

# ITECH3224/6224 World Wide Web Technology 2

## 1517/1520 Assignment 1 – "Microblog"

### Introduction

This is an individual assignment in which you are required to develop a dynamic web application as described below using PHP, MySQL, JavaScript and CSS.

Details of the weight of the assignment and due date are given in the course description.

### Task Description

#### Application

For this assignment, you will create a web-based microblogging platform, similar in functionality to "Twitter". You will have to come up with your own fancy name.

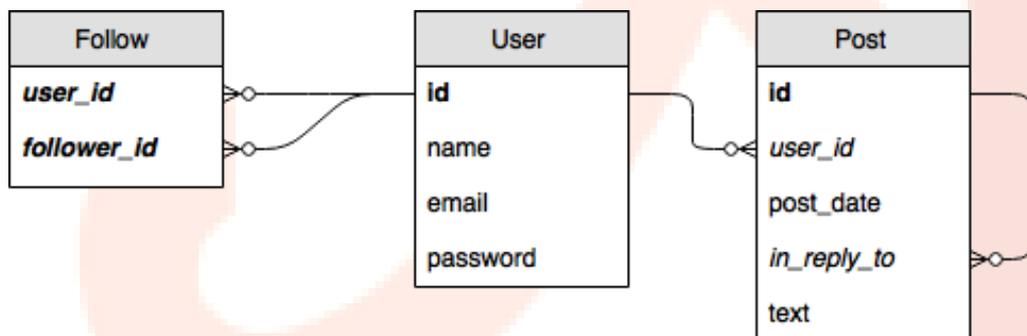
#### Database Structure

The web application uses a relational database to create a micro-blogging platform. The database has the following structure:

**User**(id, name, email, password)

**Post**(id, *user\_id*, post\_date, *in\_reply\_to*, text)

**Follow**(*user\_id*, *follower\_id*)



Each record in the Post table refers to a single post made by a user. Primary keys are indicated with underlines or bold formatting, and foreign keys are italicized.

The following constraints should be applied in implementing the application:

- The **user\_id** and **follower\_id** fields in the Follow table form a compound primary key, and both refer to the **id** field in the User table. The user **id** field should be an integer primary key, and should automatically increment.
- The **post\_date** field may be stored as either strings, or using MySQL datetime types, and must include time as well as date.

- The **text** field in the Post table should be limited to 160 characters.
- The **in\_reply\_to** field in the Post table is a **nullable foreign key** that refers to the **id** field on the same table. If this field is **null**, then the post is not a reply to another post – if it is non-null then the post is a reply to the post with the matching id.
- The **password** field should be a VARCHAR field of **at least 125** characters. The **name** and **email** fields should be VARCHAR of a length that you determine to be reasonable and sufficient.

### Initial data

When the database is created it should be populated with data of your own invention. Include this data as part of your written report. Each table should have between 3 and 6 records initially.

**Include yourself as a user**, using your **student id** as the name, and your real email address. Invent other users as necessary – perhaps use characters from your favourite movie or band.

### Creating the database

Create an SQL file that creates a database on the server, creates the three tables above, and populates them with your initial data.

Password data should be hashed using, *at minimum*, the **crypt()** PHP function. If available, prefer to use the PHP **password\_hash()** function to generate password hashes. It is acceptable for all initial User records to share the same password for testing. **Use of MD5 is not acceptable.**

**Include a user that can login as ‘tutor’ with password ‘guest’.**

Test your database by writing queries on the command line that display all initial data using SELECT statements, and **include the queries in your report.**

### User accounts

Write an HTML form that allows new users to sign up. The form should request a username, email address and password. The password **must be hashed** before storing it in the database.

Using PHP, validate that the username is unique, and the password is at least 5 characters.

Write PHP code to allow users to log in and log out. This will require the use of sessions or cookies.

### Global timeline

Write PHP and HTML code to display a list of the last 10 posts from all users, sorted in **descending date order** – that is, the most recent posts are at the top. This timeline of posts should be visible to anybody without logging in.

If posts are in reply to another post, include a link to the original.

### Creating posts

Logged-in users should be able to create new posts. Write PHP code to support this. **Use both PHP and Javascript to limit the length of the post to 160 characters.**

### Following

Write HTML and PHP code to allow logged-in users to **follow** other users. This should create a new entry in the **Follow** table. This may require you to create a way to view an individual user, or list of all users.

Users who are logged in should see **posts only from users that they follow**, instead of the global timeline. This list should again be limited to the last 10 posts.

## Replying

Create a link on each post that allows a logged-in user to **reply** to a post. This should allow them to create a new entry in the **Post** table, with the **in\_reply\_to** foreign-key field set appropriately.

## Aggregate data

Complete the following using SQL aggregation such as COUNT and SUM, subqueries or nested SELECT statements, inner joins and (left or right) outer joins.

- Create a page that contains a **list of the top 10 most-followed users, with their number of followers** ordered in descending order. Similarly create a page that contains a list of the **top 20 most popular posts by number of replies**.
- For each post in the timeline that **has replies**, display the number of replies with the post.
- Display the number of posts that have been made in the last 24 hours at the top of the global timeline.

## Additional task for ITECH6224 Students

On the topic of “Database Sharding” identify four (4), relevant, independent resources. Use these resources to write a discussion of the advantages and limitations of database sharding, where it is appropriate to use, and alternatives. Write this in your own words, and take care to cite appropriately. Word count should be about 500 words.

*Note: Wikipedia is not a suitable resource, but may be used to get an overview of the subject*

## Further details

### Documentation

Include a written report containing:

- Initial data details
- The SQL queries you used to test your database
- A list of parts of the assignment you have completed or not completed.
- Details of specific assistance you received from people other than your lecturer or tutor, and the names of those assisting.

### Assignment support

This assignment is supported by the first 5 lectures and the first 6 labs. Work on the assignment should be spread over a number of weeks after the relevant lab has been mastered.

Refer to the Course Description for details of submission to Moodle, late assignments, extensions, special consideration, plagiarism, student support, presentation of academic work and adopted reference style.

### Submission

All files should be zipped and uploaded to Moodle by the due date and time. Check with your tutor as to whether a hard copy is required in addition to the electronic submission.

## ITECH3224/6224 World Wide Web Technology 2

### Assignment 1 Marking Guide

Name _____	ID _____	Marker _____
Criterion	Maximum	Obtained
Initial data – Requirements satisfied	1	
Creating the database: Table structure, data types, field lengths, initial data entry	1	
User accounts	3	
- Sign-up form is well designed and functional	1	
- Server-side validation of username and password	1	
- Appropriate password hashing	1	
Global timeline	2	
- Most recent 10 posts listed in descending order.	2	
Creating posts	2	
- Users can create new posts	1	
- Use of PHP and JavaScript to limit post length	1	
Following and Replying	3	
- Logged-in users can follow others, and see only posts from those they follow	2	
- Users can reply to other users' posts	1	
Aggregate data	3	-
- Most followed, most popular posts	1	
- Replies per post, total in last 24 hrs	2	
Written task (ITECH6224 only)	5	
- Content	3	
- Writing style	1	
- Adherence to academic standards	1	
Documentation	3	
- Initial data	1	
- Completion of tasks	1	
- Assistance statement	1	
Quality of code – layout, naming scheme, etc. HTML is valid	2	
Totals		
- ITECH3224	20	
- ITECH6224	25	
Final scaled mark (correct to 1 decimal place)	<b>20</b>	

Comments: