

PROGETTO BASI DI DATI

GAMONET

1 Abstract

Gamonet è un servizio web di live streaming che permette alle persone di guardare o creare contenuti di varie tipologie, videogiochi e categorie in live.

Gli utenti che vorranno usufruire del servizio dovranno creare un account sul sito fornendo come dati per la registrazione indirizzo email, username, ed una password. Una volta registrati al sito gli utenti si divideranno principalmente tra streamer e spettatori. Gli streamer sono utenti selezionati che avranno possibilità di effettuare dirette, mentre gli spettatori sono coloro che le guarderanno.

Ogni streamer ha un proprio canale, dove sono raccolte le sue informazioni personali, le sue dirette e clip. Le clip sono piccoli frammenti di diretta salvati sul canale, non più lunghi di 300 secondi, ritagliate e create dagli utenti. Lo streamer, inoltre, accedendo al proprio canale avrà la possibilità di controllare e vedere, in una parte privata di questo, le statistiche del suo canale, le informazioni della sua affiliazione, guadagni, ed altro.

Le dirette hanno anche un sistema di commenti per far interagire gli spettatori con lo streamer, in questo, sarà anche possibile utilizzare delle emote speciali del canale. Per controllare i messaggi inviati c'è la possibilità di scegliere alcuni spettatori come moderatori, affinché questi possano moderare ed eliminare i commenti effettuati durante una diretta che ritengono meno opportuni.

Infine per gli streamer c'è anche la possibilità di essere affiliato a Gamonet, avendo così dei guadagni dalle loro dirette, infatti, gli spettatori possono sostenere gli streamer grazie ad un abbonamento ricorrente, inoltre, nelle dirette c'è la possibilità per delle aziende esterne di inserire una sponsorizzazione tramite uno spot pubblicitario. Questi guadagni saranno infine pagati agli streamer tramite un metodo di pagamento da loro scelto.

2 Analisi dei requisiti

2.1 Descrizione testuale

Si vuole realizzare una base di dati per l'amministrazione delle informazioni degli utenti, le dirette e le affiliazioni all'interno del sito.

L'Account è l'entità base del database dal momento che ogni utente che usufruisce del servizio deve possederne uno. Dell'utente ci interessa salvare:

- Username
- Email
- Password
- Canali seguiti
- Canali abbonato

Inoltre, ogni utente fa parte di una categoria in base al ruolo che ricopre, infatti un utente può essere streamer, spettatore o moderatore.

Ogni **Streamer** ha accesso ad un **canale** personale, nel quale sono mostrate le sue dirette, le clip e le sue informazioni pubbliche da condividere con il pubblico, queste sono:

- Nome canale
- Informazioni personali
- Dirette
- Clip
- Emote
- Moderatori
- Abbonati
- Follower

Lo streamer, inoltre può avere un'**Affiliazione** con Gamonet, in modo tale da poter ottenere una retribuzione in base alle dirette effettuate. La retribuzione è composta dal numero di abbonati e da eventuali sponsorizzazioni effettuate dalle aziende durante le sue dirette. Nella base di dati viene salvato:

- Numero Affiliazione
- Metodo di pagamento
- Guadagni totali

Le **Aziende** esterne possono sponsorizzare i canali per farsi pubblicità tramite il servizio grazie all'inserimento di spot pubblicitari durante le varie dirette, per poter usufruire di questa possibilità devono fornire:

- P. Iva
- Email
- Nome
- Tipologia
- Link allo spot pubblicitario

Le **Clip** sono dei momenti salienti delle live che gli utenti voglio salvare all'interno del canale, di esse vengono salvate:

- Titolo
- Durata
- Link
- Visualizzazioni

Le **Emote** delle immagini personalizzate create dai canali ed utilizzabili nei commenti delle dirette in modo da arricchirli ulteriormente, per essere aggiunte e salvate in un canale bisogna dargli un nome ed il link dell'immagine:

- Nome
- Link

La **Diretta** è la vera e propria trasmissione live effettuata dagli streamer nel loro canale, quando una diretta è creata avrà:

