

SCSU Course Registration Process Redesigned

Preamble:

To facilitate the presentation of this final report a couple of changes have been made to the overall structure to omit redundant information. First, since all four parts of this project contained a project description and requirements summary, below is the final revision of these two sections moved up from part 4. Additionally, since part 4 required the evaluation plan to be revisited prior to presenting the results and no changes were made to that plan from part 3, the initial evaluation plan section from part 3 has been removed.

Project Description:

The purpose of this project was to redesign the online course registration process at SCSU to improve the subjective user satisfaction, speed of task performance, and overall usability. The main issues addressed were the lack of informative feedback and unnecessary complexity. The main goal therefore was to present a simplified interface in which the same tasks could be done faster and with less ambiguity. The users of this system were SCSU students and the main tasks supported by the system are to allow the user to search and register for courses to add to their schedule.

Requirements Summary:

The system shall allow the user to search for courses to add to a schedule. Specifically, the system shall allow the user to specify course subject, number, and use this to query the course database for matches, returning a list of potential matches. Once the course item is found, the system shall notify the user of any issues (course no longer available, unmet prerequisites, ...etc.). Given at least a course subject, the system shall then allow the user to browse through search results and pick what courses they want to add to their schedule. Once courses are added the system shall also allow the user to remove any of those courses.

At any time the system shall allow the user to pull up the search criteria, modify any parameters, and do a new search (i.e. previously set filters shall persist unless modified by the user).

The system shall also allow the user to specify time preferences and availability. In terms of availability the system shall allow the user to specify any days or times of days (e.g. early/mid/late morning, afternoon, night or specific hours of the day) during which the user does not want to take classes. In terms of preference, the system shall allow the user to specify the same time parameters so that classes during those times would be prioritized in the search, but course options outside those preferences would still be shown.

Part I - Understanding the Problem

An overview of the problem is that the current course registration process for students at SCSU is painstakingly tedious and slow. Trying to coordinate available time slots for classes to have a complete schedule is almost always a mix and match game when you often have time constraints. For example, there might be time slots conflicts between classes you wish or need to take, or time slots might interfere with other time commitments you might have. At times you might need to consider taking a different class because a class you planned to take isn't available at suitable times that fit in your schedule. Given any of these constraints you're then forced to go back and forth through various pages (classes you're registered for so far, search page, list of available classes, class description pages...etc). A better interface or system is necessary to allow students to quickly register for classes given any time constraints they might have as well as to incorporate preset curriculum pathways as best as possible given the student's academic progress.

Important characteristics of users of the system are SCSU students and any time constraints (extracurricular time commitments) they might have. Student's current progress in their curriculum, such as just beginning (freshman) or finishing (junior), are also important considerations since availability of courses is prioritized for finishing students.

Important characteristics of the tasks performed by users includes checking preset curriculum pathways (that is, curriculum completion milestones set by the department for timely completion of the curriculum), curriculum requirements fulfilled and unfulfilled, available time slots for courses being offered, and access to course and professor info.

Important characteristics of the task environment involve having student users looking for and registering for classes from anywhere they would commonly carry out such task, such as their living quarters or library. For the purpose of this project it would be done in a classroom or library setting.

A simple structured task analysis of the problem in one of the forms described in the textbook would be observing and interviewing students while they carry out a sequence of tasks. One key aspect of this analysis would be to identify the tasks' frequency and purpose to eliminate redundancy. This will then lead to a reduced number of tasks and memory load for the user.

An analysis of the existing system would include looking at its strong points and deficiencies. Some of the strong points of the current design is that preliminary information a student might need is readily available and students can view their current schedule at any time. Another strong point is that various tabs can be opened to access various sources of information. The deficiencies are in how information is too spread out, too many pages to navigate, information is too plain, and the interface makes little use of signifiers and other standard guidelines and principles for good design.

The larger social and technical system impact of the design is that students will be able to better evaluate their progress and options while registering for classes. With a speedier, more informative, and easier to use design students would spend less time asking their advisors questions about the current system and classes to take.

The initial list of usability criteria that will be used in the evaluation of the design includes consistency with guidelines and several principles for good design from Schneiderman's book. Some of the guidelines would be clear rules for interface navigation, proper organization of information on display and facilitation of data entry with minimal memory load, and methods used for getting the user's attention (colors, sizes, and fonts). Some of the principles would be timing more frequent tasks so that emphasis is put on making those tasks faster. Adherence to these guidelines and principles would be done by interviewing users as well as timing tasks and noting task frequencies.

Implications of what I've learned is that in an effort to present users with more options an interface can become overwhelming. Lack of proper signifiers make the process even more prolonged as the user has to spend more time finding and interpreting what they need. The other thing is how adaptable we are to bad designs, to the point where if we're not beginners it's hard to recognize a bad design if it's not for guidelines and principles. Task complexity and user

adaptability will be highly considered for the eventual design that will improve the existing system for course registration at SCSU.

Part II - Design Alternatives

Design Space:

What Requirements may be difficult to realize?

In particular, the requirement that may prove to be the most difficult to realize might be the issue notification subsystem as the system would have to check various parameters, prior to presenting the user with search results for such foreseeable issues.

What are some tradeoffs that you should or did explore?

Some tradeoffs that may need to be explored is how much filtering the user is allowed to use. That is, how many parameters the user is allowed to specify in their search criteria. For one, allowing the user to specify just a course number might be more convenient, but the search query might return a longer list. Another tradeoff that has to do with filtering would be if partial information is allowed, such as partial title, course number, or subject name. With partial information the system would need to search for inexact matches, whereas searching for exact matches would be easier.

Another tradeoff would be how much information is displayed in a single page/window. More information means less navigating around, but too much and could be overwhelming for the user. Too little information would then result in having to navigate through more windows/pages, which is one of the problems with the system currently in place.

Which tasks will be easiest to support? What are the hardest?

The task easiest to support should be specifying search specifications (filters), so therefore searching for a course. None should present themselves to be any harder to support, but relative to search, it would have to be registration.

Design Summary:

The design alternatives presented here were all constructed with five ideas in mind, increase usability, improve user satisfaction, provide more informative feedback, simplify the registration process to increase speed of task performance, and ultimately abide by the eight golden rules of interface design laid out by schneiderman. Aside from these three designs, no other alternatives were considered due to time constraints. These three designs were chosen however to experiment with different degrees of changes, interaction styles, guidelines and principles. Of the three designs presented here, each deviates further from the design currently in place for the SCSU registration process. The current design is one that primarily uses a navigation and menu selection with form fill-in interaction style.

Design #1 deviates the least, particularly it leans the most towards familiarity with the current system for a minimal learning curve, and yet, with enhancements to increase usability and promote the four other ideas mentioned earlier. The search page for example now contains the option to type all search items so experienced users can quickly type and tab through the search form. A new user could still scroll through each search item's options and click away. However, design #1's highlight is primarily the informative feedback for issues that come up which should improve speed of task performance as users don't waste time on actions leading to these issues.

Design #3 deviates the most, it pushes for a direct manipulation / visual representation interaction style with an emphasis on spatial mapping and differentiation of elements by color. Perhaps however, the highlight of this design is in the recommender system that would take a list of courses and construct a set schedules for the user to browse through and select one.

