delle operazioni principali

Operazione 1: registrazione di un nuovo utente su uno specifico sito (operazione effettuata 150 volte al giorno)

Operazione 2: inserimento di un commento specifico (operazione da effettuare 10 volte al giorno)

Operazione 3: controllo degli utenti che hanno accettato i cookie (operazione effettuata 10 volte al giorno)

Operazione 4: stampa il numero di commenti scritto da un utente registrato (operazione da effettuare 5 volte ogni giorno)

Operazione 5: controllo dell'articolo più letto dagli utenti (operazione effettuata 1 volta a settimana)

Operazione 6: controllo del sistema operativo utilizzato dagli utenti cookieless (operazione effettuata 1 volta ogni 2 settimane)

Operazione 7: conteggio dei tag che appaiono più frequentemente negli articoli letti (operazione da effettuare mensilmente)

3. Progettazione Concettuale

| Workpackage | Task | Responsabile |
|-------------|---------------------------|--------------------|
| WP1 | Progettazione Concettuale | Postiglione Matteo |

3.1. Schema Concettuale

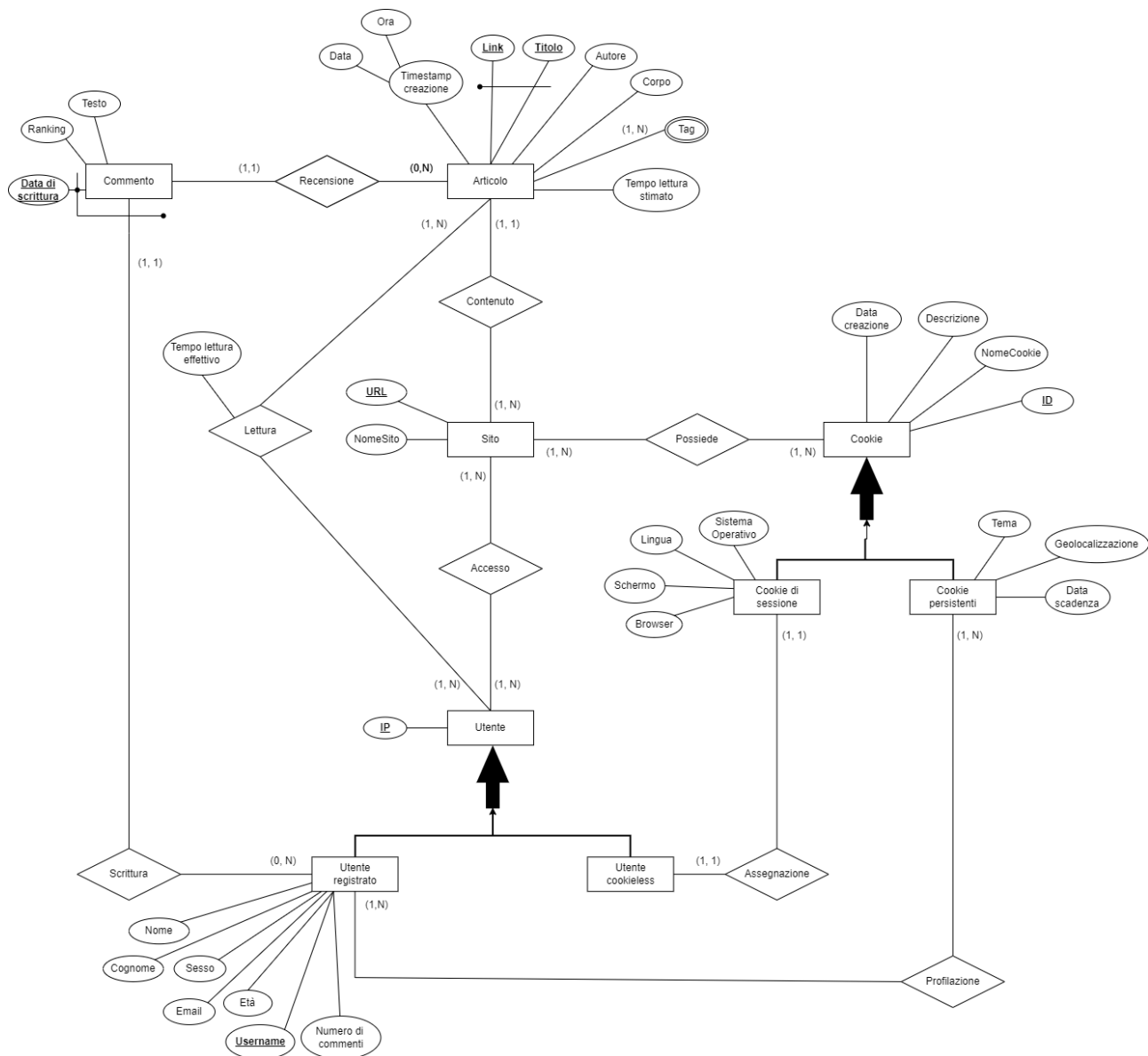


Figura 1. Schema E-R

3.1.1. Note sullo schema E-R

Per lo sviluppo dello schema concettuale si è fatto uso della strategia top-down partendo da uno schema iniziale che poi abbiamo rifinito mediante l'aggiunta degli attributi e le relative cardinalità.

3.1.2. Design Pattern

3.1.2.1. Pattern "Parte-di"

Nel contesto della progettazione del database, l'entità **Sito** rappresenta un sito web specifico, mentre l'entità **Articolo** rappresenta un articolo di giornale pubblicato su quel sito.

La relazione *parte-di* indica che ogni articolo è una parte integrante del sito su cui viene pubblicato.

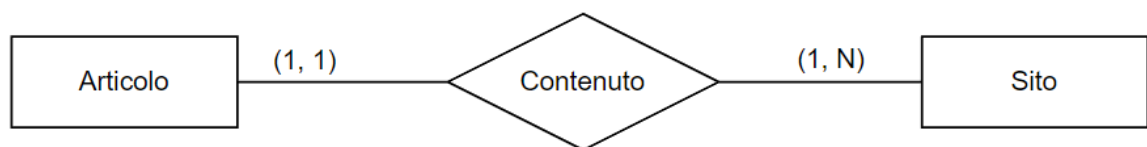


Figura 1. Schema successivo all'applicazione del Pattern PARTE DI

3.1.2.2. Pattern "Parte-di"

La relazione *parte-di* in questo caso indica che uno o più **cookie** appartengono ad uno o più **siti**.

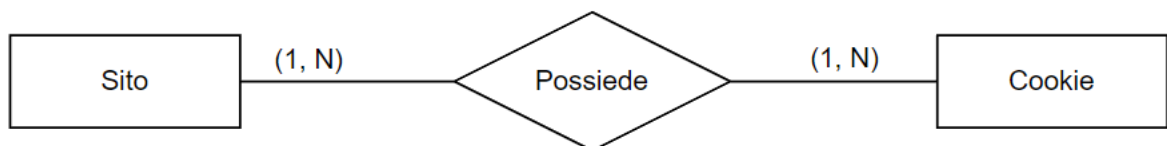


Figura 2. Schema successivo all'applicazione del Pattern PARTE DI

3.1.2.3. Pattern "Parte-di"

La relazione *parte-di* in questa casistica indica che uno o più **commenti** appartengono ad un solo **articolo**.