- Id Diretta
- Ora inizio
- Titolo
- Visualizzazioni

Le **Categorie** sono i vari argomenti che possono essere trattati in una live, essi spaziano da videogiochi ad attività della vita quotidiana, sul sito sono rappresentate da:

- Id categoria
- Nome

2.2 Glossario dei termini

Termine	Descrizione	Collegamento
Account	Profilo usato dagli utenti nel servizio.	Entità padre di Streamer, Moderatore e Spettatore.
Streamer	Account abilitato alla creazione di dirette e gestione del canale.	Entità figlia di Account, collegata a Canale ed Affiliazione.
Moderatore	Account abilitato per moderare un canale e dirette.	Entità figlia di Account, collegata a Canale.
Spettatore	Account che guarda le dirette nei vari canali.	Entità figlia di Account, collegata a Canale, Clip e Diretta.
Affiliazione	Contratto tra il servizio e lo streamer che permette a questo di ottenere guadagni dalle sue dirette.	Collegata a Streamer.
Canale	Pagine web dove visualizzare le informazioni, dirette e clip.	Collegata a Streamer, Moderatore, Spettatore, Clip, Emote, Azienda, Diretta.
Diretta	Trasmissione in diretta da uno streamer nel suo canale.	Collegata a Canale e Categoria.
Categoria	Argomento trattato dallo streamer durante la sua diretta.	Collegata a Diretta.
Azienda	Effettua sponsorizzazioni sui canali per mostrare un suo spot durante le dirette.	Collegata a Canale.
Clip	Frammento di diretta estrapolato da un utente, salvato nel canale.	Collegata a Spettatore e Canale.
Emote	Immagine personalizzata creata da un canale per essere utilizzata in chat durante la diretta.	Collegata a Canale.

2.3 Operazioni tipiche

OPERAZIONE	TIPO	FREQUENZA
Registrazione di un nuovo utente	S	10000 al giorno
Ricerca di un canale	L	5000 al giorno
Inserimento di una nuova emote	S	100 al giorno
Sponsorizzazione di un canale	S	10 al giorno
Abbonamento ad un canale	S	5000 al giorno
Iscrizione ad un canale	S	15000 al giorno
Creazione di una nuova clip	S	2500 al giorno
Visualizzazione di una clip	L	3000 al giorno
Iniziare una nuova diretta	S	1500 al giorno
Calcolare i guadagni totali di un canale	L	2000 al mese
Creazione di un canale	S	150 al giorno

3 Progettazione Concettuale

3.1 Lista entità

Gli attributi se non specificato sono NOT NULL

- **Account**

- Username VARCHAR(15) PK
- Email VARCHAR(50)
- Password VARCHAR(64)

Questa entità si divide in sottocategorie:

- Streamer : Gestisce il canale ed esegue le dirette
- Moderatore : Modera il canale e la chat durante le dirette
- Spettatore : Può seguire o abbonarsi a dei canali e guardare le dirette

- **Affiliazione**

- Id_Affiliazione INT PK
- Metodo di pagamento VARCHAR(25)
- Guadagni Totali DECIMAL

- **Canale**

- Nome VARCHAR(25) PK
- Informazioni VARCHAR(250)

- **Azienda**

- P_IVA VARCHAR(11) PK
- Email_azienza VARCHAR(50)
- Nome_azienza VARCHAR(50)
- Link VARCHAR(250)
- Tipologia VARCHAR(50)

- **Emote**

- Nome VARCHAR(25) PK
- Link VARCHAR(100)

- **Clip**

- Id_Clip INT PK
- Titolo VARCHAR(52)
- Durata INT
- File loc VARCHAR(100)

- **Diretta**

- Id_Diretta INT PK
- Titolo VARCHAR(25)
- Ora Inizio TIMESTAMP
- Visualizzazioni INT

