DATABASE MANAGEMENT
SYSTEMS PROJECT

CREATING A BACK-END DATABASE FOR A SOCIAL NETWORKING WEBSITE

Developed by:

RHEEYA UPPAAL – 13070121133
ALIASGAR MURTUZUA – 13070121105

3rd Year, Computer Science

Table of Contents

Acknowledgements

It would be our utmost pleasure to express our gratitude to our Database Management System professor, Mrs. Shruti Patil in providing a helping hand and sincere guidance in this project. Her valuable guidance, support and supervision all through this project are responsible for attaining its present form.

1. Introduction

1.1 Overview and Problem Statement

The objective of creating this software is to provide a storage system of data for a social networking website. This allows the users and administrators of the website to keep a proper track of the relevant user information, and allows administrators to maintain the records. The software will be user appropriate, easy to use, provide easy recovery of errors and have an overall end user high subjective satisfaction. The system will store the information of users and features of the social networking website. The system can generate all kinds of reports as per the user's requirements.

1.1.1 Problem Statement:

This project is the backend database of a Social Networking Application designed to make users keep track of all their data, and application administrators maintain the data efficiently.

1.2 Purpose

The document will provide a detailed description of the requirements for the Social Networking database. This document will allow for a complete understanding of what is to be expected of the Database constructed. The clear understanding of the Database and its' functionality will allow for the correct software to be developed for the end user and will be used for the development of the future stages of the project. This document will provide the foundation for the project.

1.3 Scope

This project is the backend database of a social networking website. The application is designed to help people keep track of their personal accounts and interactions with other users, pages, forums, events and applications. The database was designed according to the functional requirements.

Social networking is a widely used application. This project can thus benefit any user and all the administrators of the application.

2. Project Design

2.1 Process Model

To solve actual problems in an industrial setting, a team of software engineers must incorporate a development strategy that encompasses the process, methods and tools used, and the generic phases of development. This strategy is often referred to as a *"process model"* or a *"software engineering paradigm".* A process model is selected based on the nature of the project, its application, the methods to be used, the controls and deliverables etc.

All software development can be characterized as a problem solving loop in which four distinct stages are encountered:

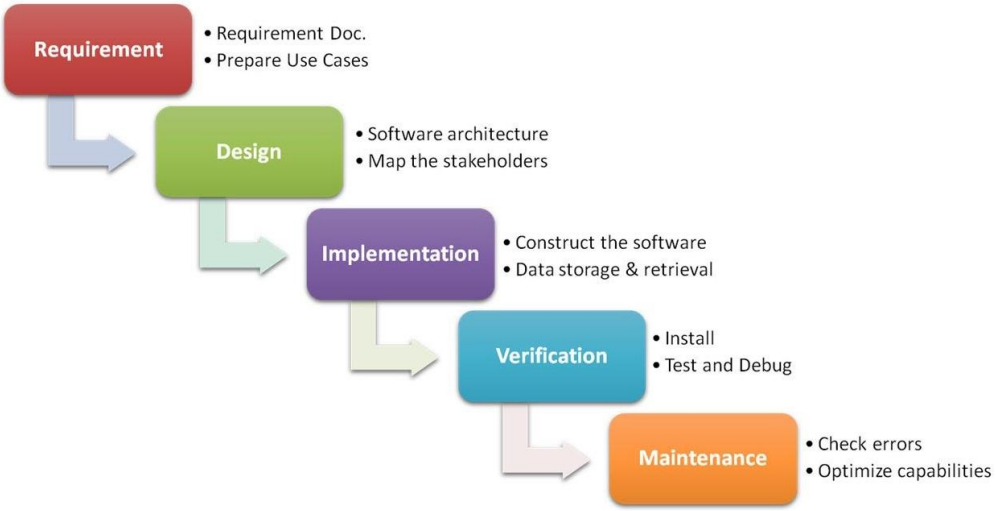
1. Status quo - The current state of affairs
2. Problem definition - Identifies the current problem to be solved
3. Technical Development - Solves the problem through the application of some technology
4. Solution Integration - Delivers the results to the internal or external customers

This problem solving loop applies to software engineering work at many different levels of resolution. These loops can also be recursive.

Realistically, it is difficult to compartmentalize activities as neatly because interdependency exists among the phases. However, this simplified view drives a point: regardless of the process model selected, the stages coexist simultaneously.

It has been suggested that a "chaos model" describes software development as a "continuum from the user to the developer to the technology." Software models attempt to bring order to an inherently chaotic activity.

The Waterfall Model has been used as a process model for this project.



Waterfall Process Model

The waterfall model is a sequential design process, used in software development processes, in which progress is seen as flowing steadily downwards (like a waterfall) through the phases of conception, initiation, analysis, design, construction, testing, implementation and maintainence.

2.2 Functional Requirements

Normal requirements consist of the objectives and goals that are stated during the meeting with the relevant customers. Normal requirements of our project are:

1. User friendly efficient and lucrative system
2. Minimum maintenance cost
3. Ease of operation
4. Measured coding
5. High efficiency

Expected requirements are implicit to the system and may be so fundamental that the customer does not state them. Their absence is a cause for dissatisfaction:

1. Development system with limited cost
2. Minimum hardware requirements
3. Efficient design of the entire system

The software shall implement the following functionalities:

1. The application must support multiple accounts.
2. No scope for human error
3. Data redundancy is avoided
4. Data is consistent
5. High security of data through multiple views
6. Easy updating of data

2.3 Non-functional Requirements

Nonfunctional requirements deal with the characteristics of the system which cannot be expressed as functions of the system, portability of the system, usability of the system, etc.