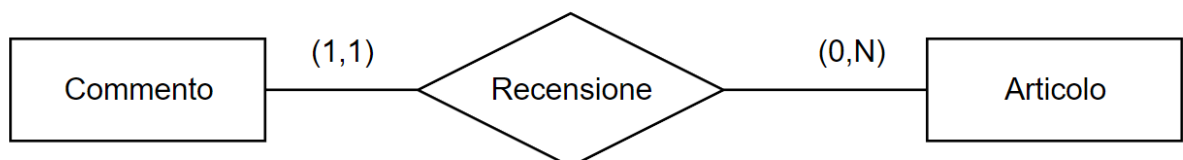


Figura 3. Schema successivo all'applicazione del Pattern PARTE DI

3.2. Dizionario dei Dati

| Entità | Descrizione | Attributi | Identificatore |
|--------|--|-----------|----------------|
| Utente | Persona che utilizza un dispositivo per navigare sul sito e ricercare gli articoli | IP | IP |

| | | | |
|--------------------|---|---|--|
| Utente registrato | Utente che naviga sul sito e/o legge articoli accettando i cookie persistenti | Nome, Cognome, Età, Sesso, E-mail, Username, Numero di commenti | Username |
| Utente cookieless | Utente che naviga sul sito e/o legge articoli profilata solo dai cookie di sessione | - | - |
| Sito | Pagina web visitabile da un utente | URL, Nome | URL |
| Articolo | Articolo giornalistico leggibile da un utente | Link, Titolo, Autore, Corpo, Tag, Tempo di lettura stimato, Timestamp Creazione | Link, Titolo |
| Cookie | Piccolo file di testo che viene inviato da un sito web al browser dell'utente e memorizzato sul dispositivo dell'utente | ID, Data di creazione, Nome, Descrizione | ID |
| Cookie di sessione | Cookie utilizzati per profilare gli utenti che declinano i cookie persistenti | Sistema Operativo, Lingua, Schermo, Browser | - |
| Cookie persistente | Cookie atti a profilare gli utenti registrati, immagazzinandone le informazioni personali | Tema, Geolocalizzazione, Data scadenza | - |
| Commento | Commento effettuato da un utente inerente all'articolo o l'argomento da esso trattato | Testo, Ranking, Data Scrittura | Data Scrittura, Username (Utente Registrato) |

Tabella 2. Dizionario dei dati – Entità

| Relazioni | Descrizione | Entità Coinvolte | Attributi |
|--------------|---|--------------------------------|-----------|
| Accesso | Associa un utente al sito che visita | Utente (1,N), Sito (1,N) | - |
| Possiede | Associa i cookie al sito | Sito (1,N), Cookie (1,N) | - |
| Contenuto | Associa gli articoli al sito visitato | Sito (1,N), Articolo (1, 1) | - |
| Assegnazione | Associa l'utente cookieless al cookie di sessione | Utente Cookieless (1,1), | - |

| | | | |
|--------------|--|--|----------------------------|
| | | Cookie di sessione (1,1) | |
| Profilazione | Associa l'utente registrato al cookie permanente | Utente Registrato (1,N), Cookie Persistente (1,N) | - |
| Scrittura | Associa il commento all'utente registrato | Commento (1,1), Utente Registrato (0,N) | - |
| Recensione | Associa il commento all'articolo | Commento (1,1), Articolo (0, N) | - |
| Lettura | Associa ogni utente agli articoli | Utente (1,N), Articolo (1,N) | Tempo di Lettura Effettivo |

Tabella 3. Dizionario dei dati - Relazioni

| Workpackage | Task | Responsabile |
|--------------------|------------------|----------------------|
| WP4 | Regole Aziendali | Quagliuolo Francesco |

3.3. Regole Aziendali

| Regole di Vincolo |
|--|
| <p>(RV1) Un Utente Cookieless non può rilasciare un commento all'articolo letto.</p> <p>(RV2) Un Utente che vuole registrarsi deve accettare i cookie persistenti.</p> <p>(RV3) Un Utente Registrato non può rilasciare più di un commento ad un articolo.</p> <p>(RV4) Il <i>Ranking</i> di un commento deve essere compreso tra 1 e 5.</p> <p>(RV5) In caso di sessione terminata da parte dell'utente cookieless con conseguente eliminazione degli stessi, essi vengono salvati in una tabella di archiviazione.</p> |

Tabella 4. Regole di vincolo

| Regole di derivazione |
|---|
| <p>(RD1) Il <i>Numero Commenti</i> di un Utente Registrato viene derivato ad ogni inserimento/eliminazione nella relazione Commento.</p> <p>(RD2) Il <i>Tempo Lettura Stimato</i> di un Articolo è calcolato come la media dei vari <i>Tempo Lettura Effettivo</i> presenti nell'associazione <u>Letture</u>.</p> |

Tabella 5. Regole di derivazione

4. Progettazione Logica

| Workpackage | Task | Responsabile |
|--------------------|----------------------|---------------------|
| WP2 | Progettazione Logica | Relmi Emanuele |

4.1. Ristrutturazione Schema Concettuale

4.1.1. Analisi delle Prestazioni

4.1.1.1. Tavola dei volumi

| Concetto | Tipo | Volume |
|--------------------|-------------|---------------|
| Utente | E | 1000 |
| Sito | E | 5 |
| Articolo | E | 5000 |
| Commento | E | 8000 |
| Utente registrato | E | 800 |
| Utente cookieless | E | 200 |
| Cookie | E | 25 |
| Cookie di sessione | E | 5 |
| Cookie persistente | E | 20 |
| --- | - | --- |
| Accesso | R | 5000 |
| Contenuto | R | 1000 |
| Recensione | R | 4000 |
| Scrittura | R | 600 |
| Lettura | R | 4000 |
| Assegnazione | R | 200 |
| Profilazione | R | 3500 |

Tabella 6. Tavola dei volumi

- *Ipotesi 1*: in media un utente scrive 10 commenti $\rightarrow 10 * 800 = 8000$ commenti;
- *Ipotesi 2*: in media un sito ha 4 cookie persistenti e 1 cookie di sessione $\rightarrow 5$ cookie per sito $\rightarrow 5 * 5 = 25$ cookie totali
- *Ipotesi 3*: in media un utente legge 5 articoli al giorno $\rightarrow 1000 * 5 = 5000$ articoli totali

4.1.1.2. Tavola delle operazioni

| Operazione | Tipo | Frequenza |
|--|-------------|------------------|
| Operazione 1 : registrazione di un nuovo utente su uno specifico sito | I | 150 al giorno |
| Operazione 2 : inserimento di un commento specifico | I | 10 al giorno |
| Operazione 3 : controllo degli utenti che hanno accettato i cookie | I | 10 al giorno |
| Operazione 4 : stampa il numero di commenti scritto da un utente registrato | I | 5 al giorno |

| | | |
|--|---|--------------------|
| Operazione 5: controllo dell'articolo più letto dagli utenti | B | 1 a settimana |
| Operazione 6: controllo del sistema operativo utilizzato dagli utenti cookieless | B | 1 ogni 2 settimane |
| Operazione 7: conteggio dei tag che appaiono più frequentemente negli articoli letti | B | 1 al mese |
| Operazione 8: calcola il tempo di lettura stimato di un articolo tramite la media dei tempi di lettura effettivi sullo stesso | I | 2 al giorno |

Tabella 7. Tavola delle operazioni

4.2. Analisi delle ridondanze

- **Ridondanza 1:** attributo *Numero di Commenti* dell'entità **Utente Registrato**.
TIPO: *Attributo Derivabile* da conteggio di occorrenze.
L'attributo *Numero di Commenti* può essere derivato contando le occorrenze dell'associazione Scrittura.
- **Ridondanza 2:** attributo *Tempo Lettura Stimato* dell'entità **Articolo**.
TIPO: *Attributo derivabile* da attributo di associazione.
Il *Tempo Lettura Stimato* è possibile derivarlo tramite l'attributo *Tempo Lettura Effettivo* dell'associazione Lettura Registrata.

