

# IMPRESSIONS

## Project Part 1

**Samiksha Goyal   N18819733**

**Kunal Relia        N14724679**



## Introduction : Impressions

Impressions is a free web application that requires registration to use. Users can upload, save and manage images—known as pins through collections known as pinboards. Impressions acts as a personalized media platform. Users can browse the content of others on the main page. Users can then save individual pins to one of their own boards using the "Pin It" button, with Pinboards typically organized by a central topic or theme. Users can personalize their experience with Pinterest by pinning items, creating boards, and interacting with other members.

Content can also be found outside of Pinterest and similarly uploaded to a board via the "Pin It" button. An account can be created and accessed by providing username and email.

Quick links to Pinterest include the "pin it" button that can be added to the bookmark bar of a web browser, "Follow me" and "Pin it" buttons added to personal .Users should be aware of certain terms and functions when using Pinterest.

A "**pinboard**" is where the user's pins are located. Users can have several boards for various items such as quotes, travel or, most popularly, weddings.

A "**pin**" is an image that has either been uploaded or linked from a website. Once users create boards and add pins, other users can now *repin*, meaning they can *pin* one user's image to their board as well. Once the user has set up their account and boards, they can browse, comment, and like other *pins*. If a user wants to turn an image online into a "pin," there are a few simple steps to do so.

First, the user must select the image to pin. Then, the user enters an image URL into the link box. Next, the user selects the exact image they want to pin, and place it on the designated board.

Other users can now click on the pin to see which board the image is pinned in, who pinned the image previously, where the original pin is from, and who has liked, commented, or repinned the image.

Users should also keep in mind that Pinterest stores actual copies (not just thumbnails and links) of the images being pinned.

The images users pin, define their identities and boards they view. Pinterest then modifies a user's homepage to reflect toward his or her interest.

A basic summary of key features are :

Sign up: User can sign up by providing username and email, and set a password. He can then can login using the same.

Profile: User can create his profile and edit and update it.

**Friends:** user can send and receive friend requests which facilitates another level of privacy.

**PinBoards:** A board is where you collect your pins. For example, you can have a board named Food where you can pin all your food related pictures. By default a board is public.

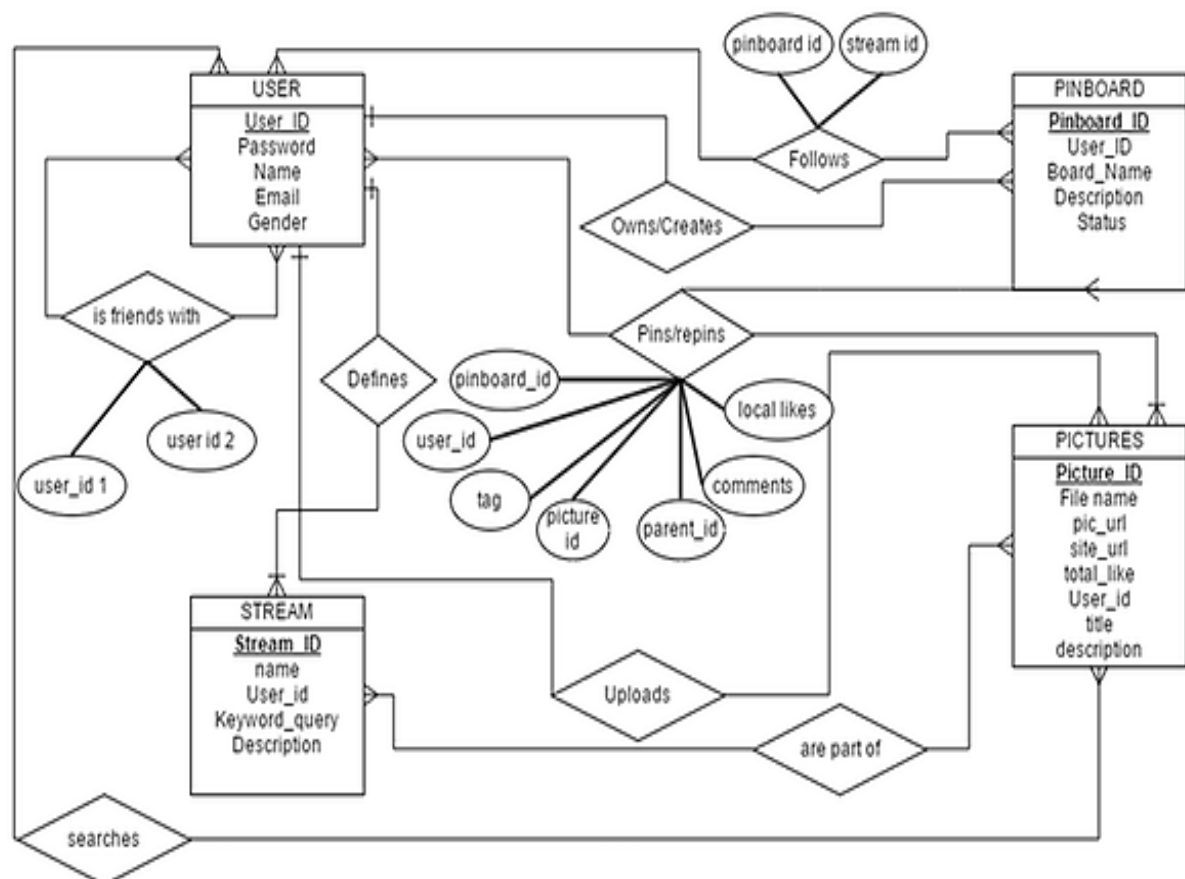
**Tags:** user can add tags or keywords to facilitate key word searching for his pins.

**Follow:** A user can follow another user or one or more of his boards.

**Follow Stream:** User can define one or more streams of his interest which will consolidate All the pictures that he chooses from the boards that he follows or pictures that the system will find him for his selected keywords.

**Search:** User can search for pins or boards or users by providing a keyword.

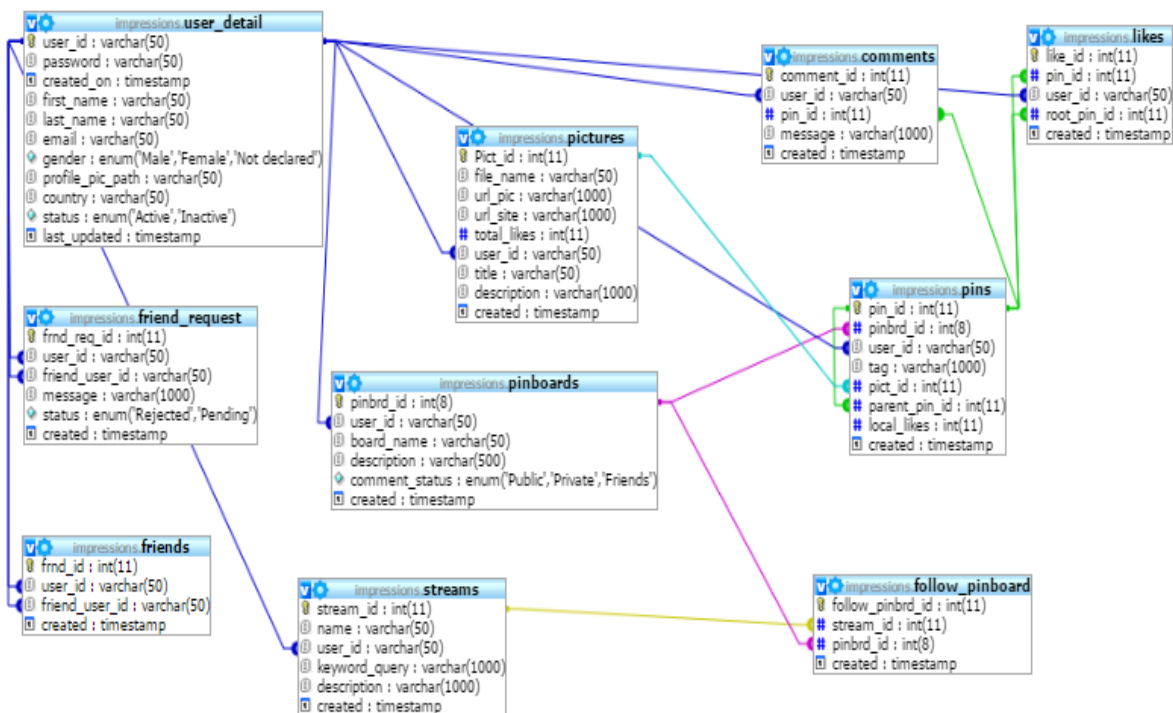
## ER Diagram :



## Assumptions :

- Two users are friends only when one user accepts to be friends with the other user .
- A stream can be defined by only one user whereas one user can define multiple streams.
- The concept of tags is implemented via the description field as well as tag field .

## Relational Schema :



## SQL Queries :

```
CREATE TABLE `user_detail` (
  `user_id` varchar(50) NOT NULL,
  `password` varchar(50) DEFAULT NULL,
  `created_on` timestamp NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  `first_name` varchar(50) DEFAULT NULL,
  `last_name` varchar(50) DEFAULT NULL,
  `email` varchar(50) DEFAULT NULL,
  `gender` enum('Male','Female','Not declared') DEFAULT NULL,
  `profile_pic_path` varchar(50) DEFAULT NULL,
  `country` varchar(50) DEFAULT NULL,
  `status` enum('Active','Inactive') DEFAULT NULL,
  `last_updated` timestamp NULL,
  PRIMARY KEY (`user_id`)
)
```

insert into

```
`user_detail`(`user_id`,`password`,`created_on`,`first_name`,`last_name`,`email`,`gender`,`profile_pic_path`,`country`,`status`,`last_updated`) values ('KunalRelia', 'kunalrelia91','2015-04-14 19:05:16','Kunal','Relia','kbr263@nyu.edu','Male',NULL, 'USA','Active', '2015-04-14 22:58:40');
```

