

# DBMS PROJECT

---

**Project Title : Social networking service.**

Team members:	Pradeep Reddy	Y10UC221
	B Mourya	Y10UC098
	K Raj Koushik Reddy	Y10UC138
	G Ruthvik Reddy	Y10UC111

**Database Schema :**

**List of Tables**

User_data	
->username	->email
->firstname	->password
->lastname	->image
->gender	->birthday

Wall_data
->email
->time
->status

Friends
->id1
->id2
->relation

Messages
->id1
->id2
->message
->time

**Constraints:**

**User\_data:**

username varchar (255) NOT NULL  
firstname varchar(255) NOT NULL  
lastname varchar(255) NOT NULL  
gender enum('m','f') NOT NULL default 'm'  
birthday date NOT NULL default '0000-00-00'  
email varchar(255) NOT NULL  
password varchar(255) NOT NULL  
dp varchar(255) DEFAULT images/def.jpg

->Primary key is email.

#### **Wall\_data:**

datetime NOT NULL default '0000-00-00 00:00:00'  
email varchar (255) NOT NULL  
status varchar(10000) NOT NULL

->Primary Key is (email+time).

->Foreign Key email references email id field of user\_data table.

#### **Messages:**

Id1 varchar (255) NOT NULL  
Id2 varchar (255) NOT NULL  
message varchar(10000) NOT NULL

time NOT NULL default 0000-00-00 00:00:00

-->Primary key is id1+id2+time.

->id1 is foreign key which references email id field in user\_data field.

-> id1 is foreign key which references email id field in user\_data field.

### **Friends:**

Id1 varchar (255) NOT NULL

Id2 varchar (255) NOT NULL

Relation char NOT NULL

->Primary key is (id1 + id2).

->id1 is foreign key which references "email id" field in user\_data field.

-> id1 is foreign key which references" email id" field in user\_data field.

### **MODULES:**

#### **Registration :**

In the user registration page, the user is asked to register by giving the required details. The data given by the user is stored in the table "user\_data". The email id given is the primary key. Only one account can be created with one email id. If the id is already registered and user tries to register again and it shows that the email id is already in use.

**Login:**

Once the user successfully registers, he can login using his “email id” and “password”. The entered email id and password if matched with the data present in “user\_data” table, the user is directed to his home page. The home page has user details wall, search bar. Otherwise it shows invalid username or password.

**Search:**

The user can search for friends. The search option takes a string from the user, the string can be a friends “firstname” or “email id”. The string is matched with firstname or email id’s present in “user\_data” table and suitable suggestions are automatically provided to the user as soon as the user starts typing in the search bar.

**Friends:**

Once a friend is searched, the user can open his profile. The user gets an option to send friend request to him, if he is not already his friend. Once a friend request is accepted the id’s of the friends are saved in “friends” table as id1, id2 and “relation” type. In friends table the relation field can be Y or N. Y implies request is sent and it is accepted. N implies friend request is pending.

**Wall:**

The user can write anything on his wall. The wall status is stored in “wall\_data” table along with its date time. Email along with its time is used as primary key to identify the wall post. The friend request can be seen on the wall of the friend. Email id is used as foreign key to link it with user.

**Messages:**

The user can send messages by clicking send message link. The message is saved in message table in message field. It is identified by message id1, id2 and time as primary key. Sent and received messages can be checked in message box provided on profile page. id1 and id2 are used to link messages with users.

**Photo Upload:**

The user can upload or update his profile pic by clicking on his image. The image is stored in "user\_data" table in the image field.

**Source code:**

Find the source code in "source\_code" folder, attached along with this document.