Design #2 was meant to hit a balance between familiarity with the current system, but leaning more towards visual representation, particularly in course results selections. The highlight of this design however is its single page dynamic implementation of the single-page application (SPA) interaction design pattern which makes the user's experience more fluid and uninterrupted.

Overall, the key enhancements provided for all designs are consistency, informative feedback, and facilitation of data entry. In particular for example, search queries persist through the user's session unless the user resets it. In the current system once a search query is committed, the data entered as search criteria is erased; in the case that an user just wants to

slightly modify their search criteria they would have to re-enter every part of it again. Therefore this design enhancement doesn't punish the user for their actions and rather it permits easy reversal of their actions and keeps them in control.


Of note is how the proposed system should work. The key idea here is to search for the courses you want to register for. During the search you're to provide the criteria by which the search is to be done. You're then presented with course options and the end-goal is to have registered for a complete schedule of courses. Throughout this process if issues arise, primarily due to time conflicts the user might have to do multiple searches for alternative/compatible courses and often check the week layout of their schedule to see what times of their weeks are available.

The Designs:

Design #1


This design is meant to be a minimalistic modification of the current design with only slight enhancements to increase usability, user satisfaction, experience and feedback. In particular it adds more signifiers to minimize labels. Informative feedback is increased by notifying the user of their actions and issues that arrive in a human friendly manner rather than an error code.

Current Schedule

Subject	Course #	Section	Cred	Title	Days	Time	
CSC	305	01	3.0	Computer Organization	MW	2:00pm-3:15pm	X
CSC	477	01	3.0	Fundamentals of Data Mining	MW	3:25pm-4:40pm	X
CSC	324	02W	3.0	Computer Ethics	M	7:35pm-10:05pm	X
CSC	465	01	3.0	Communications and Networks	TR	1:50pm-3:05pm	X
CSC	334	01	3.0	Human Computer Interaction	TR	11:00am-12:15pm	X
CSC	499	01W	3.0	Independent Study & Research	n/a	n/a	X

submit

Reset

Search

Total Credit Hours: 18.0

Maximum Hours: 18.0

Use the selection options to search for courses. You may choose at least a subject and/or an attribute type to begin a search.

Subject:

Attribute Type:

Course #:

Instructor:

Mon	Tue	Wed	Thur	Fri
<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

start time: :

end time: :

Total Credit Hours: 18.0

Maximum Hours: 18.0

Sections Found

	Subject	Course #	Section	Cred	Title	Days	Time	Instructor	Date	Location
<input type="checkbox"/>	ACC	200	01	3.0	financial accounting	TR	8:10am-9:25am	Walker	8/29-12/16	SB 020A
<input type="checkbox"/>	ACC	200	02	3.0	financial accounting	TR	12:25pm-1:40pm	Phillips	8/29-12/16	SB 020B
<input type="checkbox"/>	ACC	200	03	3.0	financial accounting	MWF	10:10am-11:00am	Liu	8/29-12/16	SB 020B
<input type="checkbox"/>	ACC	200	04	3.0	financial accounting	MWF	11:10am-12:00pm	Liu	8/29-12/16	SB 020B
<input checked="" type="checkbox"/>	ACC	200	05	3.0	financial accounting	MW	3:25pm-4:40pm	Engel	8/29-12/16	SB 020B
						MW	5:00pm-6:15pm	Krauss	8/29-12/16	EN B118

time conflict

ACC 200 - 05 conflicts with your current schedule (CSC 477 - 01)

Schedule

	Monday	Tuesday	Wednesday	Thursday	Friday
9am					
11am					
12pm		CSC 334-01 11:00am-12:15pm MO 113 Kim		CSC 334-01 11:00am-12:15pm MO 113 Kim	
1pm					
2pm	CSC 305-01 2:00pm-3:15pm MO 120 Elahi	CSC 465-01 1:50pm-3:05pm JE 139B Elahi	CSC 305-01 2:00pm-3:15pm MO 120 Elahi	CSC 465-01 1:50pm-3:05pm JE 139B Elahi	
3pm					
4pm	CSC 477-01 3:25pm-4:40pm MO 113 DaPonte		CSC 477-01 3:25pm-4:40pm MO 113 DaPonte		
5pm					
6pm					
7pm					
8pm	CSC 324-02W 7:35pm-10:05pm MO 120 Volkman				
9pm					

go back

Scenario written from a user's perspective:

Kanou is a SCSU student ready to register for his courses for an upcoming semester. Kanou lands on the registration page and sees his schedule thus far, which has all the classes he has added and now only needs one more. Kanou clicks the search button and is brought to the search page. In the search page kanou clicks the subject box and types "comp", the computer science subject is pulled up to the top of the scrollable menu, Kanou presses the tab key to autocomplete and selects the computer science subject and moves to the next search item (attribute type). Kanou skips the attribute type and clicks the instructor search item, a drop down list comes up showing only professors that teach computer science. Kanou clicks his favorite professor Chang Suk Kim. Kanou then clicks the course number item and a drop down list shows up with only one option, 334. Kanou clicks the course item and clicks the search button, which brings him to the search results page. In the search results page there is only one

course but there is no check box to select the course, in its place is a red exclamation point. Kanou hovers over the exclamation point and a tooltip (message box) pops up with the message "this course section is full!". Kanou, ends up repeating the same process by going back to the search page and trying out another course to add to his schedule.

Assessment of design

Advantages:

- Buttons layout is consistent in the bottom-left.
- Task process is similar to that of the current design so should lead to shorter learning curve for experienced users.
- Quickly remove courses from course schedule by clicking the red X in the corresponding row.
- Days and time is shown in course schedule page so user can have a last look before submitting changes / registering.
- Search items can be typed into for filtering long lists (such as the subjects list) or scrolled through. The advantage is to experienced users who might prefer to quickly type and tab along the search rather than slowly clicking away, enabling these frequent users to use shortcuts. Nonetheless, the drop down menus are there so users (in particular new users) can choose to click through the search items instead.
- In the search page, clicking a subject narrows down the course number list to only those related to that subject. The same goes for the instructor list, only showing instructors relevant to the selected subject (that is, the instructors that teach that subject).
- In the search box, if subject or attribute type are not selected, the course # is greyed out (hovering over it brings up a tooltip to notify the user that a subject must be selected first).
- The search results have a warning/caution/exclamation-point icon next to courses or course sections that can't be added to the shedule due to a present issue (such as a time conflict with courses already added to the schedule, missing pre-requisites,...etc.).

Disadvantages:

- There is still the need to open an additional browser tab to make another search if a current search needs to be put on hold due to a time conflict.

Degree of compliance with requirements:

The search criteria is complied with completely, it shows all the required filters and is persistent for the user's session unless the reset button is pressed. The design also complies with the time availability, but not the time preferences.

User Feedback:

Some students were given a short introduction and walkthrough through the current design they're familiar with and then a walkthrough to each of the 3 alternative designs presented here. This design was liked for its familiarity with the current design and yet with several tweaks that make it more useful like the tooltip in the search results section that notifies you of any issues before you even add the course rather than being notified in the current search section with some error code that is not always straightforward to decipher.

Design #2

This design is meant to enhance the registration process by keeping some similarities, but seemingly mending many of the annoyances in the current design. Primarily this design highlights the single page implementation, leaving behind the traditional multiple page layout in the current design.