Nonfunctional requirements may include:

- Reliability issues
- Accuracy of results
- Human –Computer interface issues
- Constraints on the system implementation, etc.

Performance Characteristics include:

- The software shall accommodate 1 user at a time, thus preventing any inconsistencies.
- The software shall be accessible by the end user 24/7.
- The software shall not take more than 1 second to begin running.

Safety Requirements include:

- While the end user is accessing the software, no form of virus shall disrupt/harm the computer system.

2.4 Assumptions and Risks

A risk is a potential problem – it might happen or it might not. Two characteristics of risks:

1. Uncertainty – the risk may or may not happen, that is, there are no 100% risks (those, instead, are called constraints)
2. Loss – the risk becomes a reality and unwanted consequences or losses occur

Project risks:

- They threaten the project plan
- If they become real, it is likely that the project schedule will slip and that costs will increase

Technical risks

- They threaten the quality and timeliness of the software to be produced
- If they become real, implementation may become difficult or impossible

Business risks

- They threaten the viability of the software to be built
- If they become real, they jeopardize the project or the product

A risk table provides a project manager with a simple technique for risk projection. It consists of five columns:

- Risk Summary – short description of the risk
- Risk Category
- Probability – estimation of risk occurrence based on group input
- Impact – (1) catastrophic (2) critical (3) marginal (4) negligible
- RMMM – Pointer to a paragraph in the Risk Mitigation, Monitoring, and Management Plan

Risk Summary	Risk Category	Probability	Impact	RMMM
Power Failure	Known Risk	0.3	3	Use a back up power supply. Monitor the risk often.

System Crashed	Unpredictable	0.25	2	Keep a backup of all the data in the database as there can be easy retrieval of data
Software Corrupted	Unpredictable	0.2	2	Use an antivirus or apply cyber security measures and restore points

An effective strategy for dealing with risk must consider three issues (Note: these are not mutually exclusive i.e they can occur at the same time)

- Risk mitigation (i.e., avoidance)
- Risk monitoring
- Risk management and contingency planning

Risk mitigation (avoidance) is the primary strategy and is achieved through a plan:

- During risk monitoring, the project manager monitors factors that may provide an indication of whether a risk is becoming more or less likely
- Risk management and contingency planning assume that mitigation efforts have failed and that the risk has become a reality
- RMMM steps incur additional project cost

Risks can occur after the software has been delivered to the user.

3. Conceptual Design

3.1 Entities and Relationships

The data model of a project consists of three interrelated pieces of information:

- The data object (Entity)
- The attributes that describe the data object
- The relationships that connect data objects to one another

The data object description (DOD) incorporates the data objects and all of their attributes. The Entiry Relationship Diagram (ERD) enables a software engineer to identify data objects and their relationships using a graphical notation.

A data object can be an external entity (E.g. anything that produces or consumes information), a thing (e.g., a report or a display), an occurrence (e.g., a telephone call) or event (e.g., an alarm), a role (e.g., salesperson), an organizational unit (e.g., accounting department), a place (e.g., a warehouse), or a structure (e.g., a file). Data objects are related to one another.

Attributes define the properties of a data object. They can be used to name an instance of the data object, describe the instance or make reference to another instance in another table. For example, a car is defined in terms of make, model, ID number, body type, color and owner. The body of the table represents specific instances of the data object.

The relationships are always defined by the context of the problem that is being analyzed. A relationship denotes a specific connection between the entities. Relationship pairs are bidirectional.

The entities of the project are as follows:

1. USER
   - This describes the central idea of the social networking concept: A user of the application.

2. FRIEND
   - The user may interact with other users of the social networking site. These users are called ‘Friends’.
   - A user may have many or no friends.

3. POST
   - A post is an entity created by a user. The post contains any content the user wishes to share with other users.
   - It can be created on a user’s own profile, on a page, or in a forum.

- The existence of a post is optional.
- Every post can be Liked and Commented on by other users.
- A post may contain either textual content or media.

4. PAGE

- A page is an open platform where multiple users can share posts on similar ideas (the idea being the topic of that page).
- A page is run by one or many administrators, who control the page.
- Only an admin may post on a page. Users may only view or like these posts.
- A user may like none or many pages.

5. EVENT

- An event is created by a user.
- The user shares details on the event and then invites other users (Friends).
- The invited Friends are added to the 'Invited' list.
- The friends who accept the invitation are added to the 'Going' list.

6. FORUM

- A forum is a group for like minded users to interact through posts.
- A forum has administrators who can add and delete users from a forum.
- A forum is similar to a Page. However, unlike a page, any user may post in the Forum. Also, contents of the forum are only visible to members.