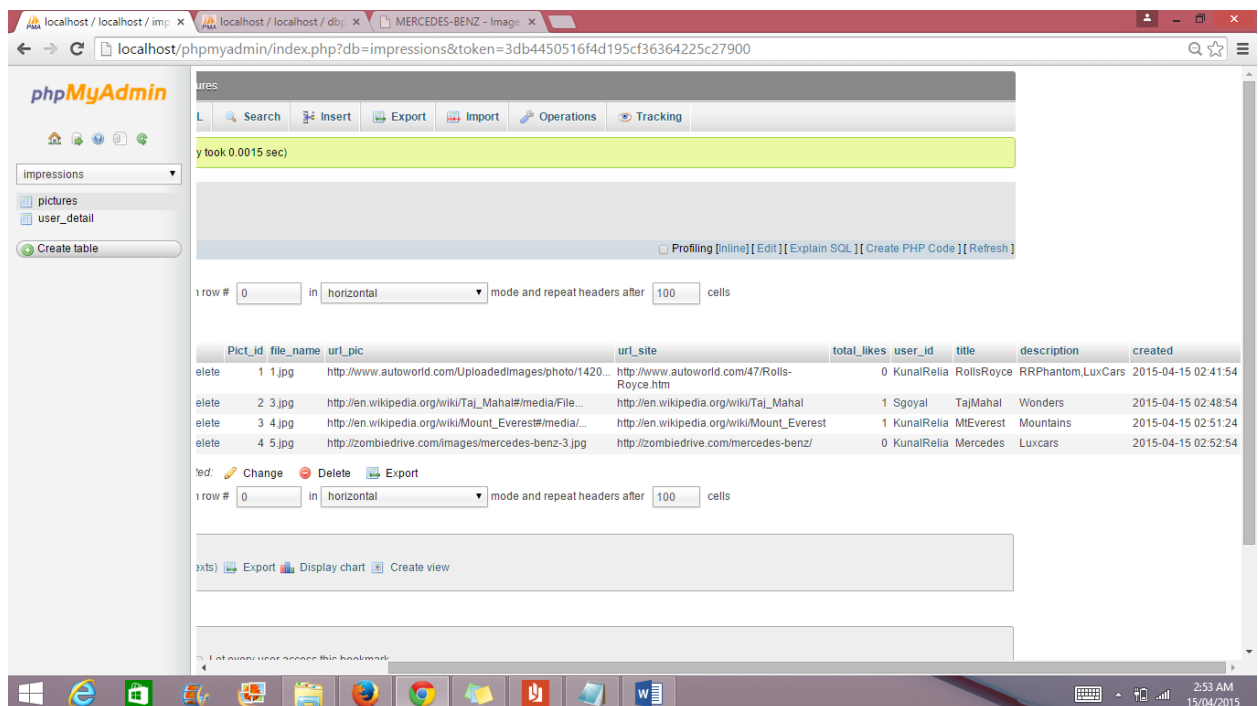
The screenshot shows the phpMyAdmin interface for the 'impressions' database. The 'user\_detail' table is selected, and the SQL query 'SELECT \* FROM `user\_detail` LIMIT 0, 30' is executed. The results show 4 rows of data:

user_id	password	created_on	first_name	last_name	email	gender	profile_pic_path	country	status	last_updated
kk	kk	2015-04-14 19:10:36	Kratika	Kasliwal	kk@nyu.edu	Female	NULL	USA	Active	2015-04-14 22:54:20
kunalrelia91	kunalrelia91	2015-04-14 19:05:16	Kunal	Relia	kbr263@nyu.edu	Male	NULL	USA	Active	2015-04-14 22:58:40
nin212	nin212	2015-04-14 19:15:36	Naimesh	Narsinghani	nin212@nyu.edu	Male	NULL	USA	Active	2015-04-14 22:59:48
sgoyal	sgoyal	2015-04-14 19:08:26	Samiksha	Goyal	sg@nyu.edu	Female	NULL	USA	Active	2015-04-14 22:59:20

```
CREATE TABLE `pictures` (
  `Pict_id` int(11) NOT NULL AUTO_INCREMENT,
  `file_name` varchar(50) DEFAULT NULL,
  `url_pic` varchar(1000) DEFAULT NULL,
  `url_site` varchar(1000) DEFAULT NULL,
  `total_likes` int(11) DEFAULT '0',
  `user_id` varchar(50) DEFAULT NULL,
  `title` varchar(50) DEFAULT NULL,
  `description` varchar(1000) DEFAULT NULL,
  `created` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`pict_id`),
  KEY `user_id` (`user_id`),
  CONSTRAINT `pic_fk_1` FOREIGN KEY (`user_id`) REFERENCES `user_detail` (`user_id`) ON UPDATE
  CASCADE
)
```

insert into

```
`pictures`(`pict_id`,`file_name`,`url_pic`,`url_site`,`total_likes`,`user_id`,`title`,`description`,`created`) values
(1,'1.jpg','http://www.autoworld.com/UploadedImages/photo/1420639359.jpg','http://www.autoworld.com/47/Rolls-Royce.htm',0,'KunalRelia','RollsRoyce','RRPhantom,LuxCars','2015-04-15 02:41:54');
```



```
CREATE TABLE `pinboards` (
  `pinbrd_id` int(8) NOT NULL AUTO_INCREMENT,
  `user_id` varchar(50) DEFAULT NULL,
  `board_name` varchar(50) DEFAULT NULL,
  `description` varchar(500) DEFAULT NULL,
  `comment_status` enum('Public','Private','Friends') NOT NULL,
  `created` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`pinbrd_id`),
  KEY `user_id` (`user_id`),
  CONSTRAINT `pinbrd_fk_1` FOREIGN KEY (`user_id`) REFERENCES `user_detail` (`user_id`) ON UPDATE CASCADE
)
```

```
insert into `pinboards`(`pinbrd_id`,`user_id`,`board_name`,`description`,`comment_status`,`created`)
values (1,'KunalRelia','LuxCars','My dream Cars','Public','2015-04-15 03:05:52');
```

The screenshot shows the phpMyAdmin web interface. The left sidebar lists databases: 'impressions', 'pictures', 'pinboards', and 'user\_detail'. The 'pinboards' table is selected. The main panel shows the table structure and data. A SQL query is entered: `SELECT * FROM pinboards LIMIT 0, 30`. The results show 3 rows. Below the table, there are options to 'Check All / Uncheck All With selected', 'Change', 'Delete', and 'Export'. At the bottom, there is a 'Bookmark this SQL query' section with a label input field and a checkbox 'Let every user access this bookmark'.

pinbrd_id	user_id	board_name	description	comment_status	created
1	KunalRelia	LuxCars	My dream Cars	Public	2015-04-15 03:05:52
2	Sgoyal	PlacesToVisit	My dream places that I will visit once	Public	2015-04-15 03:10:25
3	KunalRelia	DreamVacation	I wanna climb Everest!!!	Private	2015-04-15 03:12:06

```
CREATE TABLE `pins` (
  `pin_id` int(11) NOT NULL AUTO_INCREMENT,
  `pinbrd_id` int(8) DEFAULT NULL,
  `user_id` varchar(50) DEFAULT NULL,
  `tag` varchar(1000) DEFAULT NULL,
  `pict_id` int(11) DEFAULT NULL,
  `parent_pin_id` int(11) DEFAULT NULL,
  `local_likes` int(11) DEFAULT '0',
  `created` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`pin_id`),
  KEY `user_id` (`user_id`),
  KEY `pins_fk_1` (`pict_id`),
  KEY `pins_fk_2` (`pinbrd_id`),
  KEY `pins_fk_3` (`parent_pin_id`),
  CONSTRAINT `pins_fk_2` FOREIGN KEY (`pinbrd_id`) REFERENCES `pinboards` (`pinbrd_id`) ON DELETE
  CASCADE,
  CONSTRAINT `user_id` FOREIGN KEY (`user_id`) REFERENCES `user_detail` (`user_id`) ON UPDATE
  CASCADE,
  CONSTRAINT `pins_fk_3` FOREIGN KEY (`parent_pin_id`) REFERENCES `pins` (`pin_id`) ON DELETE
  CASCADE,
  CONSTRAINT `pins_fk_1` FOREIGN KEY (`pict_id`) REFERENCES `pictures` (`Pict_id`) ON DELETE CASCADE
)
```

```
insert into `pins`(`pin_id`,`pinbrd_id`,`user_id`,`tag`,`pict_id`,`parent_pin_id`,`local_likes`,`created`) values
(1,1,'KunalRelia','LuxCars',1,1,1,'2015-04-15 03:36:29');
```

The screenshot shows the phpMyAdmin interface for a database named 'impressions'. The 'pins' table is selected, and the SQL query 'SELECT \* FROM `pins` LIMIT 0, 30' is executed. The table structure is as follows:

pin_id	pinbrd_id	user_id	tag	pict_id	parent_pin_id	local_likes	created
1	1	KunalRelia	LuxCars	1	1	1	2015-04-15 03:36:29
2	3	KunalRelia	PlacesToVisit	3	2	1	2015-04-15 03:42:44
3	2	Sgoyal	PlacesToVisit	2	3	1	2015-04-15 03:42:29

The interface also shows options for sorting, displaying, and exporting the query results.



```
CREATE TABLE `streams` (
  `stream_id` int(11) NOT NULL AUTO_INCREMENT,
  `name` varchar(50) DEFAULT NULL,
  `user_id` varchar(50) DEFAULT NULL,
  `keyword_query` varchar(1000) DEFAULT NULL,
  `description` varchar(1000) DEFAULT NULL,
  `created` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`stream_id`),
  KEY `user_id` (`user_id`),
  CONSTRAINT `streams_fk_1` FOREIGN KEY (`user_id`) REFERENCES `user_detail` (`user_id`) ON UPDATE CASCADE
)
```

```
insert into `streams`(`stream_id`,`name`,`user_id`,`keyword_query`,`description`,`created`) values
(1,'Cars','KunalRelia','Cars,Luxury Cars','All the cars people dream to own!','2015-04-15 03:47:26');
```

The screenshot shows the phpMyAdmin web interface. The left sidebar lists databases: impressions, pictures, pinboards, pins, streams, and user\_detail. The 'streams' table is selected under the 'impressions' database. The main panel shows the table structure and data. The SQL query executed is 'SELECT \* FROM `streams` LIMIT 0, 30'. The table has 6 columns: stream\_id, name, user\_id, keyword\_query, description, and created. There are 3 rows of data displayed.

stream_id	name	user_id	keyword_query	description	created
1	Cars	KunalRelia	Cars,Luxury Cars	All the cars people dream to own!	2015-04-15 03:52:24
2	Mountains	KunalRelia	Mountains,Hills	All the mountain terrain	2015-04-15 03:50:26
3	Monuments	SGoyal	Wonders,Monuments,Agra	The most famous Wonder of World	2015-04-15 03:51:26

```
CREATE TABLE `friends` (
  `frnd_id` int(11) NOT NULL AUTO_INCREMENT,
  `user_id` varchar(50) DEFAULT NULL,
  `friend_user_id` varchar(50) DEFAULT NULL,
  `created` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`frnd_id`),
  KEY `user_id` (`user_id`),
  KEY `friend_user_id` (`friend_user_id`),
  CONSTRAINT `friends_fk_1` FOREIGN KEY (`user_id`) REFERENCES `user_detail` (`user_id`) ON UPDATE CASCADE,
  CONSTRAINT `friends_fk_2` FOREIGN KEY (`friend_user_id`) REFERENCES `user_detail` (`user_id`) ON UPDATE CASCADE
)
```

```
insert into `friends`(`frnd_id`,`user_id`,`friend_user_id`,`created`) values (1,'KunalRelia','Sgoyal','2015-04-15 03:57:21');
```

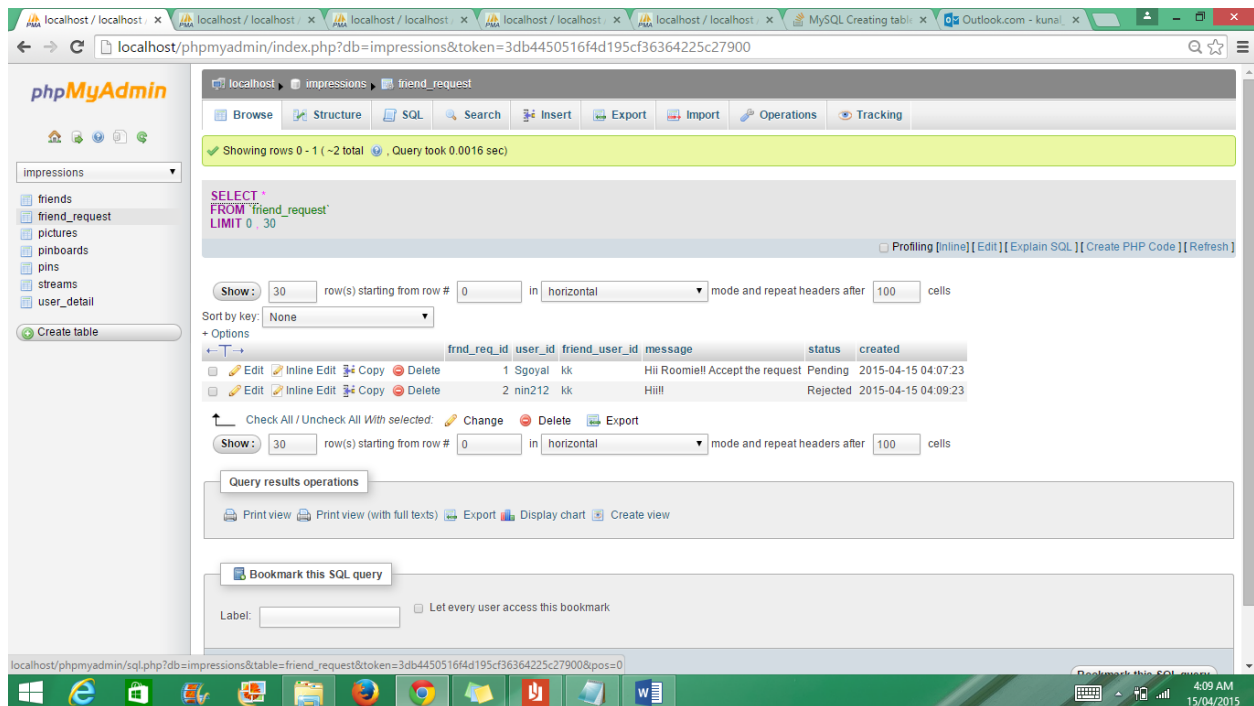
The screenshot shows the phpMyAdmin interface for a database named 'impressions'. The 'friends' table is selected, and the SQL query 'SELECT \* FROM friends LIMIT 0, 30' is executed. The results show 4 rows of data. The table structure is as follows:

frnd_id	user_id	friend_user_id	created
1	KunalRelia	Sgoyal	2015-04-15 03:57:21
2	KunalRelia	nin212	2015-04-15 03:57:21
3	kk	KunalRelia	2015-04-15 03:57:21
4	nin212	Sgoyal	2015-04-15 03:57:21

The interface also shows options for sorting, displaying, and exporting the data. The status bar at the bottom indicates the time as 3:59 AM on 15/04/2015.

```
CREATE TABLE `friend_request` (
  `frnd_req_id` int(11) NOT NULL AUTO_INCREMENT,
  `user_id` varchar(50) DEFAULT NULL,
  `friend_user_id` varchar(50) DEFAULT NULL,
  `message` varchar(1000) DEFAULT NULL,
  `status` enum('Rejected','Pending') DEFAULT 'Pending',
  `created` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`frnd_req_id`),
  KEY `user_id` (`user_id`),
  KEY `friend_user_id` (`friend_user_id`),
  CONSTRAINT `friend_request_fk_1` FOREIGN KEY (`user_id`) REFERENCES `user_detail` (`user_id`) ON
UPDATE CASCADE,
  CONSTRAINT `friend_request_fk_2` FOREIGN KEY (`friend_user_id`) REFERENCES `user_detail` (`user_id`)
ON UPDATE CASCADE
)
```

```
insert into `friend_request`(`frnd_req_id`,`user_id`,`friend_user_id`,`message`,`status`,`created`) values (1
,'Sgoyal','kk','Hii Roomie!! Accept the request','Pending','2015-04-15 04:07:23');
```



```
CREATE TABLE `follow_pinboard` (
  `follow_pinbrd_id` int(11) NOT NULL AUTO_INCREMENT,
  `stream_id` int(11) DEFAULT NULL,
  `pinbrd_id` int(8) DEFAULT NULL,
  `created` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`follow_pinbrd_id`),
  KEY `follow_pinboard_fk_2` (`stream_id`),
  KEY `follow_pinboard_fk_3` (`pinbrd_id`),
  CONSTRAINT `follow_pinboard_fk_2` FOREIGN KEY (`stream_id`) REFERENCES `streams` (`stream_id`) ON
DELETE CASCADE,
  CONSTRAINT `follow_pinboard_fk_3` FOREIGN KEY (`pinbrd_id`) REFERENCES `pinboards` (`pinbrd_id`) ON
DELETE CASCADE
)
```

```
insert into `follow_pinboard`(`follow_pinbrd_id`,`stream_id`,`pinbrd_id`,`created`) values (1,1,1,'2015-04-15 04:22:15');
```

The screenshot shows the phpMyAdmin interface for the 'impressions' database. The 'follow\_pinboard' table is selected, and the SQL query 'SELECT \* FROM follow\_pinboard LIMIT 0, 30' is executed. The results are displayed in a table with 4 columns: follow\_pinbrd\_id, stream\_id, pinbrd\_id, and created. There are 3 rows of data.

follow_pinbrd_id	stream_id	pinbrd_id	created
1	1	1	2015-04-15 04:22:15
2	2	3	2015-04-15 04:23:15
3	3	2	2015-04-15 04:23:56

```
CREATE TABLE `comments` (
  `comment_id` int(11) NOT NULL AUTO_INCREMENT,
  `user_id` varchar(50) DEFAULT NULL,
  `pin_id` int(11) DEFAULT NULL,
  `message` varchar(1000) DEFAULT NULL,
  `created` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`comment_id`),
  KEY `user_id` (`user_id`),
  KEY `comments_fk_2` (`pin_id`),
  CONSTRAINT `comments_fk_2` FOREIGN KEY (`pin_id`) REFERENCES `pins` (`pin_id`) ON DELETE CASCADE,
  CONSTRAINT `comments_fk_3` FOREIGN KEY (`user_id`) REFERENCES `user_detail` (`user_id`) ON UPDATE CASCADE
)
```

```
INSERT INTO `impressions`.`comments` (`comment_id`, `user_id`, `pin_id`, `message`, `created`) VALUES
('1', 'KunalRelia', '1', 'My Dream!!!', '2015-04-15 04:28:17');
```

The screenshot shows the phpMyAdmin web interface. The left sidebar lists the database structure, including tables like `comments`, `follow\_pinboard`, `friends`, etc. The main panel displays the 'comments' table with 4 rows of data. The SQL query 'SELECT \* FROM `comments` LIMIT 0, 30' is executed, and the results are shown in a table format. The table has columns: `comment\_id`, `user\_id`, `pin\_id`, `message`, and `created`.

comment_id	user_id	pin_id	message	created
1	KunalRelia	1	My Dream!!!	2015-04-15 04:28:17
2	KunalRelia	2	My Biggest Dream!!!	2015-04-15 04:29:36
3	KunalRelia	3	Truly Awesome	2015-04-15 04:30:00
4	kk	1	Buy 1!	2015-04-15 04:31:02

```
CREATE TABLE `likes` (
  `like_id` int(11) NOT NULL AUTO_INCREMENT,
  `pin_id` int(11) DEFAULT NULL,
  `user_id` varchar(50) DEFAULT NULL,
  `root_pin_id` int(11) DEFAULT NULL,
  `created` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`like_id`),
  KEY `user_id` (`user_id`),
  KEY `likes_fk_3` (`pin_id`),
  KEY `likes_fk_5` (`root_pin_id`),
  CONSTRAINT `likes_fk_3` FOREIGN KEY (`pin_id`) REFERENCES `pins` (`pin_id`) ON DELETE CASCADE,
  CONSTRAINT `likes_fk_4` FOREIGN KEY (`user_id`) REFERENCES `user_detail` (`user_id`) ON UPDATE CASCADE,
  CONSTRAINT `likes_fk_5` FOREIGN KEY (`root_pin_id`) REFERENCES `pins` (`pin_id`) ON DELETE CASCADE
)
```

```
insert into `likes`(`like_id`,`pin_id`,`user_id`,`root_pin_id`,`created`) values (1,1,'KunalRelia',1,'2015-04-15 04:33:49');
```

The screenshot shows the phpMyAdmin web interface. The left sidebar lists databases like 'impressions', 'comments', 'follow\_pinboard', etc. The main panel shows the 'likes' table with 4 rows. The SQL query 'SELECT \* FROM `likes` LIMIT 0, 30' is entered in the query box. Below the query, the table data is displayed with columns: like\_id, pin\_id, user\_id, root\_pin\_id, and created. The data rows are: (1, 1, KunalRelia, 1, 2015-04-15 04:33:49), (2, 2, KunalRelia, 2, 2015-04-15 04:35:49), (3, 3, KunalRelia, 3, 2015-04-15 04:36:49), and (4, 3, Sgoyal, 3, 2015-04-15 04:37:49). The interface includes options for sorting, displaying, and exporting the data.