Modify Search

▼

Total Credit Hours: 18.0Maximum Hours: 18.0

Current Schedule

Accounting

Subject	Course #	Section	Cred	Title	Days	Time	Instructor	Date	Location
CSC	305	01	3.0	Computer Organization	MW	2:00pm-3:15pm	Elahi	8/29-12/16	MO 120
CSC	477	01	3.0	Fundamentals of Data Mining	MW	3:25pm-4:40pm	DaPonte	8/29-12/16	MO 113
CSC	324	02W	3.0	Computer Ethics	M	7:35pm-10:05pm	Volkman	8/29-12/16	MO 120
CSC	465	01	3.0	Communications and Networks	TR	1:50pm-3:05pm	Elahi	8/29-12/16	JE 139B
CSC	334	01	3.0	Human Computer Interaction	TR	11:00am-12:15pm	Kim	8/29-12/16	MO 113
CSC	499	01W	3.0	Independent Study & Research	n/a	n/a	Hossain	8/29-12/16	n/a

schedule by day

reset

Submit

Modify Search

Total Credit Hours: 18.0

Maximum Hours: 18.0

Current Schedule:

Accounting

Subject	Course #	Section	Cred	Title	Days	Time	Instructor	Date	Location
ACC	200	01	3.0	financial accounting	TR	8:10am-9:25am	Walker	8/29-12/16	SB 020A
ACC	200	02	3.0	financial accounting	TR	12:25pm-1:40pm	Phillips	8/29-12/16	SB 020B
ACC	200	03	3.0	financial accounting	MWF	10:10am-11:00am	Liu	8/29-12/16	SB 020B
ACC	200	04	3.0	financial accounting	MWF	11:10am-12:00pm	Liu	8/29-12/16	SB 020B
ACC	200	05	3.0	financial accounting	MW	3:25pm-4:40pm	Phillips	8/29-12/16	SB 020B
ACC	200	06	3.0	financial accounting				12/16	EN B118

time conflict

ACC 200 - 05 conflicts with your current schedule (CSC 477 - 01)

schedule by day

Total Credit Hours: 18.0

Current Schedule:

Subject	Course #	Section	Cred	Title	Days	Time	Instructor	Date	Location
ACC	200	01	3.0	financial accounting	TR	8:10am-9:25am	Walker	8/29-12/16	SB 020A
ACC	200	02	3.0	financial accounting	TR	12:25pm-1:40pm	Phillips	8/29-12/16	SB 020B
ACC	200	03	3.0	financial accounting	MWF	10:10am-11:00am	Liu	8/29-12/16	SB 020B
ACC	200	04	3.0	financial accounting	MWF	11:10am-12:00pm	Liu	8/29-12/16	SB 020B
ACC	200	05	3.0	financial accounting	MW	3:25pm-4:40pm	Phillips	8/29-12/16	SB 020B
ACC	200	06	3.0	financial accounting				12/16	EN B118

schedule by day

Modify Search

Total Credit Hours: 18.0 Maximum Hours: 18.0

Current Schedule

Accounting

Subject	Course #	Section	Cred	Title	Days	Time	Instructor	Date	Location
ACC	200	01	3.0	financial accounting	TR	8:10am-9:25am	Walker	8/29-12/16	SB 020A
ACC	200	02	3.0	financial accounting	TR	12:25pm-1:40pm	Phillips	8/29-12/16	SB 020B
ACC	200	03	3.0	financial accounting	MWF	10:10am-11:00am	Liu	8/29-12/16	SB 020B
ACC	200	04	3.0	financial accounting	MWF	11:10am-12:00pm	Liu	8/29-12/16	SB 020B
ACC	200	05	3.0	financial accounting	MW	3:25pm-4:40pm	Engel	8/29-12/16	SB 020B
ACC	200	06	3.0	financial accounting	MW	5:00pm-6:15pm	Krauss	8/29-12/16	EN B118

schedule by day
reset
add to schedule

Total Credit Hours: 18.0

Current Schedule

Subject	Course #	Section	Cred
ACC	200	01	3.0
ACC	200	02	3.0
ACC	200	03	3.0
ACC	200	04	3.0
ACC	200	05	3.0
ACC	200	06	3.0

Subject Accounting

Attribute Type

course # 200

instructor

Days:

Mon	Tue	Wed	Thur	Fri
<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

start time

end time

Date	Location
8/29-12/16	SB 020A
8/29-12/16	SB 020B
8/29-12/16	SB 020B
8/29-12/16	SB 020B
8/29-12/16	SB 020B
8/29-12/16	EN B118

schedule by day
add to schedule

Scenario written from a user's perspective:

Zerou is an SCSU student ready to register for the courses in the upcoming semester. When Zerou enters the registration page, he sees a selected tab with an empty schedule. Above the empty schedule is a large search button with an arrow pointing down. He clicks the search button and a box drops down, the arrow points up. Most fields in the search form are drop down lists, Zerou uses them to quickly find what he needs which gets narrowed down dynamically the more fields he fills. Zerou clicks the arrow pointing up in the search box and the

box recoils up. A loading circle comes up and eventually a new tab shows up with the search results. Zerou clicks the classes to add to his schedule which get highlighted and then clicks the "add to schedule" button. Zerou clicks the search button again which now says "modify search" and the search box drops down. Zerou makes the appropriate changes, once again they're shown in the search tab, Zerou selects the desired courses and adds them to his schedule. On the last search, the search results are greyed out. Zerou hovers over the row and clicks it noticing it doesn't get highlighted like the previous ones but a dialog pops up telling the user there is a time conflict with his current schedule. Zerou closes the dialog box and clicks the "schedule by day" button. A week layout of his schedule opens up and Zerou sees where the time conflict is. Deciding it doesn't matter, Zerou closes the image and clicks on the current schedule tab. His current schedule is shown and on the bottom there is a "submit" button. Zerou clicks the submit button. A confirmation page comes up with the message "registration is complete!".

Assessment of design

Advantages:

- Everything is done in one interactive page without having to go to multiple pages.
- Buttons layout is consistent in the bottom.
- Table view of schedule or search results is consistent throughout the whole process.
- Location, days and time are identical to search results so user can have a last look before submitting changes / registering.
- The search clickable tab pulls down a search box to make a search rather than going to a separate page.
- Search items can be typed into for filtering long lists (such as the subjects list) or scrolled through (drop down menu). The advantage is to experienced users who might prefer to quickly type and tab along the search rather than slowly clicking away. Nonetheless, the drop down menus are there so users can choose to click through the search items instead.
- In the search box, clicking a subject narrows down the course number list to only those related to that subject. The same goes for the instructor list, only showing instructors relevant to the selected subject (that is, the instructors that teach that subject).

- In the search box, if subject or attribute type are not selected, the course # is greyed out (hovering over it brings up a tooltip to notify the user that a subject must be selected first).
- In the search results courses or course sections can be selected by simply clicking its corresponding row. The row changes color to signify selection to the user and the “add to schedule” button enables itself (which was previously greyed out).
- Search results for which there is an issue are greyed out, clicking them or hovering over them brings up a message box with a description of the issue for why that course or course section can’t be added to the schedule, intuitively letting the user know how to fix it.

Disadvantages:

- A potentially longer learning curve since it deviates a bit from the current design, though for a truly experienced user this should increase performance.