4.2.1. Analisi della ridondanza 1

- **Operazione 4: Stampa il numero di commenti scritto da un utente registrato**

Con Ridondanza

| CONCETTO | COSTRUTTO | ACCESSI | TIPO |
|-------------------|-----------|---------|------|
| Utente Registrato | E | 1 | L |

Costo = Costo complessivo = $1L = 1$ accesso

Costo complessivo = $5 * \text{Costo} = 5$ accessi

Senza Ridondanza

| CONCETTO | COSTRUTTO | ACCESSI | TIPO |
|-------------------|-----------|---------|------|
| Utente Registrato | E | 1 | L |
| Scrittura | R | 10 | L |

Costo = $1L + 10L = 11$ accessi

Costo complessivo = $5 * \text{Costo} = 55$ accessi

- **Operazione 8: Calcola il tempo di lettura stimato di un articolo tramite la media dei tempi di lettura effettivi sullo stesso**

Con Ridondanza

| CONCETTO | COSTRUTTO | ACCESSI | TIPO |
|----------|-----------|---------|------|
| Utente | E | 1 | L |
| Lettura | R | 1 | L |
| Articolo | E | 1 | L |
| Articolo | E | 1 | S |

Costo = $1L + 1L + 1L + 1S^* = 5$ accessi

* $1S = 2L$

Costo complessivo = $2 * \text{Costo} = 10$ accessi

Senza Ridondanza

| CONCETTO | COSTRUTTO | ACCESSI | TIPO |
|----------|-----------|---------|------|
| Utente | E | 1 | L |
| Lettura | R | 4 | L |

Costo = $1L + 4L = 5$ accessi

Costo complessivo = $2 * \text{Costo} = 10$ accessi

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 15 accessi giornalieri
- L'occupazione di memoria è di 24000 byte, calcolato come segue:
 - o 4byte per ogni **Utente**, per via dell'attributo ridondante *Numero Commenti* (INTEGER), dunque $4 \times 1000 = 4000$ byte;
 - o 4byte per ogni **Articolo**, per via dell'attributo ridondante *Tempo Lettura Stimato* (INTEGER), dunque $4 \times 5000 = 20000$ byte;
- In assenza di ridondanza il costo delle operazioni è di 75 accessi giornalieri

Pertanto, si decide di mantenere la ridondanza in quanto riduce il numero di accessi, a fronte di un'occupazione di memoria pari a 24kB, i quali risultano essere marginali considerando le risorse disponibili oggi giorno.

4.3. Eliminazione delle generalizzazioni

Nello schema concettuale sono presenti due gerarchie: quella relativa a **Utente** e quella relativa a **Cookie**.

4.3.1. Generalizzazione Utente

Per gli utenti si può notare che le entità figlie, **Utente Registrato** ed **Utente Cookieless**, hanno attributi specifici che li distinguono.

Essendo la generalizzazione totale procediamo ad accorpare l'entità padre all'interno delle figlie.

4.3.2. Generalizzazione *Cookie*

Per i cookie notiamo che le entità figlie, **Cookie di Sessione** e **Cookie Persistenti**, hanno attributi specifici che le distinguono. Anche in questo caso, essendo la generalizzazione totale, procediamo ad accorpare l'entità padre all'interno delle figlie.

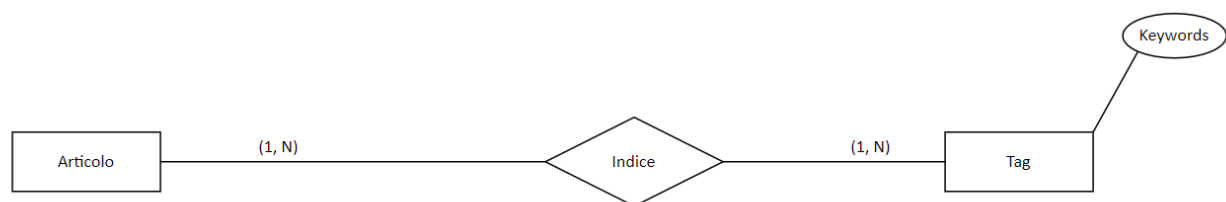
4.4. Partizionamento/Accorpamento Entità e Associazioni

Un possibile accorpamento nel nostro schema può essere effettuato tra le entità **Utente Cookieless** e **Cookie Di Sessione**. Si può osservare che l'operazione 6 fa riferimento al *Sistema Operativo* utilizzato dagli **Utenti Cookieless**. Si potrebbe pensare quindi di accorpare le due entità.

Tuttavia, anche se andassimo ad accorparle, volendo conoscere solamente il *Sistema Operativo*, ci basterebbe accedere una sola volta all'entità *Cookie Di Sessione* senza dover passare per l'utente. Decidiamo quindi di non effettuare tale accorpamento.

Per poter eliminare l'attributo multivalore *Tag* associato all'entità **Articolo**, introduciamo una nuova entità **Tag** legata tramite un'associazione molti a molti con l'entità **Articolo**, che viene privata del relativo attributo.

Di seguito mostriamo l'attributo modificato.



4.5. Scelta degli identificatori principali

Solo l'entità **Utente Registrato** presenta due identificatori: l'*indirizzo IP* e lo *Username*. Tra i due nel nostro caso è preferibile scegliere il secondo, per motivi di accessibilità.

Infatti, poiché l'**Utente Registrato** può scrivere un commento ed essendo quest'ultimo collegato al primo, risulta essere più identificativo l'utilizzo dello *Username* al fine della profilazione.