like_id	pin_id	user_id	root_pin_id	created
1	1	KunalRelia	1	2015-04-15 04:33:49
2	2	KunalRelia	2	2015-04-15 04:35:49
3	3	KunalRelia	3	2015-04-15 04:36:49
4	3	Sgoyal	3	2015-04-15 04:37:49




















## Few Other Queries :

### 1) Signing up

Here, the user is expected to fill out few of the details like Username, Password, Name, E-Mail ID, Country, etc. Once, the correct details are filled out, the user will be entered into the database using the below given query.

insert into user\_detail

(`user\_id`,`password`,`created\_on`,`first\_name`,`last\_name`,`email`,`gender`,`profile\_pic\_path`,`country`,`status`,`last\_updated`) values ('JayPatel','jp12345','2015-04-15 04:54:21','Jay','Patel','jp@nyu.edu','Male',NULL,'USA','Active','2015-04-15 04:54:21');

	user_id	password	created_on	first_name	last_name	email	gender	profile_pic_path	country	status	last_updated
   	JayPatel	jp12345	2015-04-15 04:54:21	Jay	Patel	jp@nyu.edu	Male	NULL	USA	Active	2015-04-15 04:54:21
   	kk	kk	2015-04-14 19:10:36	Kratika	Kasliwal	kk@nyu.edu	Female	NULL	USA	Active	2015-04-14 22:54:20
   	KunalRelia	kunalrelia91	2015-04-14 19:05:16	Kunal	Relia	kbr263@nyu.edu	Male	NULL	USA	Active	2015-04-14 22:58:40
   	nin212	nin212	2015-04-14 19:15:36	Naimesh	Narsinghani	nin212@nyu.edu	Male	NULL	USA	Active	2015-04-14 22:59:48
   	Sgoyal	sgoyal	2015-04-14 19:08:26	Samiksha	Goyal	sg@nyu.edu	Female	NULL	USA	Active	2015-04-14 22:59:20

### 2) Pinning on boards

The registered user is expected to upload a picture or provide its URL to pin a picture to his/her board.

insert into

`pictures`(`pict\_id`,`file\_name`,`url\_pic`,`url\_site`,`total\_likes`,`user\_id`,`title`,`description`,`created`) values (5,'6.jpg',









[http://upload.wikimedia.org/wikipedia/commons/thumb/5/59/BMW\\_328i\\_F30\\_2012\\_vl\\_2.jpg/1024px-BMW\\_328i\\_F30\\_2012\\_vl\\_2.jpg](http://upload.wikimedia.org/wikipedia/commons/thumb/5/59/BMW_328i_F30_2012_vl_2.jpg/1024px-BMW_328i_F30_2012_vl_2.jpg),

<http://en.wikipedia.org/wiki/BMW>,'0','JayPatel','BMW','LuxCars','2015-04-15 05:21:34');

	Pict_id	file_name	url_pic	url_site	total_likes	user_id	title	description	created
delete	1	1.jpg	<a href="http://www.autoworld.com/UploadedImages/photo/1420...">http://www.autoworld.com/UploadedImages/photo/1420...</a>	<a href="http://www.autoworld.com/47/Rolls-Royce.htm">http://www.autoworld.com/47/Rolls-Royce.htm</a>	0	KunalRelia	RollsRoyce	RRPhantom,LuxCars	2015-04-15 02:41:54
delete	2	3.jpg	<a href="http://en.wikipedia.org/wiki/Taj_Mahal#/media/File...">http://en.wikipedia.org/wiki/Taj_Mahal#/media/File...</a>	<a href="http://en.wikipedia.org/wiki/Taj_Mahal">http://en.wikipedia.org/wiki/Taj_Mahal</a>	1	Sgoyal	TajMahal	Wonders	2015-04-15 02:48:54
delete	3	4.jpg	<a href="http://en.wikipedia.org/wiki/Mount_Everest#/media/...">http://en.wikipedia.org/wiki/Mount_Everest#/media/...</a>	<a href="http://en.wikipedia.org/wiki/Mount_Everest">http://en.wikipedia.org/wiki/Mount_Everest</a>	1	KunalRelia	MT Everest	Mountains	2015-04-15 02:51:24
delete	4	5.jpg	<a href="http://zombiedrive.com/images/mercedes-benz-3.jpg">http://zombiedrive.com/images/mercedes-benz-3.jpg</a>	<a href="http://zombiedrive.com/mercedes-benz/">http://zombiedrive.com/mercedes-benz/</a>	0	KunalRelia	Mercedes	Luxcars	2015-04-15 02:52:54
delete	5	6.jpg	<a href="http://upload.wikimedia.org/wikipedia/commons/thu...">http://upload.wikimedia.org/wikipedia/commons/thu...</a>	<a href="http://en.wikipedia.org/wiki/BMW">http://en.wikipedia.org/wiki/BMW</a>	0	JayPatel	BMW	LuxCars	2015-04-15 05:21:34

insert into `pins`

(`pin\_id`,`pinbrd\_id`,`user\_id`,`tag`,`pict\_id`,`parent\_pin\_id`,`local\_likes`,`created`) values (5,4,'JayPatel','Carrsss',5,5,0,'2015-04-15 05:24:53');

   	4	4	JayPatel	Carrsss	1	1	0	2015-04-15 05:02:13
   	5	4	JayPatel	Carrsss	5	5	0	2015-04-15 05:24:53