Degree of compliance with requirements:

The search criteria is complied with completely, it shows all the required filters and is persistent for the user’s session unless the reset button is pressed. The design also complies with the time availability, but not the time preferences.

User Feedback:

Some students were given a short introduction and walkthrough through the current design they’re familiar with and then a walkthrough to each of the 3 alternative designs presented here. This design was liked the most out of all 3 designs for its all in one page approach with the ability to pull up things as needed and quickly tabbing to multiple searches without having to open new browser tabs and copying and pasting the URL. Another feature liked the most was the message box in the search results section that notifies the user of any issues before the user even adds the course rather than being notified in the current search section with some error code that is not always straightforward to decipher.

Design #3

This design is meant entirely to implement the direct manipulation interaction style. Every part is self explanatory without having to read the instructions just based on the purpose

select times to prioritize or exclude:


(click preferred times, double click to exclude times from search,
right click to unselect)


	Mon	Tue	Wed	Thu	Fri
8AM					
10AM					
12PM					
2PM					
5PM					
7PM					

preferred times

excluded times

finish

 preferred times

 excluded times

finish

Search for courses to add:
(search by course subject, number, title, and/or instructor)

CSC		
CSC 153 - Computer Programming	TBA	3
CSC 207 - Computer Systems	Clark	3
CSC 212 - Data Structures	Seyed	3

prioritize

!	Course	Instructor	Cred	
<input checked="" type="checkbox"/>	CSC 305 - Computer Organization	Elahi	3	X
<input checked="" type="checkbox"/>	CSC 465 - Comm & Networks	Elahi	3	X
<input type="checkbox"/>	CSC 334 - Human Comp. Interaction	Kim	3	X
<input type="checkbox"/>	CSC 324 - Computer Ethics	Volkman	3	X
<input checked="" type="checkbox"/>	CSC 499 - Ind. Study & Research	Hossain	3	X

finish

Change Time
Availability

Search for courses to add:
(search by course subject, number, title, and/or instructor)

CSC 207 - Computer Systems, Clark		
CSC 153 - Computer Programming	TBA	3
CSC 207 - Computer Systems	Clark	3
CSC		

prioritize

!	Course	Instructor	Cred	
<input checked="" type="checkbox"/>	CSC 305 - Computer Organization	Elahi	3	X
<input checked="" type="checkbox"/>	CSC 465 - Comm & Networks	Elahi	3	X
<input type="checkbox"/>	CSC 334 - Human Comp. Interaction	Kim	3	X
<input type="checkbox"/>	CSC 324 - Computer Ethics	Volkman	3	X
<input checked="" type="checkbox"/>	CSC 499 - Ind. Study & Research	Hossain	3	X

finish

Change Time
Availability

Sorry, all sections for CSC 207 are full!

Browse "saved for later" list

Search for courses to add:
 (search by course subject, number, title, and/or instructor)
 ▼

Change Time Availability

	Monday	Tuesday	Wednesday	Thursday	Friday
morning					
	CSC 465-01 1:50pm-3:05pm		CSC 465-01 1:50pm-3:05pm		
afternoon	CSC 305-01 2:00pm-3:15pm		CSC 305-01 2:00pm-3:15pm		
	CSC 477-01 3:25pm-4:40pm	CSC 334-01 3:00am-4:15pm	CSC 477-01 3:25pm-4:40pm	CSC 334-01 3:00am-4:15pm	
evening					

Save for later

select

Browse "saved for later" list

Search for courses to add:
 (search by course subject, number, title, and/or instructor)
 ▼

Change Time Availability

	Monday	Tuesday	Wednesday	Thursday	Friday
morning					
		CSC 334-01 11:00am-12:15pm		CSC 334-01 11:00am-12:15pm	
afternoon	CSC 305-01 2:00pm-3:15pm	CSC 465-01 1:50pm-3:05pm	CSC 305-01 2:00pm-3:15pm	CSC 465-01 1:50pm-3:05pm	
	CSC 477-01 3:25pm-4:40pm		CSC 477-01 3:25pm-4:40pm		
evening					
	CSC 324-01 7:35pm-10:00pm				

CSC 477-01 - Data Mining
 Dr. DaPonte
 seat availability: 25/30
 3:25pm - 4:40pm
 Morrill Hall 113

X

Save for later

select

Scenario written from a user's perspective:

Anon is a SCSU student ready to register for classes for the upcoming semester. Anon enters the registration page and is prompted to select time availability and preferences. Anon doesn't have any particular time preferences or availability issues so clicks on the finish button. A confirmation dialog box pops up asking Anon if she is sure, with a yes and no button. Anon clicks the yes button. Next, Anon is prompted for classes to include in a search. Since Anon had previously talked with an academic advisor and determined a core group of classes to take, along with some alternatives, Anon begins to type into the search bar and selects the classes to include one after the other. With all the classes Anon wants listed, Anon prioritizes those classes she must take next semester by toggling the checkbox next to each class. Next, Anon clicks the finish button. A confirmation dialog box pops up asking Anon if she is sure, with a yes and no button, Anon clicks the yes button. The search box goes away and in its place is a loading circle. After a couple of seconds Anon is presented with a schedule containing the classes that were prioritized, but none of the additional ones. Anon click the arrow buttons at the sides of the screen to navigate between different schedules until finally finding an ideal one and clicks on the select button. A confirmation dialog box pops up asking Anon if she is sure, with a yes and no button, Anon clicks the yes button. Another dialog box pops up to inform Anon that registration is complete. Anon exists the registration page and is done registering for courses.

Assessment of design

Advantages:

- Everything is done in one interactive page without having to go to multiple pages.
- Buttons layout is consistent in the bottom and top right corner.
- Various schedule combination options are shown to the user to browse through and select one from.
- There is a selections list to compare several schedule combinations if multiple searches are made.
- Can select multiple courses from to construct schedules with or without all the selected courses.
- Can prioritize course selections so that they're given priority over non-prioritized courses for any conflict issues that might arise.

- Can specify time preference and time availability.
- The search clickable tab pulls down a search box to make a search rather than going to a separate page.
- There is only a search bar to type into (like google search bar) and a drop down list is dynamically created as the user types. The advantage is to experienced users who might prefer to quickly type rather than slowly clicking away.
- Search results for which there is an issue brings up a message box if selected with a description of the issue for why that course or course section can't be added to the schedule, intuitively letting the user know how to fix it.
- Clicking a class block in a schedule brings up a message box with more info on that course.
- Conveniently assists user through their task.

Disadvantages:

- A completely different design from the current design could cause an initial longer learning curve for the user.
- Less familiarity with the current design.
- The hardest to implement.
- Could be limited by screen size and therefore screen resolution.

Degree of compliance with requirements:

The search criteria is complied with completely, it shows all the required filters and is persistent for the user's session unless the reset button is pressed. The design also complies with the time availability and preferences.

User Feedback:

Some students were given a short introduction and walkthrough through the current design they're familiar with and then a walkthrough to each of the 3 alternative designs presented here. This design was liked for its in one page approach with the ability to pull up things as needed and quickly making a search similar to google searches. Though surprisingly the majority thought the design was overkill for its purpose.

