Accouts table hods information related to the user account . Ths account is linked to User account by Foriegn key user\_details. user can also link his twiter or facebook account by sharing the Authid. we can store the key and type of autprovider to this table. This table also handles administration of the Account like suspending of the account and so on .

CREATE TABLE `account` (

`idAccount` int(11) NOT NULL AUTO\_INCREMENT,

`user` int(11) DEFAULT NULL,

`suspened` datetime DEFAULT NULL,

`authProvider` enum('twitter','facebook') DEFAULT NULL,

`authId` varchar(50) DEFAULT NULL,

PRIMARY KEY (`idAccount`),

UNIQUE KEY `idAccount\_UNIQUE` (`idAccount`),

UNIQUE KEY `user\_UNIQUE` (`user`),

UNIQUE KEY `authId\_UNIQUE` (`authId`),

CONSTRAINT `user\_details` FOREIGN KEY (`user`) REFERENCES `users` (`id`) ON DELETE CASCADE ON UPDATE CASCADE

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='Account details of the user are stored in this table ';

Table user holds the infromation of individual user who have signed up to creative witer website.

CREATE TABLE `users` (

`FirstName` varchar(45) DEFAULT NULL,

`id` int(11) NOT NULL AUTO\_INCREMENT,

`LastName` varchar(45) DEFAULT NULL,

`UserName` varchar(45) DEFAULT NULL,

`password` varchar(45) DEFAULT NULL,

`Country` varchar(45) DEFAULT NULL,

`state` varchar(45) DEFAULT NULL,

`district` varchar(45) DEFAULT NULL,

`StreeAddress1` varchar(45) DEFAULT NULL,

PRIMARY KEY (`id`),

UNIQUE KEY `id\_UNIQUE` (`id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

Posts table contains posts of the user that are linked to the user post\_user foriegn key

CREATE TABLE `posts` (

`idposts` int(10) unsigned NOT NULL,

`title` varchar(45) DEFAULT NULL,

`tags` varchar(45) DEFAULT NULL,

`category` varchar(45) DEFAULT NULL,

`description` longtext,

`user` int(11) DEFAULT NULL,

`post\_date` datetime NOT NULL,

`postlikecount` int(11) DEFAULT NULL,

PRIMARY KEY (`idposts`),

KEY `post\_user\_idx` (`user`),

CONSTRAINT `post\_user` FOREIGN KEY (`user`) REFERENCES `users` (`id`) ON DELETE CASCADE ON UPDATE CASCADE

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='Contatins post of the users ';

Favourites table contains the likes of the user on certain posts . based on the type of Favourite we can check wether the faourites are for posts or comments . this is not linked to the table by any primary key.

CREATE TABLE `favourites` (

`idfavourites` int(11) NOT NULL,

`favourite\_type` enum('post','comment') DEFAULT NULL,

`contnet\_id` int(11) DEFAULT NULL,

PRIMARY KEY (`idfavourites`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='Contains the faviourate or likes for comments and posts';

Comments table contains the comments on posts made by the user . and is lined to the customer and the post by forigen keys comment\_user and post\_of\_comment.

CREATE TABLE `comments` (

`idcomments` int(11) NOT NULL AUTO\_INCREMENT,

`comment` varchar(45) DEFAULT NULL,

`post` int(10) DEFAULT NULL,

`comentcount` int(11) DEFAULT NULL,

`commentdate` datetime DEFAULT NULL,

`commentlikecount` int(11) DEFAULT NULL,

`commentuseremail` varchar(45) DEFAULT NULL,

`commentuser` int(11) DEFAULT NULL,

PRIMARY KEY (`idcomments`),

UNIQUE KEY `idcomments\_UNIQUE` (`idcomments`),

KEY `comment\_user\_idx` (`commentuser`),

KEY `post\_related\_to\_commnet\_idx` (`post`),

CONSTRAINT `comment\_user` FOREIGN KEY (`commentuser`) REFERENCES `users` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION,

CONSTRAINT `post\_related\_to\_commnet` FOREIGN KEY (`post`) REFERENCES `posts` (`user`) ON DELETE NO ACTION ON UPDATE NO ACTION

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='has the comments of all the user per post , comments have one to many relationship with posts ';

This is table contains all the comments made for each post . there is chance that there are more the one comment by more than ne user on one post and that needs to be show this will be help full to fastly retrive data from the two tables.

