Schedule Builder

Problem 1: Deciding on a sorting mechanism for classes

Our program's main function is to be able to sort thru classes offered in order to build an optimal schedule, however we debated heavily on the most elegant way to do this, from a UI standpoint. We wanted a way for the user to be able to immediately narrow down the selection of classes wanted, based on certain criteria such as time start. However, we did not know if we wanted to do this using a dropdown menu with all the various options, or a field of checkable boxes. Each side had their pros and cons, as followed:

<u>Dropdown Menu:</u>

Pros:

- Significantly more compact
- Less complicated to code

Cons:

- Dropdown menu would be far too long
- Every possible scenario would have to be thought out and listed, or multiple menus would have to be created
    - "I want an hour for lunch" and "I want an hour for lunch and to start after 9" would have to be two separate options

      OR
    - Starting time, hours off, finishing time, or whatever limitations one could come up with would have to be separate dropdowns, invalidating both pro points
- Looks ugly

<u>Checkbox Field:</u>

Pros:

- Able to cover all possibilities far more elegantly than a dropdown
- More attractive looking
- No ridiculously long menus
- More space efficiency

Cons:

- Harder to code than the basic dropdown
- Takes up more space than the basic dropdown

We quickly realized that, while the basic dropdown would be the least complicated in terms of programming, the complicatedness of the options in the dropdown quickly made it not worth the effort saved. We decided to go with the checkbox field.

Problem 2: Deciding whether to use CRNs or subject search

In this scenario for our Schedule Builder application, our group had to decide whether to have users select courses from a subject drop down list, then the section, time, and professor. The second option would be to have user just either enter or select the class' CRN, which would fulfill the professor, time, and section requirements. For us this problem came down to a difference between ease of use for the user and ease of development for us developers.

<u>CRN option:</u>
Pros:

- User just has to enter CRN
- Less for us to code and in complexity

Cons:

- User might not know CRN
- People don't really know CRN's
- Less attractive to the user

<u>Subject Search</u>
Pros:

- Users can mess around with different classes
- Easier for user's to visualize course, professors, and time

Cons:

- More work for the user
- Higher complexity and more code for us to write

Decision: Our group ended up deciding to go with the traditional, subject search routine. We felt that this is a lot easier for the user, especially a casual user that is not familiar with CRN's.

Problem 3: Sorting by professors vs buildings

While selecting through courses for the schedule builder, we came across a few alternatives to allow users to simplify the course options. Our beginning idea was to have the user select which buildings they would prefer their classes to be in and then cut out all the professors that don't have class in that building. We then decided the best solution would be to allow the user to select times and professors that work with them.

<u>Buildings</u>
Pros:

- Specific cut off for walking distance
- Fast solution over picking a lot of professors

Cons:

- If the user doesn't care about building then it wastes time
- Cuts off important information for class

<u>Professors</u>

Pros:

- More specific over buildings
- The user has more choice over their classes

Cons:

- A lot of professors to sort through in general classes
- Not as fast at creating a schedule