4.6. Schema ristrutturato finale

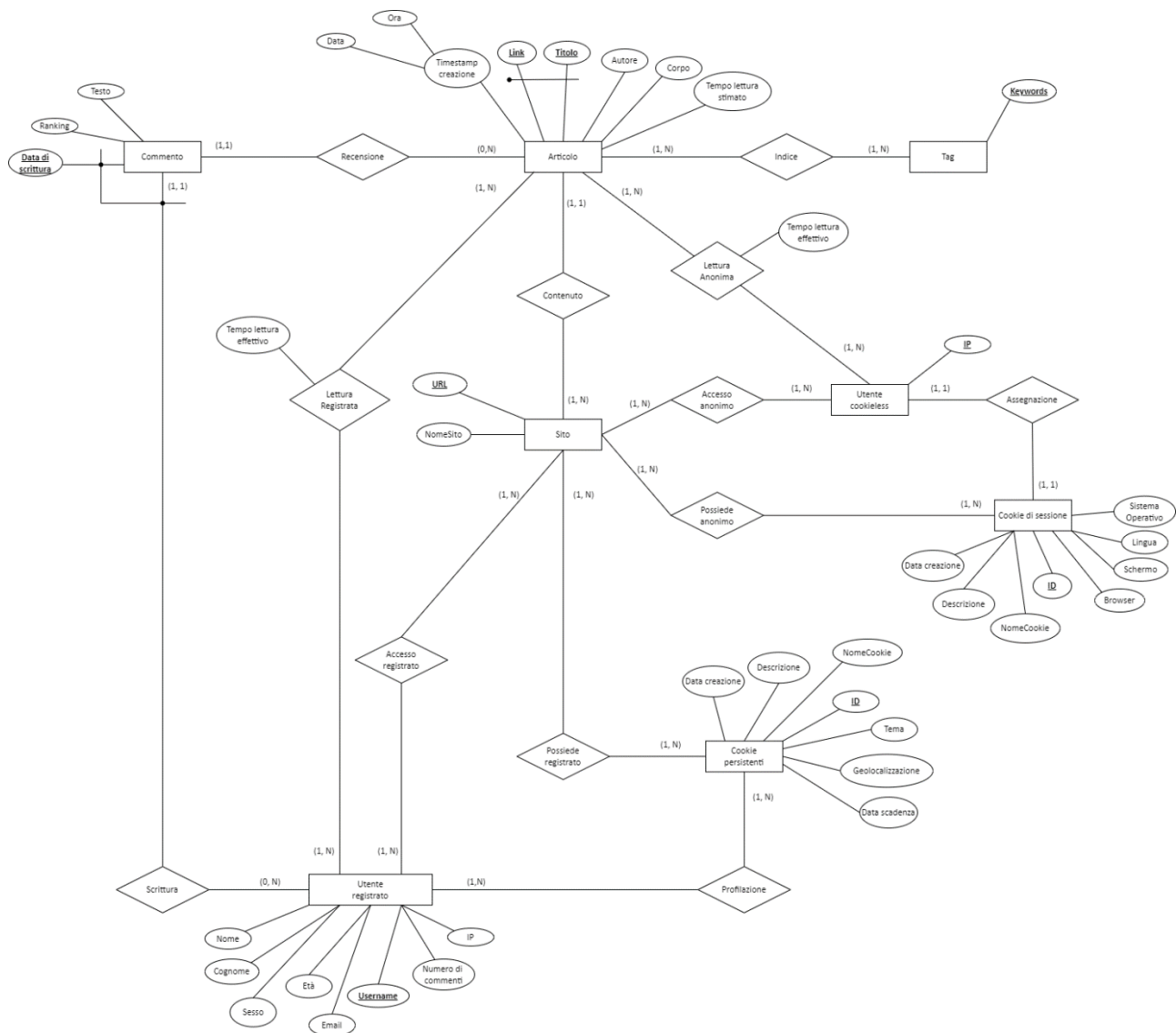


Figura 3. Schema ER Ristrutturato

4.7. Schema logico

Utente Registrato (Username, Nome, Cognome, Età, Sesso, E-mail, IP, Numero Commenti)

Profilazione (Utente Registrato, Cookie Persistenti)

Accesso Registrato (Sito, Utente Registrato)

Lettura Registrata (Utente Registrato, Link Articolo, Titolo Articolo, Tempo Lettura Effettivo)

Utente Cookieless (IP, ID Cookie)

Lettura Anonima (Link Articolo, Titolo Articolo, Utente Cookieless, Tempo Lettura Effettivo)

Accesso Anonimo (Sito, Utente Cookieless)

Cookie Persistenti (ID, Nome Cookie, Descrizione, Geolocalizzazione, Tema, Data Creazione, Data Scadenza)

Possiede Registrato (Sito, Cookie Persistente)

Cookie Sessione (ID, Nome Cookie, Descrizione, Sistema Operativo, Lingua, Schermo, Browser, Data Creazione)

Possiede Anonimo (Cookie Sessione, Sito)

Sito (URL, Nome Sito)

Articolo (Link, Titolo, Corpo, Autore, Data Creazione, Ora Creazione, Tempo Lettura Stimato, URL Sito)

Commento (Data Scrittura, Username Utente, Ranking, Testo, Link Articolo, Titolo Articolo)

Tag (Keywords)

Indice (Link Articolo, Titolo Articolo, Tag)

5. Normalizzazione

| Workpackage | Task | Responsabile |
|--------------------|-----------------|---------------------|
| WP3 | Normalizzazione | Raffone Riccardo |

| Relazione | FN | Tabelle |
|--------------------|-----------------------|---|
| Utente Registrato | 1-2-3 | R1(<u>Username</u> , Nome, Cognome, Età, Sesso, E-mail, Numero Commenti, IP) |
| Sito | 1-2-3 Boyce & Codd | R1(<u>URL</u> , Nome Sito) |
| Articolo | 1-2-3 | R1(<u>Titolo</u> , <u>Link</u> , Autore, Corpo) T2(<u>Autore</u> , Data Creazione, Ora Creazione, Tempo Lettura Stimata) |
| Tag | 1-2-3 Boyce & Codd | R1(<u>Keywords</u>) |
| Commento | 1-2-3 Boyce & Codd | R1(<u>Data Scrittura</u> , <u>Username</u> , Ranking, Testo) |
| Utente Cookieless | 1-2-3 Boyce & Codd | R1(<u>IP</u>) |
| Cookie di Sessione | 1-2-3 Boyce & Codd | R1(<u>ID</u> , Nome Cookie, Descrizione, Data Creazione, Sistema Operativo, Lingua, Schermo, Browser) |
| Cookie Persistenti | 1-2-3 Boyce & Codd | R2(<u>ID</u> , Nome Cookie, Descrizione, Data Creazione, Data Scadenza, Tema, Geolocalizzazione) |

Per quanto concerne la normalizzazione dello schema relativo alla progettazione logica precedentemente effettuata, abbiamo avuto modo di denotare come, ad eccezione della relazione **Articolo**, l'intero schema fosse già adattato in *terza forma normale (3FN)*.