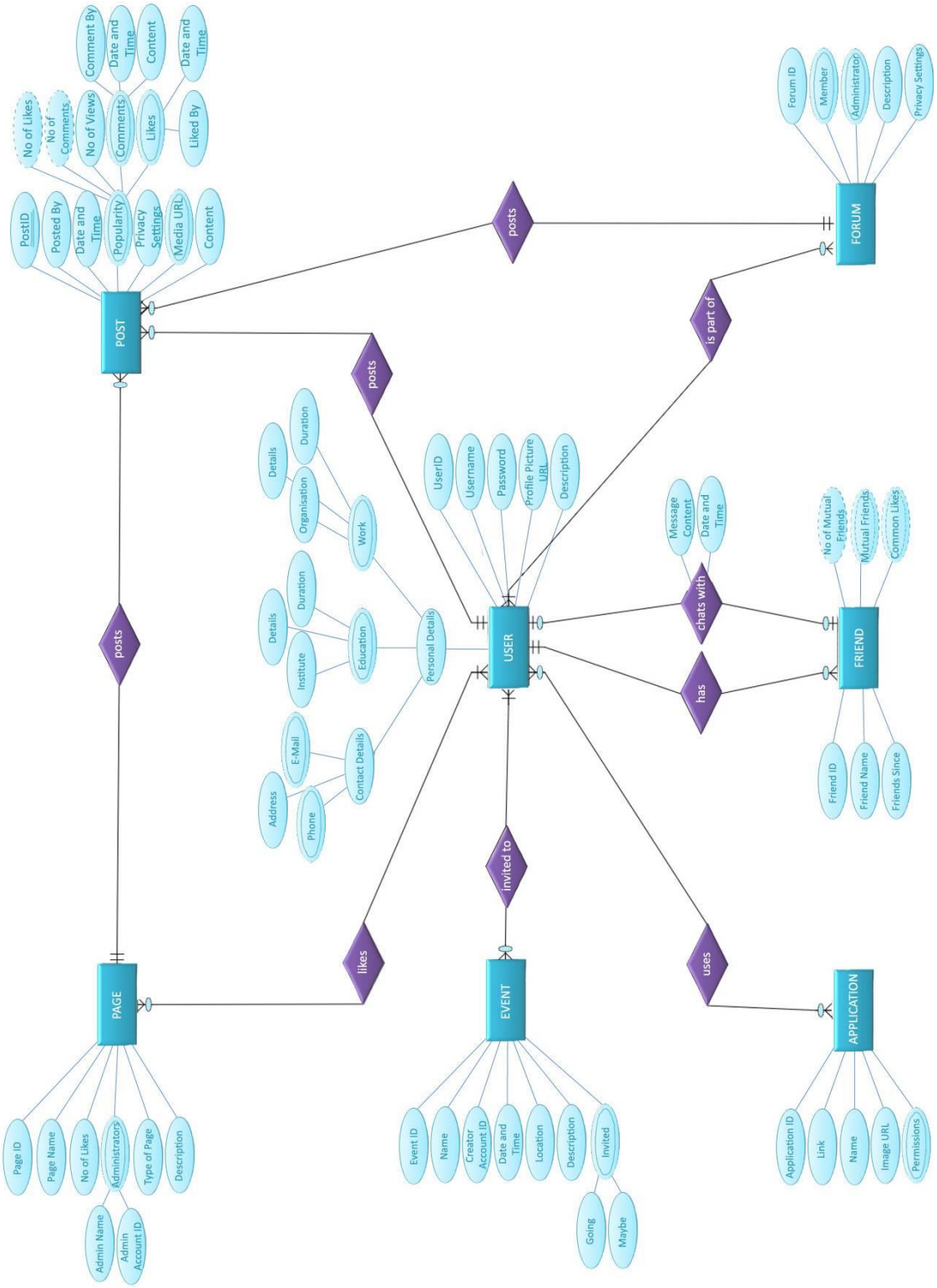
7. APPLICATION

- An application is not directly connected to the social networking website. It is run by a separate organization. Only links for the applications are mentioned on the social networking website.
- When a user selects a particular application, the user is directed to the application's website. The social networking website then has no control or responsibility for the user.

The entities are related to each other as follows:

- A USER has none or many FRIENDS
- A USER may chat with only one FRIEND at a time
- A USER is a part of none or many FORUMS
- A USER likes none or many PAGES
- A USER is invited to none or many EVENTS
- A USER uses none or many APPLICATIONS
- A USER, PAGE or FORUM may post none or many POSTS

3.2 Entity Relationship Diagram



4. Logical Design

4.1 Conversion of E-R Diagram into Relational Schema

ER diagrams are conceptual level models. They are based on the notion of real world entities and the relationships among them. They comprise of :

- Entities and their attributes
- Relationships among entities

The relational schema for the project's ER Diagram is as follows:

PAGE

PageID	Name	No of likes	Type of Page	Description

Admin

PageID	AdminUserID	AdminName

POST

PostID	Content	Date_and_Time	MediaURL	Privacy	PageID	UserID	ForumID

Popularity

PostID	No of Views

Comment

PostID	UserID	Date_and_Time	Content

Likes

PostID	UserID	Date_and_Time

USER

UserID	Name	Password	ProfilePictureURL	Description

ContactDetails

UserID	Address

Phone

UserID	PhoneNo

E-Mail

UserID	E-MailID

Education

UserID	Institute	Duration	Details

Work

UserID	Organisation	Duration	Details

FRIEND

FriendID	Name	Friends Since	UserID

FORUM

ForumID	Description	Privacy

Members

ForumID	MemberName	UserID

Administrators

ForumID	AdminName	UserID

EVENT

EventID	Name	CreatorUserID	Date_and_Time	Location	Description

Invited

EventID	UserID	Going	Invited

APPLICATION

ApplicationID	Link	Name	ImageURL

Permissions

ApplicationID	Permission

ChatsWith

UserID	FriendID	Content	Date_and_Time

Likes

UserID	PageID

InvitedTo

UserID	EventID

Uses

UserID	ApplicationID

IsPartOf

UserID	ForumID

4.2 Functional Dependencies

A functional dependency is an association between two attributes of the same relational database table. One of the attributes is called the determinant and the other attribute is called the determined. For each value of the determinant there is associated one and only one value of the determined.

The types of Functional Dependencies are:

1. Partial Dependency and Full Dependency.
2. Transitive and Non Transitive Dependency.
3. Single valued and Multivalued Dependency.
4. Trivial and Non Trivial Dependency.

Table	Attributes	Dependencies	Type of Dependency	Normalized Form
PAGE	**PageID**			
	Name	PageID -> Name	Full	
	No of likes	PageID ->Likes	Full	BCNF
	Type of Page	PageID -> Type	Full	
	Description	PageID -> Description	Full	
Admin	PageID			
	AdminUserID	PageID -> AdminUserID	Full	2NF
	AdminName	Page ID -> AdminName	Full	
		AdminID -> Admin Name	Transitive	
POST	**PostID**			
	Content	PostID -> Content	Partial	
	Date_and_Time	PostID -> Time	Partial	
	MediaURL	PostID -> Media	Partial	1NF
	Privacy Settings	PostID -> Settings	Partial	
	PageID			
	UserID			
	ForumID			
Popularity	PostID			BCNF
	No of Views	PostID -> Views	Full	
Comments	PostID			
	UserID	PostID -> UserID	Transitive	1NF
	Date_and_Time	PostID -> Time	Partial	
	Content	PostID -> Content	Partial	
Likes	PostID			
	UserID	PostID -> UserID	Transitive	1NF
	Date_and_Time	PostID -> Time	Partial	

USER	**UserID**			
	Name	UserID -> Name	Full	BCNF
	Password	UserID -> Password	Full	
	Profile Picture URL	UserID -> URL	Full	
	Description	UserID -> Description	Full	
ContactDetails	UserID			BCNF
	Address	UserID -> Address	Full	
Phone	UserID			BCNF
	PhoneNo	UserID -> PhoneNo	Full	
E-Mail	UserID			BCNF
	E-MailID	UserID -> Email	Full	
Education	UserID			3NF
	Institute			
	Duration	UserID, Institute -> Duration	Full	
	Details	UserID, Institute -> Details	Full	
Work	UserID			3NF
	Organisation			
	Duration	UserID, Organisation -> Duration	Full	
	Details	UserID, Organisation -> Details	Full	
FRIEND	**FriendID**			1NF
	Name	FriendID -> Name	Partial	
	Friends Since	FriendID, UserID -> UserID	Full	
	UserID			
FORUM	**ForumID**			BCNF
	Description	ForumID -> Description	Full	
	Privacy Settings	ForumID -> Settings	Full	
Members	ForumID			BCNF
	MemberName	ForumID, UserID -> Name	Full	
	UserID			
Administrators	ForumID			BCNF
	AdminName	ForumID, UserID -> AdminName	Full	
	UserID			

EVENT	**EventID**			
	Name	EventID -> Name	Partial	1NF
	CreatorUserID	EventID -> CreatorID	Transitive	
	Date_and_Time	EventID -> Time	Partial	
	Location	EventID -> Location	Partial	
	Description	EventID -> Description	Partial	
Invited	EventID UserID			BCNF
	Invited	EventID, UserID -> Invited	Full	
	Going	EventID, UserID -> Going	Full	
APPLICATION	**ApplicationID**			
	Link	ApplicationID -> Link	Full	BCNF
	Name	ApplicationID -> Name	Full	
	ImageURL	ApplicationID -> ImageURL	Full	
Permissions	ApplicationID Permission	ApplicationID -> Permissions	Full	BCNF
ChatsWith (User-Friend)	UserID FriendID Content Date_and_Time	UserID, FriendID, Time -> Content	Full	3NF
Likes (Page-User)	UserID PageID	No dependencies		BCNF
InvitedTo (User-Event)	UserID EventID	No dependencies		BCNF
Uses (Application-User)	UserID ApplicationID	No dependencies		BCNF
IsPartOf (Forum-User)	UserID ForumID	No dependencies		BCNF

4.3 Anomalies

Problems that arise from relations that are generated directly from user views are called anomalies. Relations that have redundant data may have problems called ”update anomalies”, which are classified as:

- Insertion anomalies

- Deletion anomalies
- Modification anomalies

The anomalies of the project have been listed as follows:

Table	Insertion Anomaly	Deletion Anomaly	Modification Anomaly
PAGE	None	On deleting an entry, other tables using PageID as a foreign key are affected	None
Admin	A new entry in this table will affect the Page table, where PageID is a primary key.	None	Modifying an entry in this table will affect the Page table, where PageID is a primary key.
POST	A new entry in this table will affect the Page, User and Forum tables, where PageID, UserID and ForumID are primary keys.	Deleting an entry in this table will affect the Page, User and Forum tables, where PageID, UserID and ForumID are primary keys.	Updating an entry in this table will affect the Page, User and Forum tables, where PageID, UserID and ForumID are primary keys.
Popularity	A new entry with a new PostID cannot be added as there is no new post	None	Modifying an entry in this table will affect the Post table, where PostID is a primary key.
Comments	A new entry with a new PostID cannot be added as there is no new post	None	Modifying an entry in this table will affect the Post table, where PostID is a primary key.
Likes	A new entry with a new PostID cannot be added as there is no new post	None	Modifying an entry in this table will affect the Post table, where PostID is a primary key.
USER	None	On deleting an entry, other tables using UserID as a foreign key are affected	None
ContactDetails	A new entry with a new UserID cannot be added as there is no new user	None	Modyfing the UserID must lead to multiple modifications in all tables where UserID has been used
Phone	A new entry with a new UserID cannot be added as	None	Modyfing the UserID must lead to multiple

	there is no new user		modifications in all tables where UserID has been used
E-Mail	A new entry with a new UserID cannot be added as there is no new user	None	Modyfing the UserID must lead to multiple modifications in all tables where UserID has been used
Education	A new entry with a new UserID cannot be added as there is no new user	None	Modyfing the UserID must lead to multiple modifications in all tables where UserID has been used
Work	A new entry with a new UserID cannot be added as there is no new user	None	Modyfing the UserID must lead to multiple modifications in all tables where UserID has been used
FRIEND	A new entry with a new UserID cannot be added as there is no new user	None	Modyfing the FriendID must lead to multiple modifications in all tables where FriendID has been used
FORUM	None	On deleting an entry, other tables using ForumID as a foreign key are affected	None
Members	A new entry with a new ForumID or UserID cannot be added as there is no new User or Forum	None	Modifying an entry in this table will affect the Forum and User tables, where ForumID and UserID are primary keys.
Administrators	A new entry with a new ForumID or UserID cannot be added as there is no new User or Forum	None	Modifying an entry in this table will affect the Forum and User tables, where ForumID and UserID are primary keys.
EVENT	A new entry with a new UserID cannot be added as there is no new user	None	Modyfing the EventID must lead to multiple modifications in all tables where EventID has been used

Invited	A new entry with a new EventID or UserID cannot be added as there is no new User or Event	None	Modifying an entry in this table will affect the Forum and User tables, where ForumID and UserID are primary keys.
APPLICATION	None	On deleting an entry, other tables using ApplicationID as a foreign key are affected	None
Permissions	A new entry with a new ApplicationID cannot be added as there is no new Application	None	Modifying an entry in this table will affect the Application table, where ApplicationID is a primary key.
ChatsWith	A new entry with a new UserID cannot be added as there is no new User or Friend	None	Modifying an entry in this table will affect the Friend and User tables, where FriendID and UserID are primary keys.
Likes	A new entry with a new PageID or UserID cannot be added as there is no new User or Page	None	Modifying an entry in this table will affect the Page and User tables, where PageID and UserID are primary keys.
InvitedTo	A new entry with a new EventID or UserID cannot be added as there is no new User or Event	None	Modifying an entry in this table will affect the Forum and User tables, where ForumID and UserID are primary keys.
Uses	A new entry with a new ApplicationID or UserID cannot be added as there is no new User or Application	None	Modifying an entry in this table will affect the Application table, where ApplicationID is a primary key.
IsPartOf	A new entry with a new ForumID or UserID cannot be added as there is no new User or Forum	None	Modifying an entry in this table will affect the Forum and User tables, where ForumID and UserID are primary keys.

4.4 Normalization of Tables

Database designed based on ER model may have some amount of inconsistency, ambiguity and redundancy. To resolve these issues some amount of refinement is required. Normalization is the process of splitting relations into structured relations that allow users to inset, delete, and update tuples without introducing database inconsistencies. Without normalization many problems can occur when trying to load an integrated conceptual model into the DBMS.

Normalization has three goals:

- To eliminate redundant data (e.g. storing the same data in more than one table)
- To store only related data in a same table.
- Organize data efficiently

These goals reduce the amount of space a database consumes, ensures data is logically stored, and maximize operational efficiency. A good database design includes the normalization, without normalization a database system may slow, inefficient and might not produce the expected result.

Normalization is often executed as a series of steps. Each step corresponds to a specific normal form that has known properties. As normalization proceeds, the relations become progressively more restricted in format, and also less vulnerable to update anomalies.

The imperative normal forms are:

- First Normal Form (1NF) is a relation in which the *intersection of each row and column contains one and only one value.*
- Second normal form (2NF) is a relation that is in first normal form and every non-primary-key attribute is fully functionally dependent on the primary key.
- Third Normal Form is a relation that is in first and second normal form, and in which no non-primary-key attribute is transitively dependent on the primary key.
- A relation is in Boyce-Codd normal form (BCNF), if and only if, every determinant is a candidate key.
- Fourth normal form (4NF) : A relation that is in Boyce-Codd normal form and contains no nontrivial multi-valued dependencies.
- Fifth normal form (5NF): A relation that has no join dependency.

