CS108 Final Project Requirements

Description

You will develop a web-based application with a database back-end for storage and retrieval of data. Choose your own adventure: select a subject matter related to your major or personal interests.

This is an individual project (i.e., no groups or teams), and counts for 15% of your course grade.

Due date: Wednesday 5/3 at 1:00pm. Late submissions will not be accepted.

Examples

Some recent CS108 projects have included:

- Interactive dictionary for English/Junta language translation
- Inventory system for chemistry lab / reactive agents
- Social network for a gardening club
- High school basketball win/loss records & standings
- Fisheries information tracking endangered species
- Geographic information system for archeology dig
- Compact disc inventory and sales
- Calorie counting database/user interface
- Local artists list their works for sale (with pictures)
- Car rental and return
- Special of the day listing for restaurant

Some examples of students' final projects are posted on this YouTube playlist:
http://www.youtube.com/playlist?list=PLDn7eKZUvBJHwUx_-eaVe1zTe3BFgTsmV

Detailed Requirements

You should begin by picking a concrete but simple interaction and modeling the data and functionality to implement it. For example, in a car rental application you might show a list of cars and allow a customer to complete a rental for one car. Start small; you can always add more interactions and details after you get the first one working.

Your application will model some kind of useful function, with a web-based interface (HTML) and data storage in a relational database (MySQL). A single interaction might be sufficient for this project, but a single web page certainly is not. You should have multiple web pages with logical transitions between the pages using buttons or links.

Projects must demonstrate reading from, inserting into, and updating database tables from your web interface. Points will be awarded for the general complexity of the project, with more complex projects scoring higher than very simple projects that mimic in-class assignments. After your work is functionally complete, you should enhance the look and feel of the application as the aesthetics (colors, images and fonts) will count for a small portion of your grade.

Topic Proposal

You must submit a topic proposal to me by email no later than **Friday 4/7 at 4pm.**

Your proposal should be brief (about 3 sentences) and describe the general subject matter (e.g., car rental), the specific interaction you will model (e.g. car selection and checkout) and the data you will require for this interaction (e.g., table of cars, table of customers, table of rentals which matches cars with customers).

<u>Projects on unapproved topics will not be accepted.</u>

Data Requirements

What kind of data will your application need to interact with? Describe the data in terms of records and fields. What attributes does each record need to have? How will you uniquely identify a record?

Make up some sample data (about 5 records) so that you can use to begin thinking about and testing the program's interactions. You might want to use a spreadsheet program like Excel to create these records, which will also enable importing them into the database later. We will demonstrate how to create tables and records in MySQL at the scheduled lab session this week.

Planning Your Time

You have 4 weeks to complete this project. Here are some guidelines on what you should be working on at what time:

Week 1 (through 4/12):

- Select your topic and email to Aaron by ***Friday 4/7 at 4pm***.
- Define your project's main interaction(s) (e.g., select car for rental, collect customer info, create rental) and which database tables you will need (e.g., cars, customers, rentals)
- Make up some sample data for each table you will need in Excel (a few records will suffice)

Week 2 (approximately 4/12 through 4/19):

- Build your database in MySQL, import your sample data
- Create a basic web application to view your data in a web page
- Begin building your interaction via forms, output pages, etc.

Week 3 (approximately 4/19 through 4/26):

- Continue building your interaction via forms, output pages, etc.
- Debugging, refine data as needed
- Validation of user inputs (i.e., don't let your program crash due to missing or wrong inputs)
- Add additional interactions to increase the scope/complexity of the project

Week 4 (approximately 4/26 through 5/3):

- Debugging, refine data as needed
- Continue adding additional interactions to increase the scope/complexity of the project
- Work on the look and feel of the project (clean up HTML, CSS, tables, fonts, colors, images)
- Documenting your code with comments
- Prepare your short presentation of the project (plan for a 2-3 minute demo)

Project Demonstration Video

You will present your project by making a short (2-3 minute) video screen capture, with you narrating as you demonstrate the application. Think about this as a demonstration for a potential customer or employer. This will speed the grading, and provide you with a lasting artifact of your work, which you can demonstrate to potential employers. Your demonstration should not show any code or database records, but must demonstrate every major feature of your application.

I recommend that you write a brief outline of what you will show in which order. Then, use a screen capture tool (I suggest QuickTime player for Mac or Tiny Take for Windows or any similar program). Record your video, and post it to YouTube (a private/unlisted URL is fine).

What to Submit

You will submit all python source code files to Blackboard. Also, post a comment with your submission giving the correct and working URL to your web page on the cs-people.bu.edu server, e.g.,

http://cs-webapps.bu.edu/cgi-bin/cs108/azs/miniFacebook.py (yours will be different, of course!),

and a link to your YouTube video demonstration.

Academic Honesty and Collaboration

Cooperation is recommended in understanding programming concepts and system features. But the actual solution of the assignments including all programming must be your individual work. For example, copying without attribution any part of someone else's program is plagiarism, even if you have modified the code. The University takes acts of cheating and plagiarism very seriously; first time violators are routinely suspended for a semester.