COMP 220 Fall 2022

Team Coding Project Handout

With your team, set up meeting dates, times and locations as follows.

|  |  |
| --- | --- |
| Oct 31st - Nov 2nd | Before your first meeting, read through this handout carefully.  Meet with your team (approximately 30 minutes). See Initial Team Meeting below. |
| Nov 7th – 9th | Meet with your team (approximately 90 minutes). See Check In Meeting below. |
| Nov 14th – 16th | Meet with your team (could take multiple hours). See Code Integration Meeting below. |
| By Wed Nov 23rd 11pm | Submit assessment. See instructions below. |

Overview

In your team, you will create a text editor using php and mysql. The user creates a document by typing data into a text area. Menus include File (new, open and save), Edit (find) and Font (name and size). We will work with only one file: editor.dat.

The COMP 220 course materials and the COMP 205 playbook and in class website will be handy for this assignment.

You will work in a team. The teams are listed on Blackboard. Partners are indicated by the colour coding in the spreadsheet. Note that due to varying enrolment, your COMP 220 team may be different than your CARE 10 team. You and your partner will pair code together. Each pair creates one third of an application.

Throughout the project, keep notes of the work you are doing. You will need it later.

This assessment builds skills you need for interviews, work placement and your career: pair coding, working effectively in a team, accountability, walking people through your code, essential employability skills and presentation skills.

You will be graded on the following:

* Presentation video and code (75%).
* Peer evaluation (25%).

Initial Team Meeting

Throughout the project, each student keeps their own record of each teammates progress. For each student in your group, document if they were on time and engaged in the initial team meeting.

To Do List

1. Decide who will be Pair A, Pair B, Pair C.
2. Confirm date and time for your next team meeting.
3. In your group, draw out a rough draft GUI plan on paper or on computer. Include a colour scheme and all required elements.
4. After the meeting, each student is to:
   1. create a sample text file for testing.
   2. meet with partner and make progress on pair coding. You will present your progress at the next meeting.
   3. run mysql workbench and create the fontNames table.

Use comp220db;

create table fontNames (fontName varchar(50), primary key (fontName));

insert into fontNames values ('Arial');

insert into fontNames values ('Brush Script MT');

insert into fontNames values ('Comic Sans MS');

insert into fontNames values ('Courier New');

insert into fontNames values ('Verdana');

Check In Meeting

Each student keeps their own record of their teammates progress. For each student in your group, document the following. You will need this information later.

* Was their work done in a timely manner?
* Were they on time and engaged in the team meeting?

To Do List

1. Each member speaks to the following items. Be sure to record your teammates comments and progress.
   * what I have been working on since last meeting.
   * what problems did I have and how they were resolved.
   * what are my problem areas.
   * what I will do next.
2. Confirm date and time for your next team meeting.
3. After the meeting, complete your assigned coding tasks.

Code Integration Meeting

Each student keeps their own record of their teammates progress. For each student in your group, document the following. You will need this information later.

* Was their work done in a timely manner?
* Were they on time and engaged in the team meeting?

To Do List

1. Decide which team member is responsible for submitting the code by the deadline.
2. Decide which team member is responsible for submitting the video by the deadline.
3. Combine code and get the application working.
4. Create the presentation. If needed, schedule an additional meeting before the deadline.

GUI

There is exactly one web page to this project. Include the following:

* Heading with all students first and last names.
* Colour scheme.
* Horizontal menu with 3 main items: File, Edit, Font. Each will hold a dropdown menu.
* File drop down has buttons for New, Open and Save.
* Edit drop down with:
  + Find label and search textbox.
  + Case sensitive label and checkbox.
  + Find button.
* Font drop down allows the user to change the font in the text area.
  + Font label and list box. Fonts will be read in from the fontNames table and loaded in to the listbox.
  + Font size label and listbox. The listbox has 3 items: small, medium and large.
* Large text area.
  + On initial load and on File New, text area displays a placeholder: *enter text here*.
  + On save and open, text area displays file contents.
* Additional page elements and styling are optional.

Code Design

This code is about understanding concepts and thinking through an effective solution. There are not a lot of code lines. The main php file is likely less than 200 lines; javascript 10 – 15 lines.

Pair A Tasks

Code function drawMenu to create the GUI for the dropdown menus. To build the dropdown menu design, you can create your own or use one from the internet. Call drawFileDropDown, drawEditDropDown and drawFontDropDown as appropriate.

Code functions drawFileDropDown and drawEditDropDown. Do not code drawFontDropDown as that is being done by another pair. The above functions have no parameters and do not return a value.

Code function saveFile and function openFile. Handy functions are fwrite and fread. The filename is editor.dat. Display the messages below as appropriate:

* *File saved.*
* *Error Saving File.*
* *File opened.*
* *Error opening file*.

Function saveFile has one parameter: the text to be saved to the file (one long string). It does not return a value. Function openFile has no parameters and returns the text retrieved from the file.

Note that an error is given if the user tries to open a file when editor.dat does not exist. Use the file\_exists function in openFile. If the file does not exist, display *Editor.dat does not exist. Please save file first.*

Functions can be written and tested before main is ready. File New does not require its own function. You do not have to account for the user forgetting to click Save.

Pair B Tasks

Code function drawFontDropDown. For this work, use COMP 205 and COMP 220 methodologies. The JavaScript for this web site is at the complexity level of COMP 205. Functionality:

* Retrieve font names from table fontNames and load into the list box. Assume the number of fonts and the font names will change over time as the table is updated regularly by another application.
* Using javascript, change the font name in the text area to the user selected font name.
* Display a font size list box containing 3 values: small, medium and large.
* Using javascript, change the font size in the text area to the user selected font size.