The final normal forms of the hitherto un-normalized tables are:

ADMIN -> Admin1

PageID	AdminUserID

POST -> Post1, Post2

PostID	Content	Date_and_Time	MediaURL	Privacy Settings

PostID PageID UserID ForumID

Comment -> Comment1, Comment2

PostID Date_and_Time Content

PostID UserID

FRIEND -> Friend1, Friend2

FriendID Friends Since UserID

FriendID Name

EVENT -> Event1, Event2

EventID Name Date_and_Time Location Description

EventID CreatorUserID

5. Implementation

5.1 Table Structures and Data

The database consists of the following populated tables:

```
mysql> select * from Page;
+--------+--------------+------------+-----------+-------------------------------------------------------------------------+
| PageID | Name         | TypeOfPage | NoOfLikes | Description                                                             |
+--------+--------------+------------+-----------+-------------------------------------------------------------------------+
| P1     | SIT          | Education  | 1000      | A page for the college Symbiosis Institute of Technology.               |
| P2     | Harry Potter | Books      | 7000      | For all Harry Potter fans!                                              |
| P3     | Lawn Tennis  | Sport      | 9000      | A page that describes all facets of the game that is Lawn Tennis.       |
| P4     | Cakes        | Culture    | 5000      | Decadent cakes of all kinds!                                            |
| P5     | Nature       | Lifestyle  | 8000      | This page showcases the beauty of nature.                               |
+--------+--------------+------------+-----------+-------------------------------------------------------------------------+
```

```
mysql> select * from Admin1;
+--------+-------------+
| PageID | AdminUserID |
+--------+-------------+
| P1     | U1          |
| P4     | U2          |
| P5     | U2          |
| P2     | U3          |
| P3     | U3          |
+--------+-------------+
```

```
mysql> select * from Post1;
--------+---------------------+--------------------------------+-----------------------------------------------------+-----------------+
 PostID | Time                | Content                        | MediaURL                                            | PrivacySettings |
--------+---------------------+--------------------------------+-----------------------------------------------------+-----------------+
 P01    | 2015-06-06 17:00:00 | SIT is the best!               | null                                                | Private         |
 P02    | 2015-06-07 18:00:00 | Harry Potter is the best!      | null                                                | Public          |
 P03    | 2015-06-08 19:00:00 | Jokes are funny!               | null                                                | Public          |
 P04    | 2015-06-09 20:00:00 | Cakes are so tasty!            | null                                                | Private         |
 P05    | 2015-06-10 21:00:00 | Nature is so calm and peaceful. | https://www.SocialNetwork.com/ProfilePictures/P05.jpg | Public          |
--------+---------------------+--------------------------------+-----------------------------------------------------+-----------------+
```

```
mysql> select * from Post2;
+--------+--------+--------+---------+
| PostID | UserID | PageID | ForumID |
+--------+--------+--------+---------+
| P01    | U1     | null   | null    |
| P02    | null   | P2     | null    |
| P03    | null   | null   | F3      |
| P04    | U4     | null   | null    |
| P05    | null   | P5     | null    |
+--------+--------+--------+---------+
```

```
mysql> select * from Popularity;
--------+-----------+
 PostID | NoOfViews |
--------+-----------+
 P01    | 4         |
 P02    | 200       |
 P03    | 500       |
 P04    | 10        |
 P05    | 300       |
--------+-----------+
```

```
mysql> select * from Comment1;
+--------+-----------+---------------------+
| PostID | Content   | Time                |
+--------+-----------+---------------------+
| P01    | So true!  | 2015-06-06 18:00:00 |
| P02    | Yes!      | 2015-06-07 19:00:00 |
| P02    | Indeed!   | 2015-06-07 20:00:00 |
| P03    | They are! | 2015-06-08 20:00:00 |
| P05    | I agree.  | 2015-06-10 22:00:00 |
+--------+-----------+---------------------+
```

```
mysql> select * from Comment2;
+--------+--------+
| PostID | UserID |
+--------+--------+
| P01    | U2     |
| P02    | U2     |
| P02    | U1     |
| P03    | U3     |
| P05    | U1     |
+--------+--------+
```

```
mysql> select * from LikesForPost;
--------+--------+---------------------+
 PostID | UserID | Time                |
--------+--------+---------------------+
 P01    | U2     | 2015-06-06 18:00:00 |
 P02    | U2     | 2015-06-07 19:00:00 |
 P02    | U1     | 2015-06-07 20:00:00 |
 P03    | U4     | 2015-06-08 20:00:00 |
 P04    | U5     | 2015-06-09 21:00:00 |
 P05    | U3     | 2015-06-10 22:00:00 |
--------+--------+---------------------+
```

```
mysql> select * from User;
+--------+--------+----------+-----------------------------------------------------+--------------------------------+
| UserID | Name   | Password | ProfilePictureURL                                   | Description                    |
+--------+--------+----------+-----------------------------------------------------+--------------------------------+
| U1     | User 1 | pass1    | https://www.SocialNetwork.com/ProfilePictures/U1.jpg | I love SIT!                    |
| U2     | User 2 | pass2    | https://www.SocialNetwork.com/ProfilePictures/U2.jpg | I love Nature and Cakes!       |
| U3     | User 3 | pass3    | https://www.SocialNetwork.com/ProfilePictures/U3.jpg | I love Harry Potter and Tennis! |
| U4     | User 4 | pass4    | https://www.SocialNetwork.com/ProfilePictures/U4.jpg | I love Physics and Cakes!      |
| U5     | User 5 | pass5    | https://www.SocialNetwork.com/ProfilePictures/U5.jpg | I love Dogs!                   |
+--------+--------+----------+-----------------------------------------------------+--------------------------------+

mysql> select * from Phone;
+--------+---------+
| UserID | PhoneNo |
+--------+---------+
| U1     | 1111    |
| U2     | 1112    |
| U2     | 222     |
| U3     | 333     |
| U4     | 444     |
| U5     | 555     |
+--------+---------+

mysql> select * from Email;
+--------+----------+
| UserID | EmailID  |
+--------+----------+
| U1     | 1@sn.com |
| U2     | 2@sn.com |
| U2     | 3@sn.com |
| U3     | 4@sn.com |
| U4     | 5@sn.com |
+--------+----------+

mysql> select * from Education;
+--------+-----------+----------+---------------+
| UserID | Institute | Duration | Details       |
+--------+-----------+----------+---------------+
| U1     | I1        | 2 years  | School Topper |