Alternatively, an already pinned picture can be re-pinned. The query for re-pinning can be as shown below.

```
insert into `pins`
(`pin_id`,`pinbrd_id`,`user_id`,`tag`,`pict_id`,`parent_pin_id`,`local_likes`,`created`) values
(4,4,'JayPatel','Carrsss',1,1,0,'2015-04-15 05:02:13');
```

					pin_id	pinbrd_id	user_id	tag	pict_id	parent_pin_id	local_likes	created
<input type="checkbox"/>	Edit	Inline Edit	Copy	Delete	1	1	KunalRelia	LuxCars	1	1	1	2015-04-15 03:36:29
<input type="checkbox"/>	Edit	Inline Edit	Copy	Delete	2	3	KunalRelia	PlacesToVisit	3	2	1	2015-04-15 03:42:44
<input type="checkbox"/>	Edit	Inline Edit	Copy	Delete	3	2	Sgoval	PlacestoVisit	2	3	1	2015-04-15 03:42:29
<input type="checkbox"/>	Edit	Inline Edit	Copy	Delete	4	4	JayPatel	Carrsss	1	1	0	2015-04-15 05:02:13

- 3) A user can become friends with other people. Hence the query that can be used to send a friend request will be like

```
insert into `friend_request` (`frnd_req_id`,`user_id`,`friend_user_id`,`message`,`status`,`created`)
values (3,'JayPatel','kk','Hii please Accept the request','Pending','2015-04-15 05:07:23');
```

					frnd_req_id	user_id	friend_user_id	message	status	created
<input type="checkbox"/>	Edit	Inline Edit	Copy	Delete	1	Sgoval	kk	Hii Roomie!! Accept the request	Pending	2015-04-15 04:07:23
<input type="checkbox"/>	Edit	Inline Edit	Copy	Delete	2	nin212	kk	Hii!!	Rejected	2015-04-15 04:09:23
<input type="checkbox"/>	Edit	Inline Edit	Copy	Delete	3	JayPatel	kk	Hii please Accept the request	Pending	2015-04-15 05:07:23

- 4) A user can accept or reject a received request. Let us consider a case where in the request is accepted.

Select \* from `friend\_request` where friend\_user\_id = 'kk' AND status = 'Pending'

					frnd_req_id	user_id	friend_user_id	message	status	created
<input type="checkbox"/>	Edit	Inline Edit	Copy	Delete	1	Sgoval	kk	Hii Roomie!! Accept the request	Pending	2015-04-15 04:07:23
<input type="checkbox"/>	Edit	Inline Edit	Copy	Delete	3	JayPatel	kk	Hii please Accept the request	Pending	2015-04-15 05:07:23

Now, if kk becomes friends with JayPatel, then the following query will be executed.

```
insert into `friends`(`frnd_id`,`user_id`,`friend_user_id`,`created`) values (5,'kk','JayPatel','2015-04-15 05:13:39');
```



+ Options					frnd_id	user_id	friend_user_id	created
	Edit	Inline Edit	Copy	Delete	1	KunalRelia	Sgoyal	2015-04-15 03:57:21
	Edit	Inline Edit	Copy	Delete	2	KunalRelia	nin212	2015-04-15 03:57:21
	Edit	Inline Edit	Copy	Delete	3	kk	KunalRelia	2015-04-15 03:57:21
	Edit	Inline Edit	Copy	Delete	4	nin212	Sgoyal	2015-04-15 03:57:21
	Edit	Inline Edit	Copy	Delete	5	kk	JayPatel	2015-04-15 05:13:39

- 5) A user is also given the privilege of following a particular pinboard.

insert into `follow\_pinboard`(`follow\_pinbrd\_id`,`stream\_id`,`pinbrd\_id`,`created`) values (4,3,2,'2015-04-15 05:49:55');

	follow_pinbrd_id	stream_id	pinbrd_id	created
	1	1	1	2015-04-15 04:22:15
	2	2	3	2015-04-15 04:23:15
	3	3	2	2015-04-15 04:23:56
	4	3	2	2015-04-15 05:49:55

Selected: Change Delete Export

- 6) Further adding onto the functionality, a user can also like or comment as per the pin's privilege level.

insert into `likes`(`like\_id`,`pin\_id`,`user\_id`,`root\_pin\_id`,`created`) values (5,5,'JayPatel',5,'2015-04-15 05:54:44');

+ Options				like_id	pin_id	user_id	root_pin_id	created
<input type="checkbox"/>					1	1	KunalRelia	1 2015-04-15 04:33:49
<input type="checkbox"/>					2	2	KunalRelia	2 2015-04-15 04:35:49
<input type="checkbox"/>					3	3	KunalRelia	3 2015-04-15 04:36:49
<input type="checkbox"/>					4	3	Sgoyal	3 2015-04-15 04:37:49
<input type="checkbox"/>					5	5	JayPatel	5 2015-04-15 05:54:44

- 7) One of the last functionality is the advantage of searching streams using certain keywords.

Select \* from streams where keyword\_query like "%car%"

+ Options				stream_id	name	user_id	keyword_query	description	created
<input type="checkbox"/>					1	Cars	KunalRelia	Cars,Luxury Cars	All the cars people dream to own! 2015-04-15 03:52:24

Check All / Uncheck All With selected: Change Delete Export