

Final Project

Web-Based Chatting Application (40%)

DUE DATE

Session 15

OBJECTIVE

The main objectives of this project are to:

- Interpret specifications and analysis performed
- Design a solution based on the requirements and specifications
- Design the logic required for a complete program design solution
- Write the code for the program logic.
- Create the database that will be used for the Chatting applications
- Debug and test your program and make sure it is working properly.

DESCRIPTION

This project will allow you to apply your knowledge and skill in PHP and MYSQL Programming. You will need to read and analyze the information provided and extract the vital pieces of information that will allow you to design the program logic and features of the solution to the problem presented. For the purposes of this project you will need to leverage the major components, features techniques and procedures that you learned in this course in order to complete the requirements of this project. You will need to create flowcharts, write pseudocode, prepare diagrams, write the code, create a database, test your web application and make sure it is working properly without any errors or missing links as part of your final solution.

Time Required

You will require 10-12 hours to complete this project.

Required Material

You will need the following material to complete this project:

- PHP Editor
- Server.
- MYSQL program
- Microsoft Word (for document preparation)

In programming there are generally multiple possible solutions to the same problem. Your solution may not be identical to someone else's, but that does not mean that it is wrong. Your solution will not and

should not look exactly like your colleague's solution. Any good solution is acceptable if produced following the principles and guidelines presented in this course.

INSTRUCTIONS

In the professional world, a large amount of communication between individuals is done via messages. It is one of the most secure and fast mediums of communication for a modern era. Messaging is also a good way to connect with your friends; it helps people to be active in social life.

What is a **chatting application**? Chatting applications are programmed to allow you to send messages to your friends, who are registered with that application. It allows a user to transfer text and data like picture, video, mp3, to another user. However, it typically has some limitations on various parameters such as the size of the attachment. However, you can change those parameters for your server and as per business needs.

In a chatting application, a user can register herself/himself. After registration, they can login with their existing user id and password.

In this project you will create a simple web based chat application with PHP and MYSQL.

The Chatting Application must be able to do these functions:

1. Allow the user to login or logout.
2. Handle user's inputs.
3. Displaying messages to other logged in users.
4. Display a chat log.

Front End: HTML, CSS, JavaScript

1. **HTML**: It is the program that is used to create and save HTML documents.
2. **CSS**: (Cascading Style Sheets) It is a style sheet language. It is used to describe the look and format of a document written in a markup language.
3. **JavaScript**: It is a dynamic computer programming language. JavaScript is mostly used as part of web browsers.

Back End: PHP, MySQL

1. **PHP**: Hypertext Preprocessor (PHP) is a technology that allows software developers to create dynamically generated web pages, in HTML, XML, or other document types, as per client request. PHP is open source software.
2. **MYSQL**: MYSQL is a database, widely used for accessing querying, updating, and managing data in databases.

Key Functions

Your system should contain the following Functions:

1. Allow the user to login
2. Allow the user to log out
3. Allow users to send messages
4. Display messages from users
5. Create and update the log file for the chatting room.

Requirements:

Model the Chatting Application:

Make sure you read through all of the specifications and extract the key pieces of information. Be on the lookout for processes that repeat themselves in various locations, this can be helpful to apply modularization techniques. Consider the various messages that the system is required to produce and document them.

Design the Program Logic

You will create the various, flowcharts, pseudocode and coding to implement the Chatting Application:

1. Flowcharts and Pseudocode

Model an algorithm for each of the processes described above using a flowchart or pseudocode

2. Database

Create the tables with all the needed columns for the Chatting application. Add some dummy data for the users Database.

3. Coding the program logic

Using PHP, MYSQL, HTML, CSS and JavaScript create the Chatting application. Connect the application to the database.

GUI DESIGN

This Chatting application proposes different screens that the users will see at any given point in time. As part of any program design, the GUI design is an important component as it allows the designer to adapt the screen layouts to meet the design and functionality standards.

Requirements

1. Create the different wireframes for the screens that your Chatting applicaiton will use. Your wireframes, must include all the options that the program will provide.
2. Prepare a storyboard showing how the screens are supposed to function together and the sequence of the screens.
3. Prepare an Object dictionary which lists the interface objects used in the program, where they are found (Screen) any variables that they impact and any procedure, method or function that they invoke
4. Human program interaction is very important to allow the user to communicate with your program in an easy and efficient way.

Class Design:

Object Oriented design you should be able to identify and design a number of classes for this solution. Create a class diagram for the created classes.

Where applicable, you must show inheritance, dependency or association between the different classes. Your class diagrams must use the proper format and naming conventions

SUBMISSION INSTRUCTIONS

Your project must include the following:

1. A project report containing:
 - a title page with your name, the submission date and your instructor's name
 - a table of contents
 - the project specifications
 - all relevant documentation including diagrams, flowcharts, pseudocode, coding, Database.
2. The source code of your program.
3. You must include an "install.php" file to perform the creation of the database and the transferring the application file structure to the local machine.
4. To receive a mark, the application must install and work properly.
5. Place the entire directory structure in a single folder.
6. Zip the folder and submit a single zip file with all your work in the submission page for this assignment.

Work must be submitted in the correct file type and be properly labelled as per the College naming convention:

NAME_COURSE_ASSIGNMENT. E.g. XuXiaLing_FM50D_A01.

GRADING CRITERIA

Assignment Value: **40%**

Grading Criteria	Grading
Analysis for the chatting application	10
Design for the chatting application	10
Implementation for the chatting application	25
Database created	10

Communication between the code and the database	10
GUI design	15
Class diagrams	5
The website functionality	15
TOTAL	/100

Penalties

- A penalty of 5% will be deducted from your mark for each day your project is late.
- Any project submitted more than 3 calendar days late will receive a maximum mark of 60%.