Requirements Changes:

Some changes to the requirements were made primarily because they were too precise to design #3. The changes made generalized some of the requirements so they were more about the task completion of the user rather than a particular design. For example, rather than requiring the system to provide the user with various schedules, it was changed to providing the user with search results (a list of courses and/or course sections to pick from). Branching from this was also the requirement to allow the user to navigate through these various schedules which was removed as it was too specific to design #3.

Part III - System Prototype

Prototype Description:

This high-fidelity prototype was implemented to address the core requirements (e.g. adding/removing/searching/selecting courses); a simplified form of some of the requirements (e.g. issues and specification of time availability); and provide a simple, dynamic, single page interface for the user to interact with. The prototype supports a complete flow of typically adding several courses to the schedule and doing multiple searches, with some basic data available to work with (back-end functionality). Actions are emphasized through color coding (e.g. buttons to press, selecting records) to make it easier for the user to isolate these mappings and signifiers. And lastly, informative feedback was implemented all throughout so the user is always receiving some feedback from their interactions with the interface.

The following image shows the main screen the user would see when arriving at the registration portal (in this case the user is done and ready to submit their schedule).

Current Schedule

Search Results

Subject	CourseNo	SectionNo	Credits	Title	Days	Time	Instructor	Date	Location
CSC 101 01	3,000	Intro to Computers & Applic	MWF	10:10AM-11:00AM	Chang Suk Kim	08/29-12/16	MO 8		
CSC 152 03	3,000	Computer Programming I	MWF	11:10AM-12:00PM	Chang Suk Kim	08/29-12/16	JE 139A		
CSC 334 01	3,000	Human Computer Interactions	TR	11:00AM-12:15PM	Chang Suk Kim	08/29-12/16	MO 113		
CSC 200 08	3,000	Info Mgmt/ProductivitySoftware	TR	04:45PM-06:00PM	Ronald R. Benner	08/29-12/16	MO 13		

Upon clicking the search button the following dialog window pops up and the background is greyed out, this is the search form for querying the database. The user can tab through each field and type out the options they want or click the drop down lists and select their options/filters. Search can be carried out by pressing enter or clicking the search button. Search can be cancelled by pressing the cancel button, the X on the top right corner, or anywhere in the grey area. This approach facilitates faster searching for experienced users and a more direct selection for beginner users.

Current Schedule

Search Results

Subject	CourseNo	SectionNo	Credits	Title	Days	Time	Instructor	Date	Location
CSC 101 01	3,000	Intro to Computers & Applic	MWF	10:10AM-11:00AM	Chang Suk Kim	08/29-12/16	MO 8		
CSC 152 03	3,000	Computer Programming I	MWF	11:10AM-12:00PM	Chang Suk Kim	08/29-12/16	JE 139A		
CSC 334 01	3,000	Human Computer Interactions	TR	11:00AM-12:15PM	Chang Suk Kim	08/29-12/16	MO 113		
CSC 200 08	3,000	Info Mgmt/ProductivitySoftware	TR	04:45PM-06:00PM	Ronald R. Benner	08/29-12/16	MO 13		

Subject:

Course #:

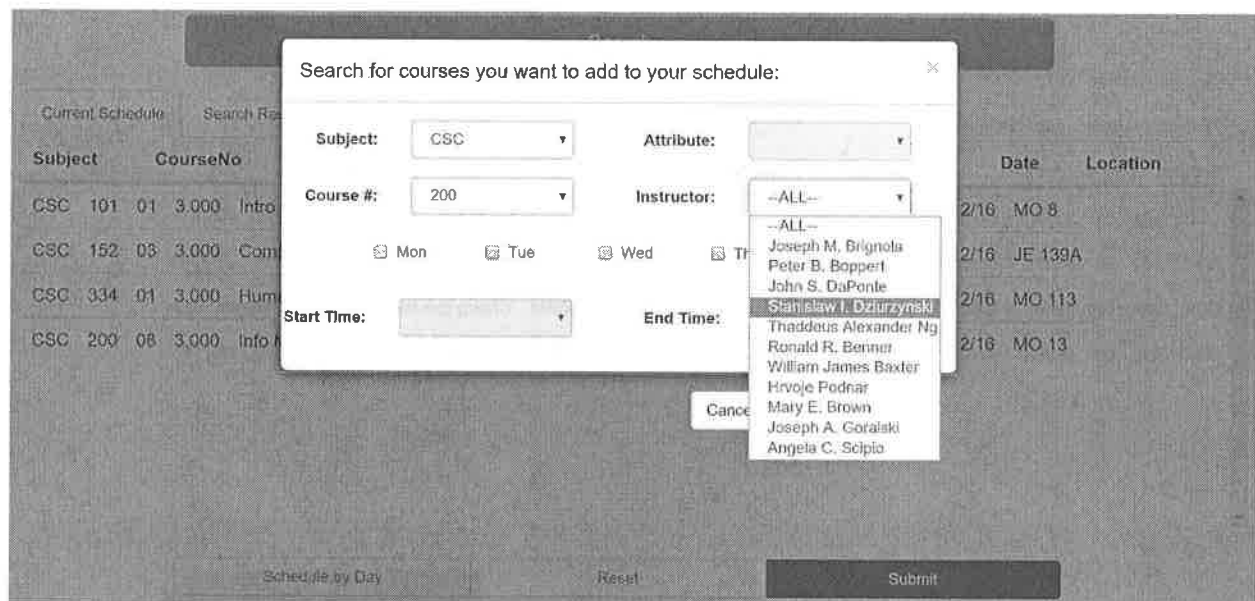
☒ Mon
 ☐ Tue
 ☐ Wed
 ☐ Thu
 ☐ Fri

Start Time:

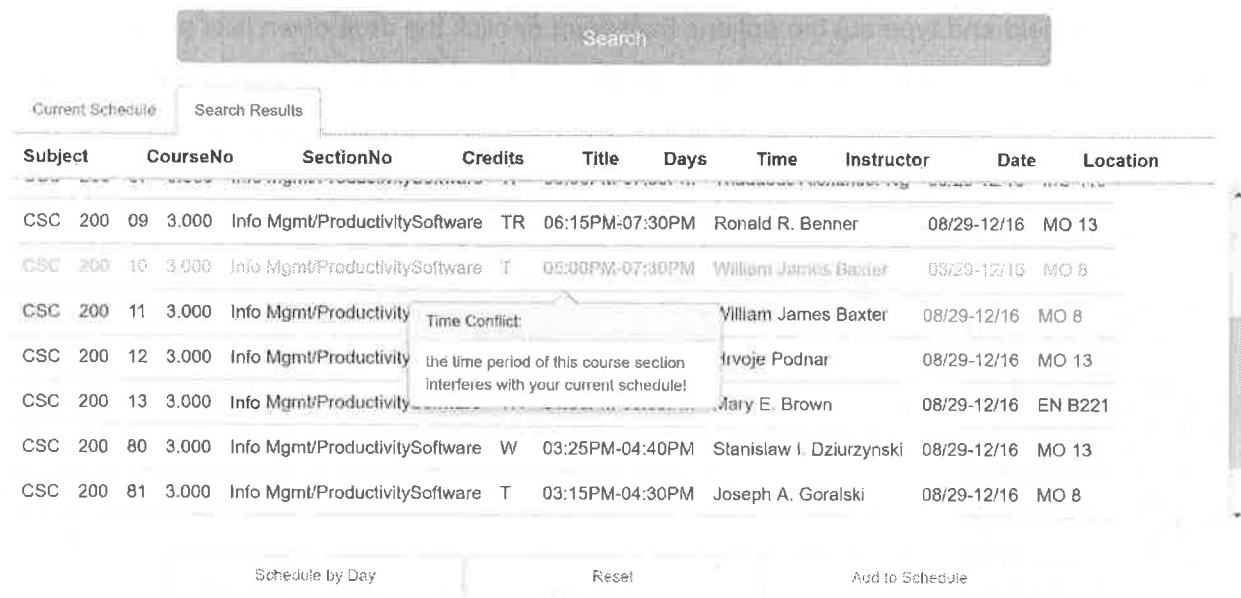
Attribute:

Instructor:

End Time:



Search results are greyed out when there are conflicting issues, hovering over the record will show a popup notification to the user.