The function will be called by the drawMenu function. If your JavaScript event will not fire, try refreshing the browser with ctrl-F5. Note that font changes are for display only. They will not be saved to the file. To do so, an api is required.

Pair C Tasks

Code function findTextInFile. Find results are displayed with an echo statement. Sample results when searching for *feline cat*:

*String feline cat not found*

*Feline cat was found at position 4*.

When displaying the position number to the user, the first position should be 1, not 0.

The function can be written and tested before main is ready. While waiting for main to be completed, assign sample values for the text area and for the case sensitive variable.

Code main to manage the GUI. This portion of the code involves clear program design and planning: more time will be spent on this than coding. Main is responsible for the features listed below.

* Save text area contents to the post array.
* Call the following functions, as appropriate.
  + openFile.
  + saveFile.
  + findTextInFile.
  + drawMenu.
* Implement File New: display the place holder in the text area.
* Display the text area with the appropriate contents. Attributes:
  + - autofocus so focus goes to it on page load.
    - wrap: hard.
    - spell check: true.
* For testing purposes
  + Create New, Open, Save and Find html buttons. When all editor code is combined, remove the buttons.
  + Write an echo statement as a placeholder for each function call. For example:

echo "Running the open function here.";

Presentation

As a team, discuss and answer the Project Debrief questions below. Decide who will present each component. The presentation time should be split evenly among the teammates.

Create a 15 – 20 minute video of a walkthrough of your application. Before you start recording, load your code in Visual Code and display the application in Chrome. Your walk through should include the following:

* Demonstration of all software features, from the user’s perspective. Be sure to cover all components, even if they are not working. Presentation should consist of the steps below in the provided order.
  + File menu
    - File New. Type in a few sentences.
    - File Save.
    - File New.
    - File Open.
  + Edit menu
    - Find. Perform a successful case sensitive search.
    - Find. Perform an unsuccessful case sensitive search.
    - Find. Perform a successful case insensitive search.
    - Find. Perform an unsuccessful case insensitive search.
  + Font menu
    - Change font name to Brushscript and size to large.
    - Change font name to Courier New and size to small.
* Each student speaks for 1 - 2 minutes about the work they completed. Discuss:
  + Work you completed and challenges you faced.
  + Present a walkthrough of the code you have written. Walk throughs and debriefs are commonly used in the hiring process and in the workforce. Be sure to cover all code lines. You do not have to walk through the include file or the css file. A sample walk through is shown at <https://www.youtube.com/watch?v=elcijdLbjks>. Start at the 5:13 mark and watch to 6:30.
* Project Debrief questions
  + As a team, discuss the answers to the following questions. Note this question is asking about the team, individuals, relationships and soft skills. It is not about the software features or the coding. One team member to present the answers to the following:
    - What worked well?
    - What did not work well?
    - What could the team have done differently to have better results?

Peer Evaluations

Throughout the project, you kept a record of each teammate’s progress. After the code and presentation are complete, fill out the peer evaluation shown at the end of this document. This is to be filled out individually and privately by each student. Your evaluation is confidential. Peer Evaluation questions:

* Rate how well this student demonstrated a positive attitude during the project: availability, eagerness, respectful of others, active listening.
* Rate how well this student met deadlines.
* Did this student attend all meetings, arrive on time and participate fully?
* Did this student contribute ideas and suggestions?
* (Only answer this for your partner) Did this student contribute equally to the project?
* Explain why you chose the above ratings for this teammate.

Tips

1. You and your partner are a team and you are directed to do pair coding. Complete the work together, do not divide up the tasks. This prepares you for pair coding interviews and on the job training.
2. Read this document carefully so you understand the GUI, the functionality, and the roles of each pair.
3. Be clear on what portions of the code are your responsibility and what are that of others. Do only your tasks. You are not permitted to give code to your teammates that is outside your list of tasks. This is considered cheating. You can help them with the debugging of their code. You cannot write their code.
4. To complete your tasks, you do not have to wait for others to complete theirs. Create temporary controls, variables, files, function calls as needed for testing.
5. The specifications have been provided to you. Be sure to follow them. The code needs to be placed in the correct location, as defined in the specifications.

Submitting Your Project

1. If a team member has not submitted or completed their contribution, the presentation is to be done without it. Be sure to submit by the due date.
2. The designated team member submits the video file to Blackboard before the deadline. It is very important the instructions on Blackboard are followed exactly. All other team members are responsible for checking Blackboard to ensure it was submitted on time.
3. Create a folder named with the last names of all members listed in alphabetical order. Include one copy of the following files in the folder:
   1. main php file.
   2. js file.
   3. css file.
   4. editor.dat with a 5 – 10 complete sentences in it.
   5. include file.
   6. auth.txt.
   7. any additional files needed to run the application.

One team member was designated to submit the code before the deadline. Use Windows zip to create a zip file of the code folder. If the wrong file type is submitted, your supervisor will not be able to open it. The designated team member submits the file to Blackboard before the deadline.

1. All team members are responsible for checking Blackboard to ensure the video and the zip file were submitted on time. To do this, click on the *Team Coding Project Video* item on Blackboard. It will show if submitted or not.
2. After the code and presentation are complete, fill out the peer evaluation found at <https://forms.office.com/Pages/ResponsePage.aspx?id=JzlNl7SbyUi2XGCUOT0DC87Ac8OuM5xIqOZVcnht-dBUMUxKS1FQWDRDVTEwS1NUR1lWOVY5VUpHWC4u> . This is to be filled out individually and privately by each student. **Fill out this form for every teammate, other than yourself. If you have 5 teammates fill out the form 5 times.** A student who does not submit a fully completed peer evaluation for all team members earns a 0 on this portion of the assessment.