5.1. Normalizzazione di “Articolo”

Circa **Articolo**, abbiamo innanzitutto verificato la presenza della *prima e seconda forma normale (1FN e 2FN)*.

| Relazione | FN | Tabelle |
|------------------|-----------|---|
| Articolo | 1 | R1(<u>Titolo</u> , <u>Link</u> , Autore, Corpo, Data Creazione, Ora Creazione, Tempo Lettura Stimata) (A1, A2) → A3, A4, A5, A6, A7 |
| Articolo | 2 | R1(<u>Titolo</u> , <u>Link</u> , Autore, Corpo) R2(<u>Autore</u> , Data Creazione, Ora Creazione, Tempo Lettura Stimata) (A1, A2) → A3, A4 A3 → A5, A6, A7 |

Appurato ciò, abbiamo impostato l'attributo Autore come chiave di una seconda relazione, andando a decomporre **Articolo** in due relazioni, senza perdita di informazioni o eventuali anomalie, rispettando i requisiti della *3FN*.

| Relazione | FN | Tabelle |
|------------------|-----------|---|
| Articolo | 1-2-3 | R1(<u>Titolo</u> , <u>Link</u> , Autore, Corpo) T2(<u>Autore</u> , Data Creazione, Ora Creazione, Tempo Lettura Stimata) |

6. Script Creazione e Popolamento Database

| Workpackage | Task | Responsabile |
|--------------------|-------------------------------------|---------------------|
| WP2 | SQL: Script creazione e popolamento | Relmi Emanuele |

6.1. Script Creazione

```
--Creazione del database
CREATE DATABASE "Cookies Profilation"
WITH
    OWNER = postgres
    ENCODING = 'UTF8'
    LC_COLLATE = 'en_GB.UTF-8'
    LC_CTYPE = 'en_GB.UTF-8'
    TABLESPACE = pg_default
    CONNECTION LIMIT = -1
    IS_TEMPLATE = False;

-- Creazione delle relazioni per le entità

DROP TABLE IF EXISTS "Sito";
CREATE TABLE "Sito" (
    "URL" VARCHAR NOT NULL,
    "Nome Sito" VARCHAR NOT NULL,
    PRIMARY KEY ("URL")
);

DROP TABLE IF EXISTS "Articolo";
CREATE TABLE "Articolo" (
    "Link" VARCHAR NOT NULL,
    "Titolo" VARCHAR NOT NULL,
    "Corpo" VARCHAR NOT NULL,
    "Autore" VARCHAR NOT NULL,
    "Data Creazione" DATE NOT NULL,
    "Ora Creazione" TIME NOT NULL,
    "Tempo Lettura Stimato" INTEGER NOT NULL,
    "URL Sito" VARCHAR NOT NULL,
    PRIMARY KEY ("Link", "Titolo"),
    FOREIGN KEY ("URL Sito") REFERENCES "Sito"("URL") MATCH SIMPLE
    ON UPDATE CASCADE
    ON DELETE SET NULL
```

```
);

DROP TABLE IF EXISTS "Cookie Persistenti";
CREATE TABLE "Cookie Persistenti" (
    "ID" INTEGER PRIMARY KEY,
    "Nome Cookie" VARCHAR NOT NULL,
    "Descrizione" VARCHAR NOT NULL,
    "Geolocalizzazione" VARCHAR NOT NULL,
    "Tema" VARCHAR NOT NULL,
    "Data Creazione" DATE NOT NULL,
    "Data Scadenza" DATE NOT NULL
);

DROP TABLE IF EXISTS "Cookie Sessione";
CREATE TABLE "Cookie Sessione" (
    "ID" INTEGER PRIMARY KEY,
    "Nome Cookie" VARCHAR(20) NOT NULL,
    "Descrizione" VARCHAR NOT NULL,
    "Sistema Operativo" VARCHAR NOT NULL,
    "Lingua" VARCHAR NOT NULL,
    "Schermo" VARCHAR NOT NULL,
    "Browser" VARCHAR NOT NULL,
    "Data Creazione" DATE NOT NULL
);

DROP TABLE IF EXISTS "Utente Registrato";
CREATE TABLE "Utente Registrato" (
    "Username" VARCHAR(30) PRIMARY KEY,
    "Nome" VARCHAR(30) NOT NULL,
    "Cognome" VARCHAR(30) NOT NULL,
    "Età" INTEGER NOT NULL,
    "Sesso" CHAR NOT NULL,
    "Email" VARCHAR(40) NOT NULL,
    "IP" VARCHAR NOT NULL,
    "Numero Commenti" INTEGER NOT NULL
);

DROP TABLE IF EXISTS "Utente Cookieless";
CREATE TABLE "Utente Cookieless" (
    "IP" VARCHAR PRIMARY KEY,
    "ID Cookie" INTEGER NOT NULL,
    FOREIGN KEY ("ID Cookie") REFERENCES "Cookie Sessione"("ID") MATCH SIMPLE
    ON UPDATE CASCADE
    ON DELETE SET NULL
);

DROP TABLE IF EXISTS "Tag";
```

```

CREATE TABLE "Tag" (
    "Keywords" VARCHAR(15) NOT NULL,
    PRIMARY KEY ("Keywords")
);

DROP TABLE IF EXISTS "Commento";
CREATE TABLE "Commento" (
    "Data Scrittura" DATE,
    "Username Utente" VARCHAR NOT NULL,
    "Ranking" INTEGER NOT NULL,
    "Testo" VARCHAR(250) NOT NULL,
    "Link Articolo" VARCHAR NOT NULL,
    "Titolo Articolo" VARCHAR NOT NULL,
    PRIMARY KEY("Data Scrittura", "Username Utente"),
    FOREIGN KEY ("Username Utente") REFERENCES "Utente Registrato"("Username")
MATCH SIMPLE
    ON UPDATE CASCADE
    ON DELETE SET NULL,
    FOREIGN KEY ("Link Articolo", "Titolo Articolo") REFERENCES
"Articolo"("Link", "Titolo") MATCH SIMPLE
    ON UPDATE CASCADE
    ON DELETE SET NULL
);

-- Creazione delle relazioni per le associazioni

DROP TABLE IF EXISTS "Profilazione";
CREATE TABLE "Profilazione" (
    "Utente Registrato" VARCHAR NOT NULL,
    "Cookie Persistenti" INTEGER NOT NULL,
    PRIMARY KEY("Utente Registrato", "Cookie Persistenti"),
    FOREIGN KEY ("Utente Registrato") REFERENCES "Utente
Registrato"("Username") MATCH SIMPLE
    ON UPDATE CASCADE
    ON DELETE SET NULL,
    FOREIGN KEY ("Cookie Persistenti") REFERENCES "Cookie Persistenti"("ID")
MATCH SIMPLE
    ON UPDATE CASCADE
    ON DELETE SET NULL
);

DROP TABLE IF EXISTS "Accesso Registrato";
CREATE TABLE "Accesso Registrato" (
    "Sito" VARCHAR NOT NULL,
    "Utente Registrato" VARCHAR NOT NULL,
    PRIMARY KEY("Sito", "Utente Registrato"),