Hovering over a selectable record will highlight it light blue.

Current Schedule
Search Results

Subject	CourseNo	SectionNo	Credits	Title	Days	Time	Instructor	Date	Location
CSC	200	01	3.000	Info Mgmt/ProductivitySoftware	MWF	08:10AM-09:00AM	Joseph M. Brignola	08/29-12/16	MO 13
CSC	200	02	3.000	Info Mgmt/ProductivitySoftware	MWF	09:10AM-10:00AM	Joseph M. Brignola	08/29-12/16	MO 13
CSC	200	03	3.000	Info Mgmt/ProductivitySoftware	MWF	10:10AM-11:00AM	Peter B. Boppert	08/29-12/16	MO 113
CSC	200	04	3.000	Info Mgmt/ProductivitySoftware	MWF	11:10AM-12:00PM	Peter B. Boppert	08/29-12/16	MO 113
CSC	200	05	3.000	Info Mgmt/ProductivitySoftware	MW	02:00PM-03:15PM	John S. DaPonte	08/29-12/16	MO 13
CSC	200	06	3.000	Info Mgmt/ProductivitySoftware	TR	11:00AM-12:15PM	Stanisław I. Dziurzynski	08/29-12/16	MO 8
CSC	200	07	3.000	Info Mgmt/ProductivitySoftware	W	05:00PM-07:30PM	Thaddeus Alexander Ng	08/29-12/16	MO 113
CSC	200	09	3.000	Info Mgmt/ProductivitySoftware	TR	06:15PM-07:30PM	Ronald R. Renner	08/29-12/16	MO 13

Clicking a selectable record will highlight the record a darker blue, enable the “add to schedule” button and change its color to emphasize what to do next.

Current Schedule
Search Results

Subject	CourseNo	SectionNo	Credits	Title	Days	Time	Instructor	Date	Location
CSC	200	01	3.000	Info Mgmt/ProductivitySoftware	MWF	08:10AM-09:00AM	Joseph M. Brignola	08/29-12/16	MO 13
CSC	200	02	3.000	Info Mgmt/ProductivitySoftware	MWF	09:10AM-10:00AM	Joseph M. Brignola	08/29-12/16	MO 13
CSC	200	03	3.000	Info Mgmt/ProductivitySoftware	MWF	10:10AM-11:00AM	Peter B. Boppert	08/29-12/16	MO 113
CSC	200	04	3.000	Info Mgmt/ProductivitySoftware	MWF	11:10AM-12:00PM	Peter B. Boppert	08/29-12/16	MO 113
CSC	200	05	3.000	Info Mgmt/ProductivitySoftware	MW	02:00PM-03:15PM	John S. DaPonte	08/29-12/16	MO 13
CSC	200	06	3.000	Info Mgmt/ProductivitySoftware	TR	11:00AM-12:15PM	Stanisław I. Dziurzynski	08/29-12/16	MO 8
CSC	200	07	3.000	Info Mgmt/ProductivitySoftware	W	05:00PM-07:30PM	Thaddeus Alexander Ng	08/29-12/16	MO 113
CSC	200	09	3.000	Info Mgmt/ProductivitySoftware	TR	06:15PM-07:30PM	Ronald R. Renner	08/29-12/16	MO 13

User receives some feedback on their action when adding a course to the schedule via a message box that pops up.

course added successfully!

Current Schedule Search Results

Subject	CourseNo	SectionNo	Credits	Title	Days	Time	Instructor	Date	Location
CSC	200	01	3.000	Info Mgmt/ProductivitySoftware	MWF	08:10AM-09:00AM	Joseph M. Brignola	08/29-12/16	MO 13
CSC	200	02	3.000	Info Mgmt/ProductivitySoftware	MWF	09:10AM-10:00AM	Joseph M. Brignola	08/29-12/16	MO 13
CSC	200	03	3.000	Info Mgmt/ProductivitySoftware	MWF	10:10AM-11:00AM	Peter B. Boppert	08/29-12/16	MO 113
CSC	200	04	3.000	Info Mgmt/ProductivitySoftware	MWF	11:10AM-12:00PM	Peter B. Boppert	08/29-12/16	MO 113
CSC	200	05	3.000	Info Mgmt/ProductivitySoftware	TR	11:00AM-12:15PM	Stanislaw I. Dziurynski	08/29-12/16	MO 8
CSC	200	07	3.000	Info Mgmt/ProductivitySoftware	W	05:00PM-07:30PM	Thaddeus Alexander Ng	08/29-12/16	MO 113
CSC	200	08	3.000	Info Mgmt/ProductivitySoftware	TR	06:15PM-07:30PM	Ronald R. Benner	08/29-12/16	MO 13
CSC	200	09	3.000	Info Mgmt/ProductivitySoftware	TR	06:15PM-07:30PM	Ronald R. Benner	08/29-12/16	MO 13
CSC	200	10	3.000	Info Mgmt/ProductivitySoftware	TR	06:15PM-07:30PM	Ronald R. Benner	08/29-12/16	MO 13

Schedule by Day Reset Add to Schedule

On the "current schedule" tab records are also highlightable and selectable. When a record is selected the remove button is shown and color coded to signify its use.

