UMBC CSEE Undergraduate Advising Project – Part 2

**Due: Please look at your BB schedule.**

**Grade: Check Syllabus**

**Project Description:**

You are on your own using the same COEAdvising website in Project 1!

Here are the original Project 1 details.

|  |  |
| --- | --- |
| **Student** | **Advisors** |
| sign up/log in with student's name, ID # and major  (do not worry about verification for now) | schedule individual and group appointments during the work week (enter our available days/times for advisors) |
| individual sign ups where student would select advisor or group sign-ups (notice that group assignments will be more plentiful) | The advisors should be able to adjust the number of open seats in a session (from 10 down to 5, if we have fewer advisors for a certain day, etc…) |
| see what is available for group or individual advising, and select a date (set up a **simple way of storing appts**. Admin programmers will do that in more detail) | Advisors support Computer Science, Computer Engineering, Mechanical Engineering, Chemical Engineering and Engineering majors |
| should be able to cancel an appointment | be able to enter available times each week, or enter several weeks at once |
| search for appointments by date/time and by advisor, reschedule, and cancel appointments | print schedule for the day (by selecting the day) |
|  | Have only certain advising times for specific majors |
|  | for group advising, there are 10 possible spots |
|  | At least set up (data on advisors) for staff at bottom of [this page](http://coeit.umbc.edu/undergraduate-student-services-engineering-and-computer-science-majors/) |
|  | search for appointments by date/time and by advisor |
| **Details** | |
| Appts could start at 8:00 am and the last appointment could be 3:30 - 4:00 | |
| Both sides will need simulated data just to make sure your part of the project works. | |
| The list of advisors is on the advising web page given above. But think, they could add more people to their staff!! | |
| Individual and group appointments are the same length - 30 minutes.  The group advising is for a maximum of 10 students, individual is 1 student. | |

**Here are the Project 2** *additions***.**

|  |
| --- |
| **Advisors** |
| The advisors want two more data points added somewhere to the DB, where their respective offices are located, and where they will meet with their student. When a student signs up for an appointment, this data should also show. |
| The “work screen” (or where work shows up) is too skinny, make larger or replace so text takes up more space. |
|  |
| **Administration** |
| The overall pages are ugly. You may add different colors. |
| All footers should be an inserted “include” so if a change needs to happen, all pages change. |
| The number of $\_SESSION variables is way too high. Eliminate all but ‘userID’. If needed, some others can stay with negotiation. The rest can pull info from the DB when needed. This is a major code addition. |

**Users to accommodate:**

Student/Advisors: A features covered in previous table.

**Where to develop:**

The GL server is capable of supporting this project in two ways:

* html/php creation and service
* database/table creation using MyPHPAdmin (MySQL)
* the project will be placed in your directory on GL so it mimics below:

[www.csee.umbc.edu/~slupoli/CMSC331/Project2/](http://www.csee.umbc.edu/~slupoli/CMSC331/Project2/)

**Presentation**

* **10-15 min presentation without questions will be conducted AT Next Century’s location**
  + **be on time, or 5% penalty**
* **Professional Appearance Necessary**
* **Demonstration of the project**- will be given video equipment  
  - demo the working project   
  - I HIGHLY suggest either video or live version (this will kill a lot of time too!)
* **Project Completion List (each is a slide)  
  -** talk about
  + At the very beginning, a bio slide on each team member
    - Name
    - Pic
    - Rank
    - Any relevant experience
  + how you shared your work
  + talk about what as challenging
  + “in our NEXT release” (what didn’t get done, or what would have been nice with more time)
* **Handouts/ Folder (4 req.)**  
  - resume, with your picture on it somewhere
  + - * “Slick sheet”
  + + one page advertisement of your project that has:
* a brief description
* screen Captures (much like pulling a computer game box off the shelf would have)

**Project Submission:**

This is the most important part of the project. This is where you get to prove your work satisfies the requirements. You are to submit both an electronic and PAPER (3 ring binder with team on front, AND first page inside) version in DOCX (Word) of ***these in this order***:

**Part 1: Advising (Student and Advisor) webpage**

Length: at least 2 pages

Make sure to include:

1. **name the file index.php**
   1. **then have links to advisor or student from this page**
2. overall challenges you had
3. screen captures of your page in action!! (before and after)
   1. show how the options pop up
   2. again PROVE it works!!!
4. code (copy and paste)

**Part 2: Your personal DB table setup**

Length: at least 2 pages

Make sure to include:

1. overall challenges you had
2. screen captures of your tables!! (before and after)
   1. Click ”Structure” to show the overall structure of each table
   2. Click “Browse” to show data

**Part 3: What improvements we made**

Length: at least 2 pages

1. What did you get to improve
2. How?

**Part 4: Copy of your Presentation**

Length: at least 10 minutes

Make sure to include:

1. **10-15 min presentation without questions**
   1. **be on time, or 5% penalty**
2. **Professional Appearance Necessary**
3. **Demonstration of the program**- will be given video equipment  
   - demo the working project   
   - I HIGHLY suggest either video or live version (this will kill a lot of time too!)

**Project Completion List (each is a slide)**

* 1. At the very beginning, a bio slide on each team member
     1. Name
     2. Pic
     3. Rank (freshman, sophomore, etc…)
     4. Any relevant experience
  2. how you shared your work
  3. talk about what as challenging
  4. “in our NEXT release” (what didn’t get done, or what would have been nice with more time)

1. **Handouts/ Folder (4 req.)**  
   - resume, with your picture on it somewhere

- “Slick sheet”

+ one page advertisement of your project that has:

* a brief description
* screen Captures (much like pulling a computer game box off the shelf would have)

***YOU ARE NOT TO PUT CODE IN YOUR BINDER***

**How to submit:**

1. ALL files created in the GL directory /CMSC331/Project2/ must be submitted through GL. Use the sample below.

**submit cs331\_***instructorlastname* **Project2 \***

1. But the project will be tested using your pages created in the before mentioned CMSC331 directory.

**Project Specifications:**

1. As specified above, the application requires a MINIMUM of one(1) PHP/HTML page(s) for the users. The users requirements are also listed above.
2. You must have your material placed in the before mentioned directory
3. All pages must use PHP and use the dataserver given to you by UMBC
4. The application should be appealing and easy to use.
5. Use appropriate HTML Components (drop down boxes, radio buttons, etc…) to have validation on the front end where possible.

**Project Grading**

After the final deadline for submitting a project, your project will be graded for design and implementation, adherence to specifications, validation, appearance and ease of use. After all projects are graded, the grade will be posted in BlackBoard. The grade will detail the point deductions and penalties for your project.

|  |  |
| --- | --- |
| Design and Implementation | **40 points**  A design and implementation that meets all design and implementation expectations, but may have one or two minor problems  **30–39 points**  A design and implementation that has a few small problems and/or falls slightly short of the best approach  **20–29 points**  A design and implementation that has serious problems and/or falls significantly short of the best approach  **10–19 points**  A design and implementation with extremely serious problems, but nonetheless shows some effort and understanding of databases and PHP  **0 points**  A design and implementation that shows no effort to follow database and PHP principles |
| Documentation | **20 points**  Document specifications that meets all expectations, but may have one or two minor problems  **10 points**  Documentation has a few small problems and/or falls slightly short of the best approach in addressing the specifications required  **9-0 points**  A design and implementation that shows no effort to follow the specifications required |
| Appearance | **30 Points**  The design used special text features and layout **throughout the project** to make an appealing application.  **0 Points**  No real effort was made to make the application attractive |
| Validation | **10 points**  The design took great effort to make the application easy to use for the user  **0 Points**  No real effort was made to make the application easy to use |

**MULTIPLY ALL OF THESE x 2 since the Project is 200 total!!**