```

```

    FOREIGN KEY ("Utente Registrato") REFERENCES "Utente
Registrato"("Username") MATCH SIMPLE
    ON UPDATE CASCADE
    ON DELETE SET NULL,
    FOREIGN KEY ("Sito") REFERENCES "Sito"("URL") MATCH SIMPLE
    ON UPDATE CASCADE
    ON DELETE SET NULL
);

DROP TABLE IF EXISTS "Lettura Registrata";
CREATE TABLE "Lettura Registrata" (
    "Link Articolo" VARCHAR NOT NULL,
    "Titolo Articolo" VARCHAR NOT NULL,
    "Utente Registrato" VARCHAR NOT NULL,
    "Tempo Lettura Effettivo" INTEGER NOT NULL,
    PRIMARY KEY("Link Articolo", "Titolo Articolo", "Utente Registrato"),
    FOREIGN KEY ("Utente Registrato") REFERENCES "Utente
Registrato"("Username") MATCH SIMPLE
    ON UPDATE CASCADE
    ON DELETE SET NULL,
    FOREIGN KEY ("Link Articolo", "Titolo Articolo") REFERENCES
"Articolo"("Link", "Titolo") MATCH SIMPLE
    ON UPDATE CASCADE
    ON DELETE SET NULL
);

DROP TABLE IF EXISTS "Lettura Anonima";
CREATE TABLE "Lettura Anonima" (
    "Link Articolo" VARCHAR NOT NULL,
    "Titolo Articolo" VARCHAR NOT NULL,
    "Utente Cookieless" VARCHAR NOT NULL,
    "Tempo Lettura Effettivo" INTEGER NOT NULL,
    PRIMARY KEY("Link Articolo", "Titolo Articolo", "Utente Cookieless"),
    FOREIGN KEY ("Utente Cookieless") REFERENCES "Utente Cookieless"("IP")
MATCH SIMPLE
    ON UPDATE CASCADE
    ON DELETE SET NULL,
    FOREIGN KEY ("Link Articolo", "Titolo Articolo") REFERENCES
"Articolo"("Link", "Titolo") MATCH SIMPLE
    ON UPDATE CASCADE
    ON DELETE SET NULL
);

DROP TABLE IF EXISTS "Accesso Anonimo";
CREATE TABLE "Accesso Anonimo" (
    "Sito" VARCHAR NOT NULL,
    "Utente Cookieless" VARCHAR NOT NULL,

```



```

    PRIMARY KEY("Sito", "Utente Cookieless"),
    FOREIGN KEY ("Utente Cookieless") REFERENCES "Utente Cookieless"("IP")
MATCH SIMPLE
    ON UPDATE CASCADE
    ON DELETE SET NULL,
    FOREIGN KEY ("Sito") REFERENCES "Sito"("URL") MATCH SIMPLE
    ON UPDATE CASCADE
    ON DELETE SET NULL
);

DROP TABLE IF EXISTS "Possiede Registrato";
CREATE TABLE "Possiede Registrato" (
    "Sito" VARCHAR NOT NULL,
    "Cookie Persistenti" INTEGER NOT NULL,
    PRIMARY KEY("Sito", "Cookie Persistenti"),
    FOREIGN KEY ("Sito") REFERENCES "Sito"("URL") MATCH SIMPLE
    ON UPDATE CASCADE
    ON DELETE SET NULL,
    FOREIGN KEY ("Cookie Persistenti") REFERENCES "Cookie Persistenti"("ID")
MATCH SIMPLE
    ON UPDATE CASCADE
    ON DELETE SET NULL
);

DROP TABLE IF EXISTS "Possiede Anonimo";
CREATE TABLE "Possiede Anonimo" (
    "Cookie Sessione" INTEGER NOT NULL,
    "Sito" VARCHAR NOT NULL,
    PRIMARY KEY("Sito", "Cookie Sessione"),
    FOREIGN KEY ("Sito") REFERENCES "Sito"("URL") MATCH SIMPLE
    ON UPDATE CASCADE
    ON DELETE SET NULL,
    FOREIGN KEY ("Cookie Sessione") REFERENCES "Cookie Sessione"("ID") MATCH
SIMPLE
    ON UPDATE CASCADE
    ON DELETE SET NULL
);

DROP TABLE IF EXISTS "Indice";
CREATE TABLE "Indice" (
    "Link Articolo" VARCHAR NOT NULL,
    "Titolo Articolo" VARCHAR NOT NULL,
    "Tag" VARCHAR NOT NULL,
    PRIMARY KEY("Link Articolo", "Titolo Articolo", "Tag"),
    FOREIGN KEY ("Tag") REFERENCES "Tag"("Keywords") MATCH SIMPLE
    ON UPDATE CASCADE
    ON DELETE SET NULL,

```

```

    FOREIGN KEY ("Link Articolo", "Titolo Articolo") REFERENCES
"Articolo"("Link", "Titolo") MATCH SIMPLE
    ON UPDATE CASCADE
    ON DELETE SET NULL
);

```

6.2. Script Popolamento

Per quanto concerne lo script di popolamento, abbiamo deciso di inserire soltanto una tupla per ogni relazione presente nello script.

Il file allegato conterrà invece lo script integrale, completo di ogni inserimento.

```

-- Popolamento Sito

INSERT INTO "Sito" ("URL", "Nome Sito")
VALUES
    ('https://repubblica.it', 'La Repubblica');

-- Popolamento Articolo

INSERT INTO "Articolo" ("Link", "Titolo", "Corpo", "Autore", "Data Creazione",
"Ora Creazione", "Tempo Lettura Stimato", "URL Sito")
VALUES
    -- Libero
    ('https://libero.it/bimba-di-un-anno-trovata-morta', 'Bimba di un anno
trovata morta', 'Una bambina di un anno è stata trovata morta in un''auto
parcheggiata in via dei Fucilieri, a Roma in zona Cecchignola. Stella, così si
chiamava la bimba, è stata lasciata in macchina dal padre, carabiniere in
servizio nella struttura dello Stato Maggiore della Difesa che si trova nella
zona. Doveva lasciarla all''asilo nido dei figli dei dipendenti. Quando la
madre è andata a riprenderla, ha scoperto che la piccola non era mai entrata.
Le maestre le hanno detto che non era mai stata portata. È corsa verso la
macchina, ha cominciato a urlare. Un passante ha rotto il finestrino per
cercare di farla respirare. Il personale del 118, arrivato sul posto, ha
cercato di rianimare la bambina ma era troppo tardi.', 'Ubaldo Argenio', '2023-
06-01', '12:30:00', 10, 'https://libero.it');

-- Popolamento Tag

INSERT INTO "Tag" ("Keywords")
VALUES
    ('guerra');

-- Popolamento Indice

INSERT INTO "Indice" ("Link Articolo", "Titolo Articolo", "Tag")
VALUES