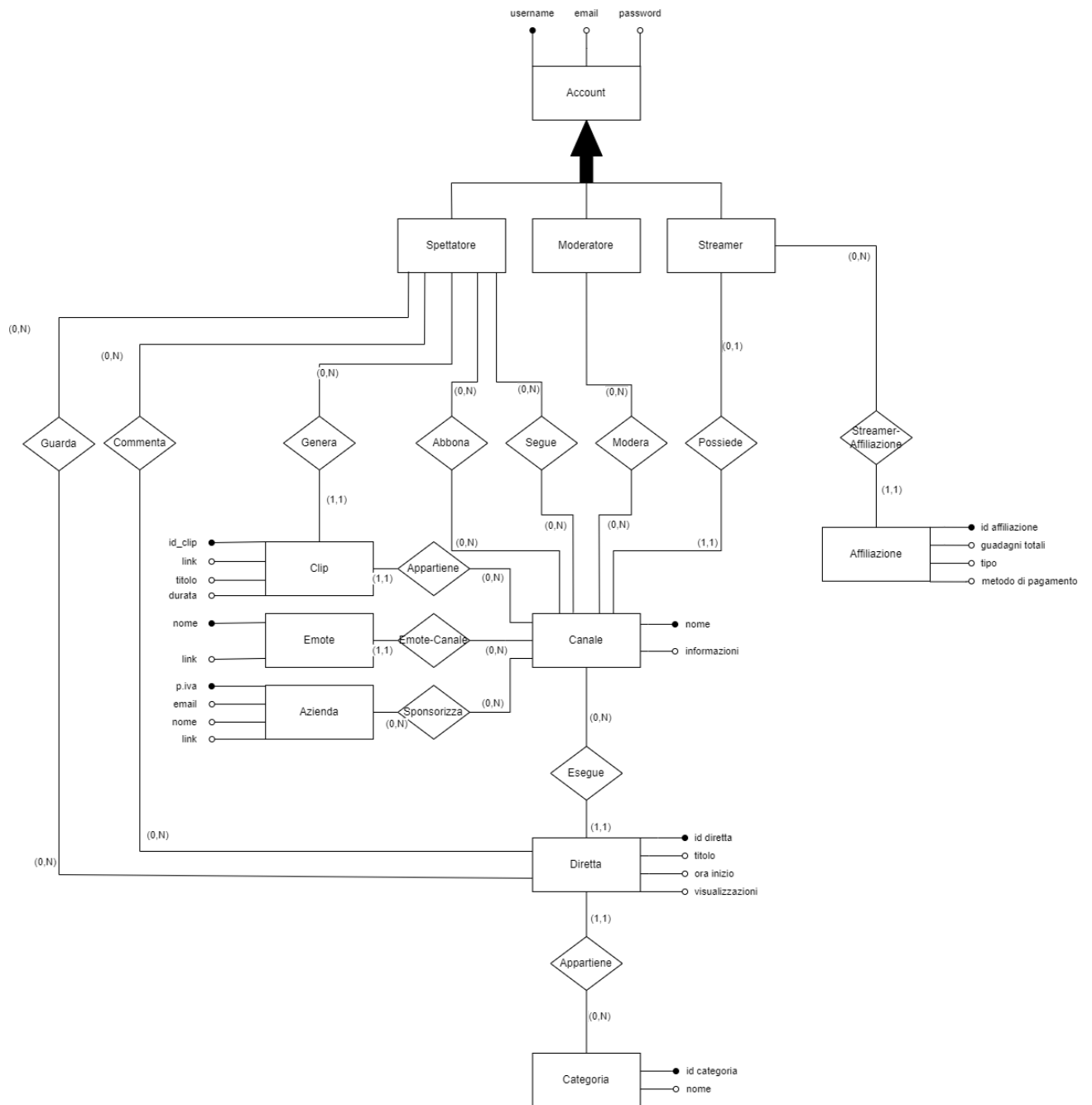
- **Categoria**

- Id_Categoria INT PK
- Nome VARCHAR(25)