| U2     | I2        | 5 years  | null          |
| U3     | I3        | 12 years | All Rounder   |
| U4     | I4        | 3 years  | Debate Team   |
| U5     | I5        | 6 years  | Sports        |
+--------+-----------+----------+---------------+

mysql> select * from Work;
+--------+--------------+----------+----------------------+
| UserID | Organisation | Duration | Details              |
+--------+--------------+----------+----------------------+
| U1     | O1           | 1 year   | Sofware Development |
| U2     | O2           | 5 years  | HR Manager           |
| U3     | O3           | 12 years | Vice President       |
+--------+--------------+----------+----------------------+

mysql> select * from Friend1;
+----------+--------+--------------+
| FriendID | UserID | FriendsSince |
+----------+--------+--------------+
| U1       | U2     | 2010         |
| U2       | U1     | 2010         |
| U2       | U3     | 2011         |
| U3       | U2     | 2011         |
| U5       | U2     | 2012         |
| U1       | U5     | 2012         |
+----------+--------+--------------+

mysql> select * from Friend2;
+----------+--------+
| FriendID | Name   |
+----------+--------+
| U1       | User 1 |
| U2       | User 2 |
| U2       | User 2 |
| U3       | User 3 |
| U5       | User 5 |
| U1       | User 1 |
+----------+--------+

mysql> select * from Forum;
+---------+-------------------------+-----------------+
| ForumID | Description             | PrivacySettings |
+---------+-------------------------+-----------------+
| F01     | For Physics Enthusiasts | Public          |
| F02     | For Animal Lovers       | Public          |
| F03     | Jokes for Everyone      | Public          |
| F04     | My Friends              | Private         |
| F05     | SIT Issues              | Private         |
+---------+-------------------------+-----------------+

mysql> select * from Member;
+---------+--------+------------+
| ForumID | UserID | MemberName |
+---------+--------+------------+
| F01     | U4     | User 1     |
| F03     | U3     | User 3     |
| F02     | U2     | User 2     |
| F03     | U1     | User 1     |
| F04     | U2     | User 2     |
| F04     | U1     | User 1     |
| F05     | U5     | User 5     |
+---------+--------+------------+

mysql> select * from Administrator;
+---------+--------+-----------+
| ForumID | UserID | AdminName |
+---------+--------+-----------+
| F01     | U4     | User 1    |
| F02     | U2     | User 2    |
| F03     | U1     | User 1    |
| F04     | U2     | User 2    |
| F04     | U1     | User 1    |
| F05     | U5     | User 5    |
+---------+--------+-----------+
```

```
mysql> select * from Event1;
+---------+-------------+------------+----------+-------------------------------+
| EventID | Name        | Time       | Location | Description                   |
+---------+-------------+------------+----------+-------------------------------+
| E1      | My Birthday! | 2015-06-05 | Add1     | NA                            |
| E2      | Reverb      | 2015-06-06 | Add 3    | SIT Festival                  |
| E3      | ExtemboBate | 2015-06-07 | Add 4    | Extempore and Debate          |
| E4      | HP Marathon | 2015-06-08 | Add 5    | Watch all Harry Potter Movies |
| E5      | Wimbledon   | 2015-06-09 | Add 6    | Watch Wimbledon together on TV |
+---------+-------------+------------+----------+-------------------------------+

mysql> select * from Event2;
+---------+-----------+
| EventID | CreatorID |
+---------+-----------+
| E1      | U1        |
| E2      | U2        |
| E3      | U3        |
| E4      | U4        |
| E5      | U5        |
+---------+-----------+

mysql> select * from Invited;
+---------+--------+-------+---------+
| EventID | UserID | Going | Invited |
+---------+--------+-------+---------+
| E1      | U5     | Yes   | Yes     |
| E2      | U4     | No    | Yes     |
| E3      | U3     | Yes   | Yes     |
| E4      | U2     | No    | Yes     |
| E5      | U1     | Yes   | Yes     |
+---------+--------+-------+---------+

mysql> select * from Application;
+---------------+--------+-----------------------------------------------------+----------------------+
| ApplicationID | Name   | ImageURL                                            | Link                 |
+---------------+--------+-----------------------------------------------------+----------------------+
| A1            | App 1 | https://www.SocialNetwork.com/ProfilePictures/A1.jpg | https://www.App1.com |
| A2            | App 2 | https://www.SocialNetwork.com/ProfilePictures/A2.jpg | https://www.App2.com |
| A3            | App 3 | https://www.SocialNetwork.com/ProfilePictures/A3.jpg | https://www.App3.com |
| A4            | App 4 | https://www.SocialNetwork.com/ProfilePictures/A4.jpg | https://www.App4.com |
| A5            | App 5 | https://www.SocialNetwork.com/ProfilePictures/A5.jpg | https://www.App5.com |
+---------------+--------+-----------------------------------------------------+----------------------+

mysql> select * from Permissions;
+---------------+----------------------+
| ApplicationID | Permission           |
+---------------+----------------------+
| A1            | User Contact Details |
| A2            | None                 |
| A3            | User Name            |
| A4            | None                 |
| A5            | User and Friends     |
+---------------+----------------------+

mysql> select * from ChatsWith;
+--------+----------+-----------------+---------------------+
| UserID | FriendID | Content         | Time                |
+--------+----------+-----------------+---------------------+
| U1     | U2       | Hello!          | 2015-06-05 20:00:00 |
| U2     | U1       | Hi! How are you? | 2015-06-05 21:00:00 |
+--------+----------+-----------------+---------------------+

mysql> select * from Likes;
+--------+--------+
| UserID | PageID |
+--------+--------+
| U1     | P1     |
| U2     | P4     |
| U2     | P5     |
| U3     | P2     |
| U3     | P3     |
| U4     | P4     |
| U5     | P5     |
+--------+--------+

mysql> select * from InvitedTo;
+--------+---------+
| UserID | EventID |
+--------+---------+
| U5     | E1      |
| U4     | E2      |
| U3     | E3      |
| U2     | E4      |
| U1     | E5      |
+--------+---------+

mysql> select * from Uses;
+--------+---------------+
| UserID | ApplicationID |
+--------+---------------+
| U1     | A1            |
| U1     | A2            |
| U1     | A3            |
| U3     | A4            |
| U5     | A5            |
+--------+---------------+

mysql> select * from IsPartOf;
+--------+---------+
| UserID | ForumID |
+--------+---------+
| U1     | F03     |
| U1     | F04     |
| U2     | F02     |
| U2     | F04     |
| U3     | F03     |
| U4     | F01     |
| U5     | F05     |
+--------+---------+
```