```

```
        ('https://tg24.sky.it/russia-ucraina-guerra-ultime-notizie', 'Guerra
Ucraina Russia, diga distrutta', 'Kiev');

-- Popolamento Cookie Persistenti

INSERT INTO "Cookie Persistenti" ("ID", "Nome Cookie", "Descrizione",
"Geolocalizzazione", "Tema", "Data Creazione", "Data Scadenza")
VALUES
    ('0612105243', 'GYGIA', 'Tecnico usato per autenticazione utente',
'Roma', 'Chiaro', '2023-06-08', '2024-06-08');

-- Popolamento Cookie Sessione

INSERT INTO "Cookie Sessione" ("ID", "Nome Cookie", "Descrizione", "Sistema
Operativo", "Lingua", "Schermo", "Browser", "Data Creazione")
VALUES
    ('168615442', 'session_id', 'Identificatore univoco assegnato alla
sessione dell''utente', 'Windows-10Home', 'Italiano', '1280x800', 'Mozilla-
Firefox', '2019-03-27');

-- Popolamento Utente Registrato

INSERT INTO "Utente Registrato" ("Username", "Nome", "Cognome", "Età", "Sesso",
"Email", "IP", "Numero Commenti")
VALUES
    ('john_doe', 'John', 'Doe', 28, 'M', 'john.doe@example.com',
'192.168.0.1', 0);

-- Popolamento Utente Cookieless

INSERT INTO "Utente Cookieless" ("IP", "ID Cookie")
VALUES
    ('192.168.0.1', '168615442');

-- Popolamento Commento

INSERT INTO "Commento" ("Data Scrittura", "Username Utente", "Ranking",
"Testo", "Link Articolo", "Titolo Articolo")
VALUES
    ('2023-10-08', 'ricky_raffons', 5, 'Questo articolo è davvero molto
interessante', 'https://libero.it/nuova-supercar-tesla', 'Nuova Supercar
Tesla');

-- Popolamento Accesso Registrato

INSERT INTO "Accesso Registrato" ("Sito", "Utente Registrato")
VALUES
```

```
    ('https://repubblica.it', 'not_quaily');

-- Popolamento Lettura Registrata

INSERT INTO "Lettura Registrata" ("Utente Registrato", "Link Articolo", "Titolo
Articolo", "Tempo Lettura Effettivo")
VALUES
    ('john_doe', 'https://tg24.sky.it/russia-ucraina-guerra-ultime-
notizie', 'Guerra Ucraina Russia, diga distrutta', '7');

-- Popolamento Lettura Anonima

INSERT INTO "Lettura Anonima" ("Utente Cookieless", "Link Articolo", "Titolo
Articolo", "Tempo Lettura Effettivo")
VALUES
    ('192.168.0.1', 'https://tg24.sky.it/russia-ucraina-guerra-ultime-
notizie', 'Guerra Ucraina Russia, diga distrutta', '5');

-- Popolamento Accesso Anonimo

INSERT INTO "Accesso Anonimo" ("Utente Cookieless", "Sito")
VALUES
    ('192.168.0.1', 'https://repubblica.it');

-- Popolamento Possiede Registrato

INSERT INTO "Possiede Registrato" ("Cookie Persistenti", "Sito")
VALUES
    ('0612105243', 'https://repubblica.it');

-- Popolamento Possiede Anonimo

INSERT INTO "Possiede Anonimo" ("Cookie Sessione", "Sito")
VALUES
    ('168615442', 'https://repubblica.it');

-- Popolamento Profilazione

INSERT INTO "Profilazione" ("Utente Registrato", "Cookie Persistenti")
VALUES
    ('not_quaily', '0612105243');
```

7. Query SQL

| <i>Workpackage</i> | <i>Task</i> | <i>Responsabile</i> |
|--------------------|-------------|---------------------|
| WP3 | SQL: Query | Raffone Riccardo |

7.1. Query con operatore di aggregazione e join: “Estrarre il numero di utenti registrati che accedono ad un sito con uno specifico indirizzo IP”

La query richiesta permette di ottenere il numero totale di utenti registrati che accedono ad un determinato sito, in base all'IP dell'Utente e al Nome Sito, sfruttando le relazioni **Utente Registrato**, **Sito** e **Accesso Registrato**.

```
SELECT COUNT (ALL ar."Utente Registrato") AS "Tot Utenti/Sito"
FROM "Accesso Registrato" ar
JOIN "Utente Registrato" ur ON ar."Utente Registrato" = ur."Username"
JOIN "Sito" s ON ar."Sito" = s."URL"
WHERE s."Nome Sito" = 'La Repubblica' AND ur."IP" = '192.168.0.11';
```

7.2. Query nidificata complessa: “Stampare i commenti con un ranking superiore alla media”

Questa query si occupa di stampare ogni commento che rispetti la condizione imposta, la quale prevede che il ranking dei commenti da restituire debba essere superiore al ranking medio per singolo articolo.

```
SELECT C.*
FROM "Commento" C
WHERE C."Ranking" > (
    SELECT AVG(C1."Ranking")
    FROM "Commento" C1
    WHERE C1."Link Articolo" = C."Link Articolo" AND C1."Titolo
Articolo" = C."Titolo Articolo"
)
ORDER BY C."Titolo Articolo";
```

7.3. Query insiemistica: “Estrarre quanti utenti di sesso femminile hanno visitato un determinato sito con un'età superiore ai 16 anni”

Per la query insiemistica, abbiamo deciso di effettuare un'operazione che va ad estrarre gli utenti registrati di sesso femminile che hanno visitato un sito specifico, nel caso in esame si è scelto “La Gazzetta dello Sport”, e che hanno un'età superiore ai 16 anni.

```
SELECT UR."Username"
FROM "Utente Registrato" UR
GROUP BY UR."Username"
HAVING UR."Età" > 16 AND UR."Sesso" = 'F'
INTERSECT
```

```
SELECT AR."Utente Registrato"  
FROM "Accesso Registrato" AR  
WHERE AR."Sito" = 'https://www.gazzetta.it'
```

7.4. Eventuali Altre query

7.4.1. “Estrarre i cookie di sessione che denotano l'utilizzo dello stesso Sistema Operativo da parte degli Utenti Cookieless”

La query indicata si occupa di individuare gli utenti cookieless, che utilizzano lo stesso Sistema Operativo, andando ad estrarre i cookie di sessione affibbiati agli stessi.

```
SELECT *  
FROM "Cookie Sessione" CS1  
WHERE EXISTS (  
    SELECT *  
    FROM "Cookie Sessione" CS2  
    WHERE CS2."Sistema Operativo" = CS1."Sistema Operativo" AND  
CS2."ID" <> CS1."ID"  
);
```

8. Viste

| <i>Workpackage</i> | <i>Task</i> | <i>Responsabile</i> |
|--------------------|-------------|----------------------|
| WP4 | Viste | Quagliuolo Francesco |

8.1. Vista Lettura Registrata

La vista ideata seleziona, dalla relazione **Lettura Registrata**, il *Titolo* degli articoli e, sfruttando la correlazione tra gli **Utenti Registrati** e gli **Articoli** letti, calcola il numero totale di letture per ogni articolo.

```
DROP VIEW IF EXISTS "Vista Lettura Registrata";
CREATE VIEW "Vista Lettura Registrata" ("Titolo Articolo", "NumUtenti") AS
    SELECT "Titolo Articolo", COUNT(*)
    FROM "Lettura Registrata"
    GROUP BY "Titolo Articolo";
```

8.1.1. Query con Vista: “Estrarre il titolo dell'articolo più letto”

La query ideata sulla vista precedentemente creata prevede l'estrazione del titolo dell'articolo più letto, basandosi sul valore massimo riscontrabile dalle letture complessive per singolo articolo individuabili tramite la vista.

```
SELECT "Titolo"
FROM "Articolo"
WHERE "Titolo" IN (
    SELECT "Titolo Articolo"
    FROM "Vista Lettura Registrata"
    WHERE "NumUtenti" = (
        SELECT MAX ("NumUtenti")
        FROM "Vista Lettura Registrata"
    )
);
```

9. Trigger

9.1. Trigger inizializzazione

| <i>Workpackage</i> | <i>Task</i> | <i>Responsabile</i> |
|--------------------|--|---------------------|
| WP1 | Trigger inizializzazione/popoloamento database | Postiglione Matteo |

