

SOCIAL NETWORK PAKY&ROM



PAKY&ROM

RELAZIONE: SOCIAL NETWORK PAKY&ROM
STUDENTI: KHAN SHAZ, NECULA ROBERT
CLASSE: 5°INA

INDICE

INDICE	1
ANALISI	2
SCHEMA E/R	3
REGOLE DI LETTURA	4
MODELLO LOGICO	4
FUNZIONAMENTO	5
NORMALIZZAZIONE	6
INTEGRITÀ REFERENZIALE	6
FUNZIONI AGGIUNTIVE	6

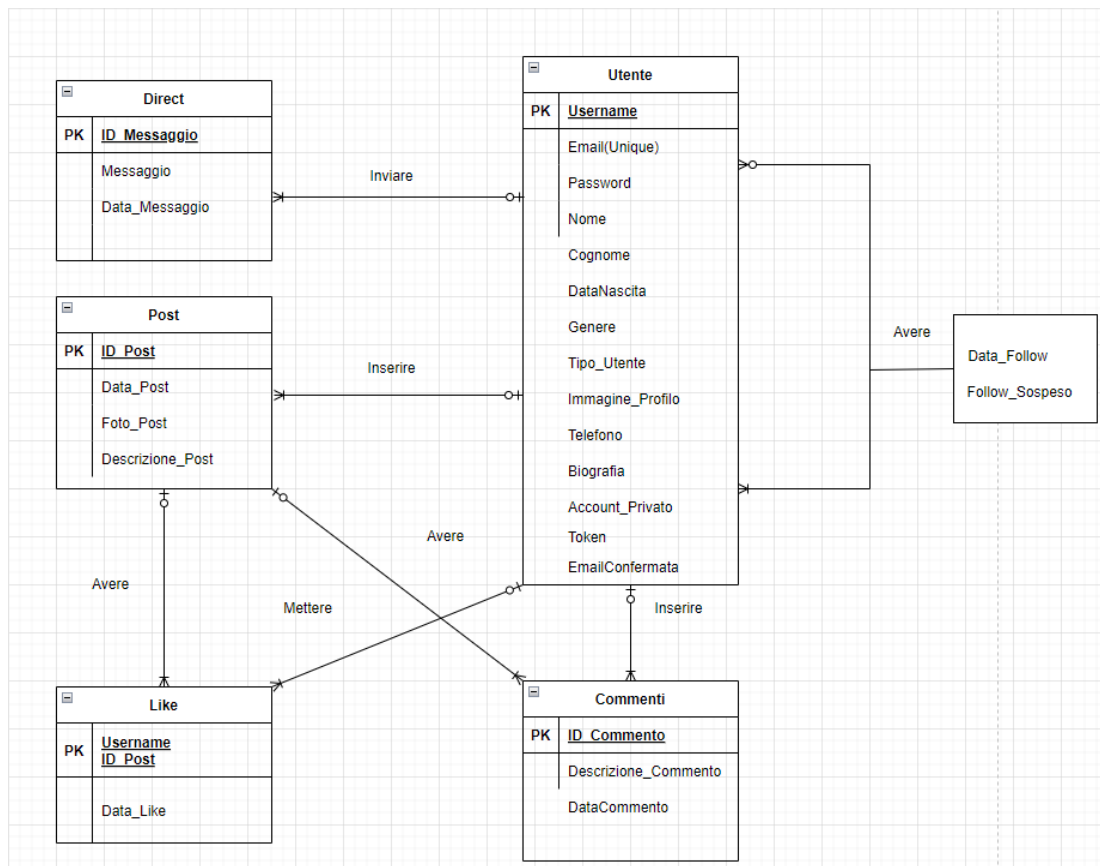
ANALISI

POSSIAMO INDIVIDUARE MOLTEPLICI ENTITA':

1. **UTENTE** che avrà come attributi Username(PK), Email(unica), Password, Nome, Cognome, DataNascita, Genere, Tipo_Utente, Immagine_Profilo, Telefono, Biografia, Account_Privato, Token, EmailConfermata;
 2. **POST** che avrà come attributi Id_Post(PK), Data_Post, Foto_Post, Descrizione_Post, Username(FK);
 3. **DIRECT** che avrà come attributi Id_Messaggio(PK), Messaggio, Data_Messaggio, Username_Mittente(FK), Username_Destinatario(FK);
 4. **COMMENTI** che avrà come attributi Id_Commento(PK), Commento, Data_Commento, Id_Post(FK), Username(FK);
 5. **LIKE** che avrà come chiave primaria una chiave combinata ovvero Username(FK di Utente) e ID_Post(FK di Post) poi avrà un attributo Data_Like.
 6. **FOLLOWER** che avrà chiave primaria una chiave combinata ovvero Username_Segue(FK di Utente) e Username_Seguito(FK di Utente) e poi avrà come attributi Data_Follow e Follow_Sospeso.
- Abbiamo creato una relazione 1 a N tra l'entità utente e l'entità direct dove l'entità direct avrà un attributo aggiuntivo ovvero la chiave primaria dell'entità utente(username) che sarà la chiave esterna dell'entità direct.
 - Abbiamo creato una relazione 1 a N tra l'entità utente e l'entità post dove l'entità post avrà un attributo aggiuntivo ovvero la chiave primaria dell'entità utente(username) che sarà la chiave esterna dell'entità post.
 - Abbiamo creato una relazione 1 a N tra l'entità utente e l'entità like dove l'entità like avrà due attributi aggiuntivi ovvero la chiave primaria dell'entità utente(username) e la chiave primaria dell'entità post(Id_Post) che formeranno una chiave combinata che sarà la chiave primaria dell'entità like.
 - Abbiamo creato una relazione 1 a N tra l'entità utente e l'entità commento dove l'entità commento avrà un attributo aggiuntivo ovvero la chiave primaria dell'entità utente(username) che sarà la chiave esterna dell'entità commento.