CREATE TABLE `comment\_posts` (

`commetcount` int(11) DEFAULT NULL,

`comment\_post\_id` int(10) DEFAULT NULL,

`comment\_id` int(11) DEFAULT NULL,

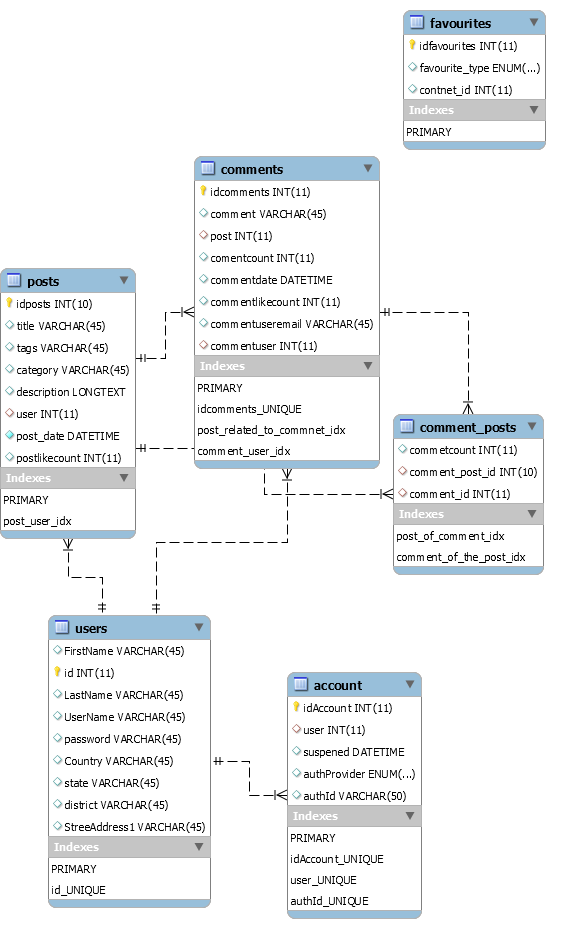
KEY `post\_of\_comment\_idx` (`comment\_post\_id`),

KEY `comment\_of\_the\_post\_idx` (`comment\_id`),

CONSTRAINT `comment\_of\_the\_post` FOREIGN KEY (`comment\_id`) REFERENCES `comments` (`idcomments`) ON DELETE NO ACTION ON UPDATE NO ACTION,

CONSTRAINT `post\_of\_comment` FOREIGN KEY (`comment\_post\_id`) REFERENCES `posts` (`user`) ON DELETE NO ACTION ON UPDATE NO ACTION

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='Posts and linked comments ';

  
  
Assignment 2:

1.What should be done to make users pair insertion unique i.e. to avoid duplicate user relationship creation?

Create a composite primary key for the columns user\_one\_id and user\_two\_id

2.What will be the insert query to insert a new Friend request sent by user 1 to user 2?

INSERT INTO `dummy\_Schema`.relationship`(`user\_one\_id`,` user\_two\_id `,`status`,`action\_user\_id`)VALUES (1,2,0,1);

3.What will be the query to update the status of the friend requesti.e. accepting friend request sent to user 2 by user 1?

INSERT INTO `dummy\_Schema`.relationship`(`user\_one\_id`,` user\_two\_id `,`status`,`action\_user\_id`)VALUES (1,2,1,2);

4.What will be the query to check if any two users are friends?

Select status from `dummy\_Schema`.relationship` where user\_one\_id = 1 and user\_two\_id = 2 and status =1

5.What will be the query to retrieve all the users’ friends? Here user 1 is the logged in user

.Select \* From `dummy\_Schema`.relationship` where user\_one\_id = 1 and status =1

6.What will be query to get the entire pending user request for the user from other users?Here user 1 is logged in.

Select \* From `dummy\_Schema`.relationship` where user\_one\_id = 1 and status =0

7.What will be the query to retrieve the friend request status when the logged in user visits the profile of another user?Here, user 1 is the logged in user. User 1 visits the profile of user7

Select status from `dummy\_Schema`.relationship` where user\_one\_id = 1 and user\_two\_id = 7 ;

.