3.2 Tabella delle relazioni

Relazione	Entità coinvolte	Descrizione	Attributi
Modera	Moderatore (0,N) Canale (0,N)	Un moderatore controlla i commenti delle dirette, eliminando messaggi inopportuni o link inviati dagli spettatori.	Nessuno
Segue	Spettatore (0,N) Canale (0,N)	Uno spettatore può seguire un canale per essere informato sulle sue attività e dirette.	Nessuno
Abbona	Spettatore (0,N) Canale (0,N)	Uno spettatore può sostenere un canale sottoscrivendo un abbonamento a questo, può scegliere tra 3 diversi tipi di abbonamento con diversi prezzi.	Grado Abbonamento INT = 4, 8, 20
Sponsorizza	Azienda(0,N) Canale (0,N)	Un'azienda può pagare un canale per trasmettere una propria pubblicità durante una diretta.	Prezzo Sponsorizzazione INT > 0
Emote-Canale	Canale (0,N) Emote (1,1)	Un canale ha la possibilità di avere le proprie emote che possono essere utilizzate dagli utenti abbonati a quel canale nei commenti delle dirette.	Nessuno
Appartiene	Canale (0,N) Clip (1,1)	Un canale ha una raccolta di clip generate dagli spettatori durante le dirette.	Nessuno
Genera	Spettatore (0,N) Clip (1,1)	Uno spettatore può generare delle clip per immortalare dei momenti salienti della diretta.	Nessuno
Commenta	Spettatore (0,N) Diretta (0,N)	Uno spettatore può commentare una diretta per avere una interazione con lo streamer.	Testo del commento VARCHAR(200)
Guarda	Spettatore (0,N) Diretta (0,N)	Uno spettatore guarda la diretta che gli interessa.	Nessuno
Possiede	Streamer (0,1) Canale (1,1)	Uno Streamer ha il proprio canale dove eseguire le dirette.	Nessuno
Streamer-Affiliazione	Streamer (0,N) Affiliazione (1,1)	Uno Streamer può avere una ed una sola affiliazione con il servizio.	Nessuno
Appartiene	Diretta (1,1) Categoria (0,N)	Una diretta, a seconda del contenuto trasmesso, si classifica in una categoria.	Nessuno
Esegue	Canale (0,N) Diretta (1,1)	Un canale trasmette le dirette.	Nessuno

3.3 Schema ER



4 Progettazione Logica

4.1 Ristrutturazione

4.1.1 Analisi delle ridondanze

Dal momento che in diretta l'attributo “**visualizzazioni**” viene aggiornato continuamente, mentre la sua lettura non è richiesta frequentemente, è stato scelto di rimuoverlo e di calcolare il valore ogni volta che una sua lettura viene richiesta.

Inoltre abbiamo deciso di rimuovere l'attributo “**guadagni totali**” dall'entità Affiliazione in quanto, essa viene richiesta un numero relativamente basso di volte, perchè può essere visualizzata solo dallo streamer, al contrario il suo valore verrebbe aggiornato ogni qual volta venga sottoscritto un nuovo abbonato al canale o venga effettuata una nuova sponsorizzazione.

4.1.2 Eliminazione delle generalizzazioni

Account ← Streamer, Moderatore, Spettatore : Dal momento che le varie tipologie di account (Streamer, Moderatore e Spettatore) non si differenziano dall'entità padre (Account) se non per le relazioni che hanno con le altre entità, quindi abbiamo eliminato le entità figlie dallo schema, aggiungendo all'entità padre un nuovo attributo “ruolo” per specificare la tipologia dell'account e permettergli di compiere le azioni ad esso disponibili.

Account (username, email, password, tipo utente)
Affiliazione (id affiliato, metodo di pagamento, account → Account.username)
Canale (nome, informazioni, streamer → Account.username)
Clip (id clip, titolo, durata, localizzazione, account → Account.username, canale → Canale.nome)
Emote (nome, link, canale → Canale.nome)
Azienda (partita iva, email, nome, link, tipologia)
Diretta (id diretta, canale → Canale.nome, titolo, ora inizio, categoria → Categoria.id)
Categoria (id, nome)
Abbonati (id account → Account.username, canale → Canale.nome, costo)
Affiliazione (id affiliato, streamer → Account.username, metodo di pagamento)
Commenta (id commento, spettatore → Account.username, diretta → Diretta.id, commento)
Guarda (spettatore → Account.username, diretta → Diretta.id)
Moderà (moderatore → Account.username, canale → Canale.nome)
Seguiti (spettatore → Account.username, canale → Canale.nome)