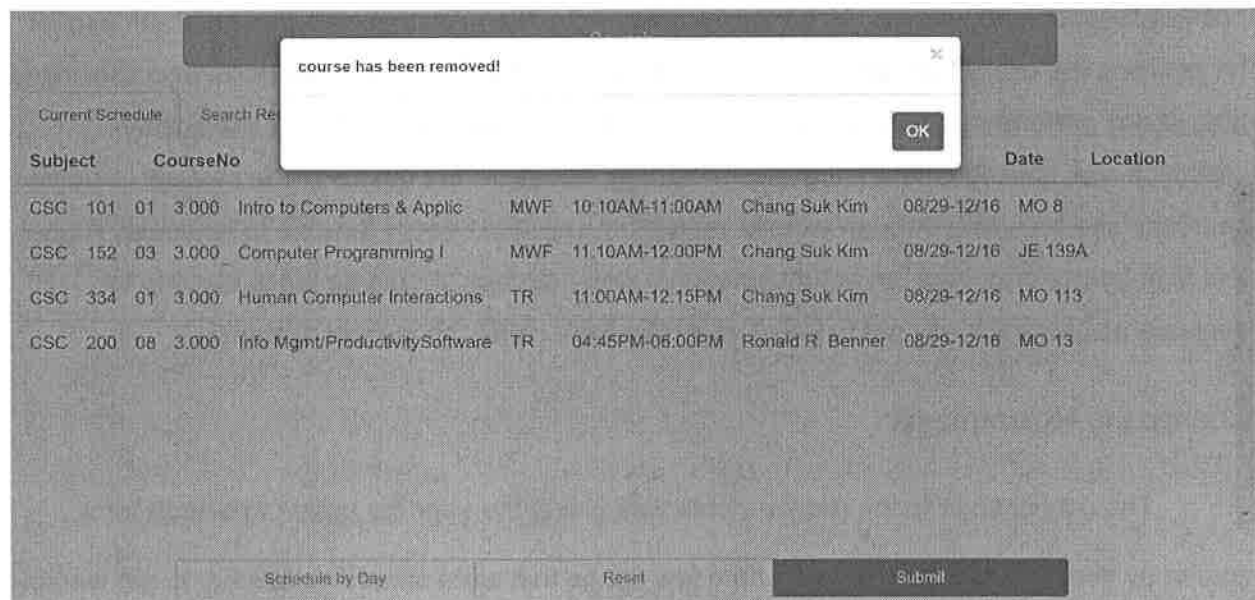
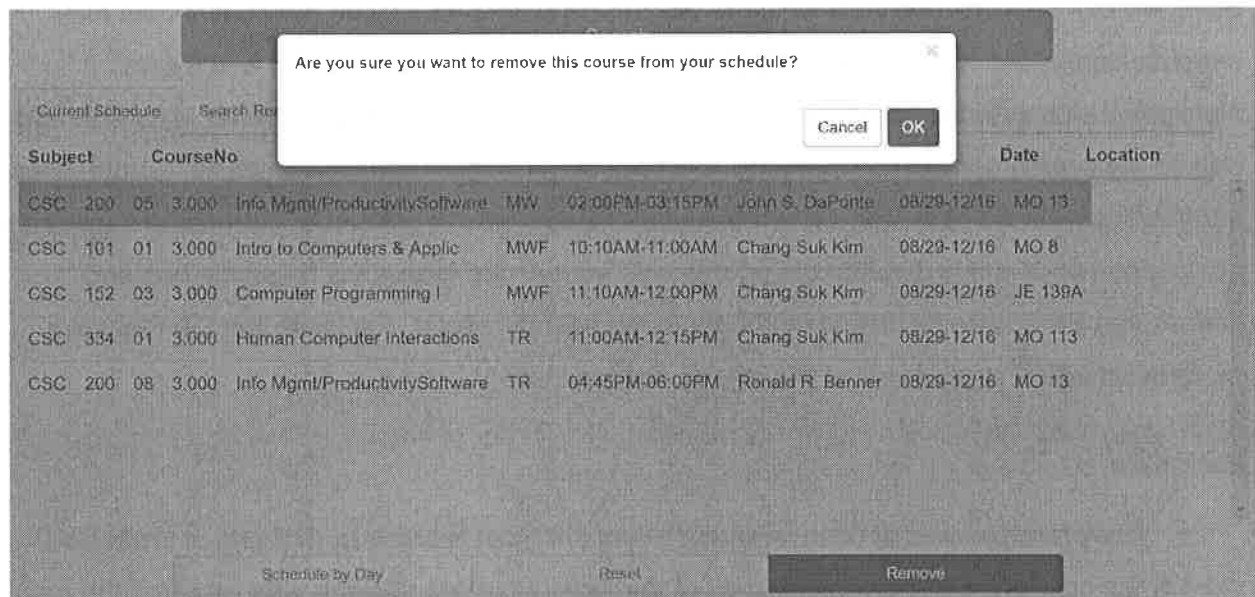
Search

Current Schedule Search Results

Subject	CourseNo	SectionNo	Credits	Title	Days	Time	Instructor	Date	Location
CSC	200	05	3.000	Info Mgmt/ProductivitySoftware	MW	02:00PM-03:15PM	John S. DePonte	08/29-12/16	MO 13
CSC	101	01	3.000	Intro to Computers & Applic	MWF	10:10AM-11:00AM	Chang Suk Kim	08/29-12/16	MO 8
CSC	152	03	3.000	Computer Programming I	MWF	11:10AM-12:00PM	Chang Suk Kim	08/29-12/16	JE 139A
CSC	334	01	3.000	Human Computer Interactions	TR	11:00AM-12:15PM	Chang Suk Kim	08/29-12/16	MO 113
CSC	200	08	3.000	Info Mgmt/ProductivitySoftware	TR	04:45PM-06:00PM	Ronald R. Benner	08/29-12/16	MO 13

Schedule by Day Reset Remove

A confirmation box shows up to confirm that the user wants to remove a record.



Scenario from user's perspective:

Jon is a SCSU student who is registering for courses for the upcoming semester. Jon enters the registration portal and clicks the large search button on the top of the screen. A small window appears in-front of the screen and the background is greyed out. Jon clicks the subject drop down list and selects CSC, then he clicks the course number drop down list and selects 101. At the bottom of the window there are three buttons, they're labeled "cancel", "reset", and "search", Jon clicks the search button. The window disappears and the regular page is no longer greyed out. The search results are shown in a table in the center of the page, Jon sees

there are four sections available for CSC 101, as he hovers his mouse over the results they become highlighted in blue. Jon clicks the first section with Chang Suk Kim which gets highlighted with a darker tone of blue. There is a group of buttons on the bottom, after Jon had clicked the course section he wanted to add, one of the buttons "add to schedule" caught Jon's attention as it turned blue. Jon clicked the now enabled "add to schedule" button and a message box popped up to notify Jon that the course was added. Jon closes the message box and notices that the course section is no longer in the search results. Jon clicks the tab that says "current schedule" and sees that the course section there.

Rationale:

This prototype was chosen because it was the most feasible to evaluate, it embodied most of the core principles and guidelines of the three alternate designs, and received the most positive feedback from users. As for the requirements, the system meets most of them except for notifying the user of unmet prerequisites or course no longer being available, and specifying time of day preferences. Perhaps all requirements could have been met if a low-fidelity prototype was chosen instead (e.g. paper-based), but given the nature of the system - involving searches and browsing through various records, it just didn't seem feasible to evaluate. Rather, this high-fidelity prototype, an actual functional web interface should be more feasible to evaluate as users actually get to use it as if they were using the current system.

Changes to requirements:

The only change to the requirements was giving the user the option to search for a course by title. This was simply one of the few things that were simply not priority to get working in the prototype for the essential functionality testing that will be done during evaluation.

Part IV - Evaluation

Overview:

To evaluate the design for adherence to the requirements that have been specified some techniques from *Designing the User Interface* by Schneiderman and et al. were used. These techniques were retrospective think-aloud, competitive usability testing, and timing speed of task performance (Schneiderman, CH.5).

Competitive Usability Testing (CUT)

This approach was used to determine if this design truly does perform better than the system that is currently in place. A set of benchmark tasks were carried out by the users on both the current system and on the prototype. Results from the prototype are compared to that of the current system.