- Abbiamo creato un relazione 1 a N tra l'entità post e l'entità like dove l'entità like avrà un attributo aggiuntivo ovvero la chiave primaria dell'entità post(id_post) che sarà la chiave esterna dell'entità like.
- Abbiamo creato un relazione 1 a N tra l'entità post e commento dove l'entità commento avrà un attributo aggiuntivo ovvero la chiave primaria dell'entità post(id_post) che sarà la chiave esterna dell'entità commento.
- Abbiamo creato una self join N a N con l'entità utente da cui si creerà un'altra entità chiamata follower che avrà una chiave combinata come chiave primaria e sarà formata dalla chiave primaria dell'entità utente (Username che in questo caso sarà chiamata Username_Segue) e l'altra parte della chiave pure sarà la chiave primaria dell'entità utente(Username che in questo caso sarà chiamata Username_Seguito) in modo tale da conoscere l'username della persona seguita e di quella che segue

SCHEMA E/R



REGOLE DI LETTURA

- Un utente può avere uno o più follower
- Un follower deve essere un utente
- Un utente può inviare uno o più direct
- Un direct deve essere inviato da un utente
- Un utente può inserire uno o più post
- Un post deve essere inserito da un utente
- Un utente può mettere uno o più like
- Un like deve essere messo da un utente
- Un utente può inserire uno o più commenti
- Un commento deve essere inserito da un utente
- Un post può avere uno o più like
- Un like deve appartenere a un post
- Un post può avere uno o più commenti
- Un commento deve appartenere a un post

MODELLO LOGICO

ENTITA'	ATTRIBUTO	CHIAVE	FORMATO	DIMENSIONE
Utente	Username	PK UNIQUE	VARCHAR	20
	Email		VARCHAR	100
	Password		VARCHAR	50
	Nome		VARCHAR	50
	Cognome		VARCHAR	50
	DataNascita		DATE	
	Genere		VARCHAR	10
	Tipo_Utente		VARCHAR	10
	Immagine_Profilo		VARCHAR	1000
	Telefono		VARCHAR	10
	Biografia		VARCHAR	150
	Account_Privato		BOOLEAN	
	Token		VARCHAR	255
	EmailConfermata		BOOLEAN	
Direct	Id_Messaggio	PK	VARCHAR	10
	Messaggio		VARCHAR	500
	Data_Messaggio	FK FK	DATE	
	Username_Mittente		VARCHAR	20
	Username_Destinatario		VARCHAR	20

Post	Id_Post	PK	VARCHAR	10
	Data_Post		DATE	
	Foto_Post		VARCHAR	1000
	Descrizione_Post		VARCHAR	100
	Username	FK	VARCHAR	20
Commenti	Id_Commento	PK	VARCHAR	10
	Descrizione_Commento		VARCHAR	100
	DataCommento		DATE	
	Id_Post	FK	VARCHAR	10
	Username	FK	VARCHAR	20
Like	Username	PK(FK)	VARCHAR	20
	Id_Post	PK(FK)	VARCHAR	10
	Data_Like		DATE	
Follower	Account_Sospeso		BOOLEAN	
	Data_Follow		DATE	
	Username_Segue	PK(FK)	VARCHAR	20
	Username_Seguito	PK(FK)	VARCHAR	20

FUNZIONAMENTO

Una volta entrati nel sito bisogna effettuare il login o nel caso non si avesse un account bisogna fare la registrazione:

- **REGISTRAZIONE**: Una volta inseriti i dati, l'utente non registrato dovrà aspettare la conferma dell'account tramite l'email specificata in precedenza, una volta confermata l'email l'account sarà attivo.
- **LOGIN**: Per accedere al proprio account bisogna inserire l'email e la password e una volta fatto ciò sarà possibile visualizzare i post delle persone seguite, creare nuovi post, seguire nuovi utenti, visualizzare il proprio profilo, visualizzare il profilo di altri utenti, modificare i propri dati, modificare l'immagine di profilo, modificare l'account da pubblico a privato o viceversa ed effettuare il logout.
- **ACCOUNT PRIVATO**: Un utente con un account privato può decidere da chi farsi seguire quindi per visualizzare i post di un utente con account privato bisogna per forza essere un suo follower.
L'utente con l'account privato riceverà una notifica la quale conterrà l'username della persona che lo vuole seguire e l'utente con l'account privato avrà la possibilità di accettare la richiesta quindi di farsi seguire o rifiutarla quindi di non farsi seguire.

NORMALIZZAZIONE

1. Il database segue la Prima Forma Normale dato che tutte l'entità presentano una chiave primaria e inoltre ogni attributo rappresenta informazioni elementari.
2. Il database segue la Seconda Forma Normale dato che segue la 1FN e inoltre tutti gli attributi non chiave dipendono dall'intera chiave composta.
3. Il database segue la Terza Forma Normale dato che segue la 2FN quindi anche la 1FN e inoltre tutti gli attributi non chiave dipendono solo da attributi chiave.

INTEGRITÀ REFERENZIALE

Il database segue le regole dell'integrità referenziale grazie alle constraint che ci permette di collegare la chiave primaria di un'entità alla chiave esterna di un'altra entità quindi la chiave esterna non può assumere valori diversi dalla chiave primaria della tabella collegata inoltre per modificare o cancellare i valori della chiave esterna bisogna prima modificare o cancellare la chiave primaria della tabella collegata.

FUNZIONI AGGIUNTIVE

1. Direct(messaggi privati)
2. Storie(post temporanei)