5 Query e indici

5.1 Query

1) Mostra il numero totale di visualizzazioni per categoria

```
SELECT COUNT(account.username), categoria.nome
FROM diretta
JOIN categoria ON categoria.id_categoria=diretta.categoria
JOIN guarda ON guarda.id_diretta=diretta.id_diretta
JOIN account ON account.username=guarda.id_account
GROUP BY categoria.nome;
```

Totale	Nome
88	Fortinte
83	AssettoCorsa
65	HotTub
92	Dota2
65	Sport
73	Apex Legend
65	Valorant
53	Minecraft
32	GtaV
63	CounterStrike
97	Factorio
80	Elden Ring
46	Fifa

2) Mostra il canale con il guadagno più alto sul sito

```
SELECT SUM(prezzo) + MAX(somma.guadagno) AS totale, somma.nome FROM (
SELECT SUM(costoAbbonamento) AS guadagno, canale.nome
FROM abbonati
JOIN canale ON canale.nome=abbonati.id_canale
GROUP BY canale.nome
ORDER BY guadagno DESC) AS somma
INNER JOIN sponsorizza ON sponsorizza.id_canale=somma.nome
GROUP BY somma.nome
ORDER BY totale DESC
LIMIT 1;
```

totale	nome
30186	qolekhov3

3) Mostra i canali che hanno più di 3 clip

```
SELECT COUNT(*), canale.nome
FROM canale
INNER JOIN clip ON clip.id_canale=canale.nome
GROUP BY canale.nome
HAVING COUNT(*)>3;
```

Totale	nome
4	smontgomeryu
5	qdewsnap3c
4	ksemper12
6	dlarwell16
4	frigbye2p
4	ksymingtong
4	jgamlin1v
5	kgarbutt3d
4	aferreras2y
4	tlOXdale1t
5	emoorton1l

4) Mostra il numero totale di commenti per la categoria: "HotTub"

```
SELECT COUNT(*), categoria.nome
FROM commenta
INNER JOIN diretta ON diretta.id_diretta=commenta.id_diretta
INNER JOIN categoria ON categoria.id_categoria=diretta.categoria
WHERE categoria.nome='HotTub'
GROUP BY categoria.id_categoria
```

```
Totale|nome
-----+-----
41|HotTub
```

5) Mostra la categoria che ha più moderatori

```
SELECT COUNT(*) as tot, categoria.nome
FROM modera
INNER JOIN canale ON canale.nome=modera.id_canale
INNER JOIN diretta ON diretta.id_canale=canale.nome
INNER JOIN categoria ON categoria.id_categoria=diretta.categoria
GROUP BY categoria.id_categoria
ORDER BY tot DESC
LIMIT 1;
```

```
totale|nome
-----+-----
8|Factorio
```

5.2 Indici

La categoria di una diretta viene mostrata ogni qual volta si inizi a guardare una diretta ed anche quando le dirette vengono ricercate in base alla categoria, quindi un numero modesto di volte, mentre, l'inserimento di nuove categorie avviene molto di rado, in quanto queste sono aggiunte solo solo all'uscita di un nuovo videogioco o quando nuove attività svolte dagli streamer diventano popolari.

Quindi, nell'ipotetico utilizzo su larga scala del servizio Gamonet da parte di un ampio numero di spettatori si è deciso di implementare un indice sugli attributi id_categoria e nome in categoria.

```
CREATE INDEX diretta_categoria ON "categoria" ("id_categoria", "nome");
```

6 Codice C++

6.1 Descrizione dell'utilizzo del codice

Il codice C++ per l'esecuzione delle query consiste in un unico file .cpp, che va compilato, attraverso il comando g++ "codice.cpp" -o querys. Per eseguire il codice, basta avviare l'eseguibile "querys". Una volta effettuata la connessione al database, il programma mostrerà a schermo la lista delle query, identificate da un numero da 1 a 5, eseguibili all'interno del programma. Per eseguire una query bisogna inserire da tastiera il numero della query scelta, mentre per terminare l'esecuzione del programma va inserito 'q'.