9.1.1. Trigger1: Cardinalità minima Articolo-Sito

Il trigger in questione si occupa di effettuare un controllo sulla relazione **Articolo**, di modo da permettere l'inserimento di un articolo, con uno specifico *Titolo*, all'interno di uno ed un solo sito.

```
DROP TRIGGER IF EXISTS tr_controllo_articolo ON "Articolo";
CREATE OR REPLACE FUNCTION controllo_articolo()
```

```

RETURNS TRIGGER AS $$
BEGIN
    IF EXISTS (
        SELECT *
          FROM "Articolo"
         WHERE "Titolo" = NEW."Titolo" AND "URL Sito" <> NEW."URL Sito"
    ) THEN
        RAISE EXCEPTION 'Ogni articolo può appartenere ad uno e un solo sito';
    END IF;
    RETURN NEW;
END;
$$ LANGUAGE plpgsql;

CREATE TRIGGER tr_controllo_articolo
BEFORE INSERT ON "Articolo"
FOR EACH ROW
EXECUTE FUNCTION controllo_articolo();

```

9.1.2. Trigger2: Controllo Utente Cookieless

Questo trigger va a verificare il rispetto della cardinalità minima tra le relazioni **Utente Cookieless** e **Cookie Sessione**, impedendo l'assegnazione di uno stesso cookie a diversi utenti.

```

DROP TRIGGER IF EXISTS tr_utente_cookieless ON "Utente Cookieless";

CREATE OR REPLACE FUNCTION controllo_utente_cookieless()
RETURNS TRIGGER AS $$
BEGIN
    IF EXISTS (
        SELECT *
          FROM "Utente Cookieless"
         WHERE "IP" = NEW."IP" AND "ID Cookie" = NEW."ID Cookie"
    ) THEN
        RAISE EXCEPTION 'Ogni cookie di sessione è affibbiato ad un solo utente
cookieless';
    END IF;
    RETURN NEW;
END;
$$ LANGUAGE plpgsql;

CREATE TRIGGER tr_utente_cookieless
BEFORE INSERT ON "Utente Cookieless"
FOR EACH ROW
EXECUTE FUNCTION controllo_utente_cookieless();

```

9.2. Trigger per vincoli aziendali

| Workpackage | Task | Responsabile |
|-------------|------|--------------|
|-------------|------|--------------|

| | | |
|-----|-------------------------------|----------------------|
| WP4 | Trigger per vincoli aziendali | Quagliuolo Francesco |
|-----|-------------------------------|----------------------|

9.2.1. Trigger1: Limite Commenti

Il trigger in esame, si occupa di effettuare un controllo sui nuovi commenti prima ancora che essi siano inseriti nel database, bloccandone l'inserimento nel caso in cui l'utente abbia già commentato quello specifico articolo.

```

DROP TRIGGER IF EXISTS check_limite_commenti ON "Commento";

CREATE OR REPLACE FUNCTION limite_commenti()
RETURNS TRIGGER AS $$
BEGIN
    -- Verifica se l'utente ha già un commento per l'articolo corrente
    IF EXISTS (
        SELECT *
        FROM "Commento"
        WHERE "Username Utente" = NEW."Username Utente" AND "Titolo Articolo" =
NEW."Titolo Articolo"
    ) THEN
        RAISE EXCEPTION 'L''utente ha già fatto un commento per questo
articolo.';
    END IF;
    RETURN NEW;
END;
$$ LANGUAGE plpgsql;

CREATE TRIGGER check_limite_commenti
BEFORE INSERT ON "Commento"
FOR EACH ROW
EXECUTE FUNCTION limite_commenti();

```

9.2.2. Trigger2: Aggiorna Numero Commenti

Questo trigger effettua un'operazione di aggiornamento dell'attributo *Numero Commenti* appartenente alla relazione **Utente Registrato**, tramite una derivazione risultante dall'inserimento e/o eliminazione di tuple nella relazione **Commento**.

```

DROP TRIGGER IF EXISTS trigger_aggiorna_numero_commenti ON "Commento";

CREATE OR REPLACE FUNCTION aggiorna_numero_commenti()
RETURNS TRIGGER AS $$
BEGIN
    IF TG_OP = 'INSERT' THEN
        UPDATE "Utente Registrato" AS ur
        SET "Numero Commenti" = ur."Numero Commenti" + 1
        WHERE ur."Username" = NEW."Username Utente";
    ELSIF TG_OP = 'DELETE' THEN

```

```

        UPDATE "Utente Registrato" AS ur
        SET "Numero Commenti" = ur."Numero Commenti" - 1
        WHERE ur."Username" = OLD."Username Utente";
    END IF;
    RETURN NULL;
END;
$$ LANGUAGE plpgsql;

CREATE TRIGGER trigger_aggiorna_numero_commenti
AFTER INSERT OR DELETE ON "Commento"
FOR EACH ROW
EXECUTE FUNCTION aggiorna_numero_commenti();

```

9.2.3. Trigger3: Controllo Ranking

Il trigger in esame controlla che, durante l'inserimento di un commento, quest'ultimo abbia l'attributo *Ranking* compreso tra i valori 1 e 5.

```

DROP TRIGGER IF EXISTS tr_controllo_ranking ON "Commento";

CREATE OR REPLACE FUNCTION controllo_ranking()
RETURNS TRIGGER AS $$
BEGIN
    IF NEW."Ranking" < 1 OR NEW."Ranking" > 5 THEN
        RAISE EXCEPTION 'Il ranking di un commento deve essere compreso tra 1 e
5.';
    END IF;
    RETURN NEW;
END;
$$ LANGUAGE plpgsql;

CREATE TRIGGER tr_controllo_ranking
BEFORE INSERT ON "Commento"
FOR EACH ROW
EXECUTE FUNCTION controllo_ranking();

```

9.2.4. Trigger4: Storico Cookie di Sessione

Nel caso siffatto, il trigger va a creare una nuova relazione, denominata “**Cookie Sessione Scaduti**” dove vengono immagazzinati i **Cookie di Sessione** eliminati al termine della navigazione da parte dell'utente.

```

DROP TRIGGER IF EXISTS tr_controllo_cookie_sessione ON "Cookie Sessione";

-- Creazione tabella di appoggio
CREATE TABLE "Cookie Sessione Scaduti" (
    "ID" INTEGER PRIMARY KEY,
    "Nome Cookie" VARCHAR(20) NOT NULL,

```

```
        "Descrizione" VARCHAR NOT NULL,  
        "Sistema Operativo" VARCHAR NOT NULL,  
        "Lingua" VARCHAR NOT NULL,  
        "Schermo" VARCHAR NOT NULL,  
        "Browser" VARCHAR NOT NULL,  
        "Data Creazione" DATE NOT NULL  
    );  
  
CREATE OR REPLACE FUNCTION controllo_cookie_sessione()  
RETURNS TRIGGER AS $$  
BEGIN  
    IF TRUE  
    THEN  
        INSERT INTO "Cookie Sessione Scaduti" ("ID", "Nome Cookie",  
"Descrizione", "Sistema Operativo", "Lingua", "Schermo", "Browser", "Data  
Creazione")  
            VALUES (OLD."ID", OLD."Nome Cookie", OLD."Descrizione",  
OLD."Sistema Operativo", OLD."Lingua", OLD."Schermo", OLD."Browser", OLD."Data  
Creazione");  
    END IF;  
    RETURN OLD;  
END;  
$$ LANGUAGE plpgsql;  
  
CREATE TRIGGER tr_controllo_cookie_sessione  
AFTER DELETE ON "Cookie Sessione"  
FOR EACH ROW  
EXECUTE FUNCTION controllo_cookie_sessione();
```