5.2 Queries Fired

The following queries were fired on the database:

5.2.1 Which Forums is User 1 part of?

```
select Description from Forum where ForumID in(
    select ForumID from IsPartOf where UserID in(
        select UserID from User where Name='User1'));
```

```
+-------------+
| Description |
+-------------+
| My Friends  |
| SIT Issues  |
+-------------+
2 rows in set (0.03 sec)
```

5.2.2 Which pages does User 3 Like?

```
select Name from Page where PageID in(
  select PageID from Likes where UserID in(
    select UserID from User where Name='User1'));
```

```
+--------------+
| Name         |
+--------------+
| Harry Potter |
| Lawn Tennis  |
+--------------+
2 rows in set (0.00 sec)
```

5.2.3 Who are attending the 'My Birthday' event?

```
select Name from User where UserID in(
    select UserID from InvitedTo where EventID in
        (select EventID from Invited where Going='Yes')
        and
        (select EventID from Event1 where Name='My Birthday!'));
```

```
+--------+
| Name   |
+--------+
| User 5 |
+--------+
1 row in set (0.00 sec)
```

5.2.4 Who are User 5's friends?

```
select Name from Friend2 where FriendID in(
  select Friend ID from Friend1 where UserID in(
    select UserID from User where Name='User 5');
```

```
+--------+
| Name   |
+--------+
| User 2 |
+--------+
1 row in set (0.00 sec)
```

5.3 Procedures

A procedure to find the number of friends of a given user was also performed:

```
Delimiter |
create function GetNoOfFriends(UserName text)
returns int
Begin
        Declare total int;
        select count(F.FriendID)  into total from User U, Friend1 F group by (UserID) having
        U.Name=Username and U.UserID=F.UserID;
        return total;
End |
```

To execute,

```
select GetNoOfFriends('User 1') as 'Number of Friends'\g
```

```
+-------------------+
| Number of Friends |
+-------------------+
| 2                 |
+-------------------+
1 row in set (0.00 sec)
```

6. Conclusions of Learning

It has been a matter of immense pride and pleasure challenge to have taken up this project and complete it successfully. While developing this project I have learnt a lot about Social Networking and its inner functioning.

During the development process I studied and understood the criteria for making a software more efficient, I also understood the importance of maintaining a minimal margin for error.

7. References

- Information shared with us by Prof. Shruti Patil
- www.wikipedia.org
- www.w3shools.org
- www.mysql.com
- www.stackoverflow.com
- Various sources from the internet