The benchmark tasks that were used and time measured are the following:

1. Search for the Computer Organization course taught by Ataollah Elahi by specifying only the professor's name in the search criteria, select and add it to the schedule.
2. Reset search form and search for the course CSC 207 by specifying course number and professor (Elahi), select and add it to the schedule.
3. Search for the course CSC 321 by specifying subject and course number, select and add it to the schedule.
4. Search for the course CSC 334-01 by specifying subject and course number, hover over it, read popup, solve issue by removing the conflicting course section and adding a different course section.
5. Search for all courses offered in CSC on Mondays, Wednesdays, and/or Fridays, select a level 400 or 300 course and add it to the schedule.

The goal with these benchmark tasks was to start simple, get the user familiar with the interface, and evaluate its various features. For example:

- task 2 and 3 were used to get the user familiar with the search function as well as giving the observer insight into how useful it is to have a persistent search form as opposed to it resetting automatically every time it's opened (like in the current system).

- Task 2 additionally gave the observer some insight on how useful it is for search options to dynamically change as more filters are set and therefore narrow down available options (e.g. selecting course subject reduces available course number options to only those numbers relevant to that subject, selecting a course number will reduce available professors to only those that teach that course).
- Task 1 was used as insight into the function of just searching by professor name when the user can't remember the course number, but they know who teaches it.
- Task 2 and 4 combined served the purpose of putting the user in the scenario where there is an issue and going about solving it. This scenario was useful to see how easy it is to solve such issue and how useful/informative is the feedback given to the user.
- Lastly, task 5 was meant to put the user in the situation where they have some vague idea of what course they need to add, but don't have anything specific. This scenario was used to gain some insight into the ease of navigating through several search results.

CUT: Data & Analysis

The following data was collected from a total of 3 users who participated in the evaluation of this design:

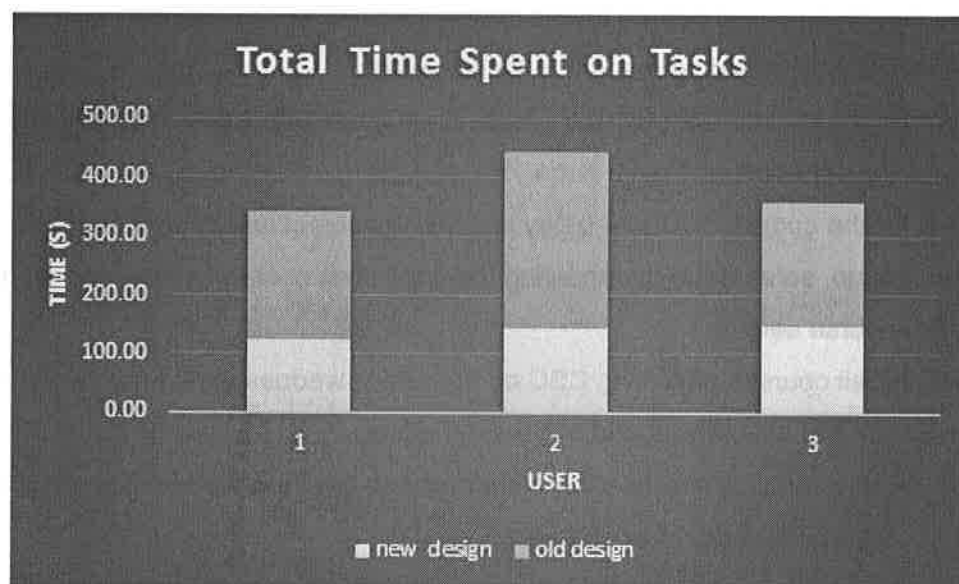


Figure 1. This graph shows the total time in seconds that each user spent on the whole set of benchmark tasks. The dark green is the data time spent on the current system design whereas the light green was the time spent on the new design presented here.

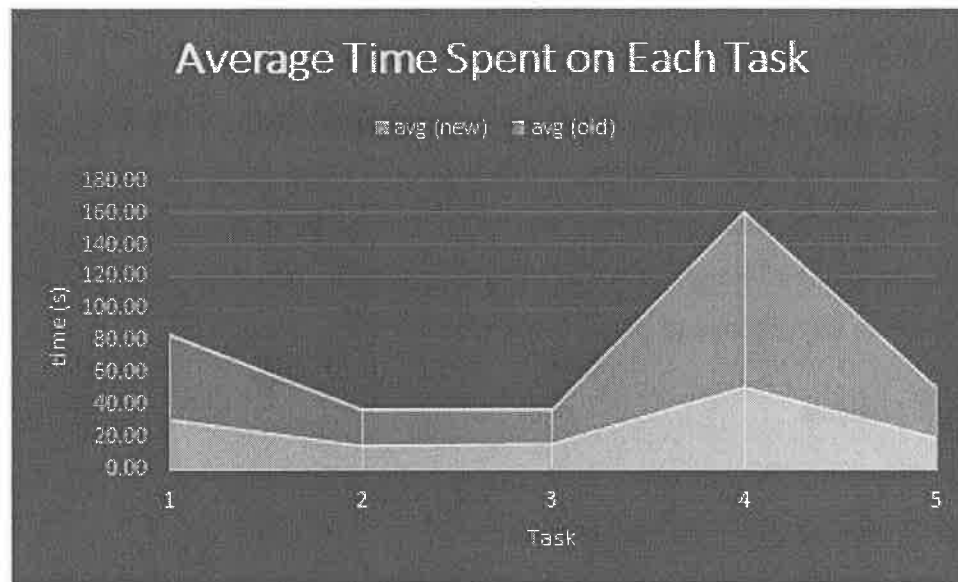


Figure 2. This graph shows the average time in seconds that users spent on the 5 benchmark tasks. The dark green is the time spent on the current system design whereas the light green was the time spent on the new design presented here.

Figure 1 and 2 indicate that the new design was successful in improving the overall speed of task performance. For task 1 and 4, it looks like speed of task performance was at least doubled in comparison to the current system design. One of the key aspects of this data is task 4 which tested a core issue with the current system design, dealing with time conflicts and having the system help the user resolve it quickly. The vast improvement in speed of task performance for task 4 with the new design is almost tripled in comparison to the current system design.

Also of note is the beginning of the test with task 1. Considering that each user first conducted the benchmark tasks on the current system design, it's hard to judge whether the improvement of the new design over the current system design is due to a better interface or whether the users were initially nervous from being observed.

CUT: Design Implications

Given these results the prototype does not need to be altered any further as it was a complete success, the speed of task performance was improved with the new design.

Retrospective Think-Aloud & Questionnaire (RTAQ):

This technique was used to get some insight into the user's satisfaction with each design, as well as any possible misinterpretations or errors they might have had. Generally, retrospective think-aloud was used to get as much information as the user could remember and articulate in terms of their thought process while carrying out the tasks, without actually interrupting them in the process of doing those tasks.

RTAQ: Data & Analysis

The following questionnaire was given (with a rating of 1 to 5 labeled as strongly disagree | disagree | no opinion | agree | strongly agree):

1. The new interface design was simpler to use than that of the current system.
2. The dynamic search was useful.
3. The tabs to quickly switch between current schedule to search results were useful.
4. The interface was self-explanatory.
5. The popup obtained from hovering over a greyed out result was helpful in solving the time conflict issue.

Afterwards, the users were asked the following open-ended questions:

6. Were you confused by any aspect of the new design? If yes, what was it?
7. With the new design, did you feel overwhelmed by the amount of information presented at any point? If so, why?
8. Was the interface design easier to navigate than that of the current system?

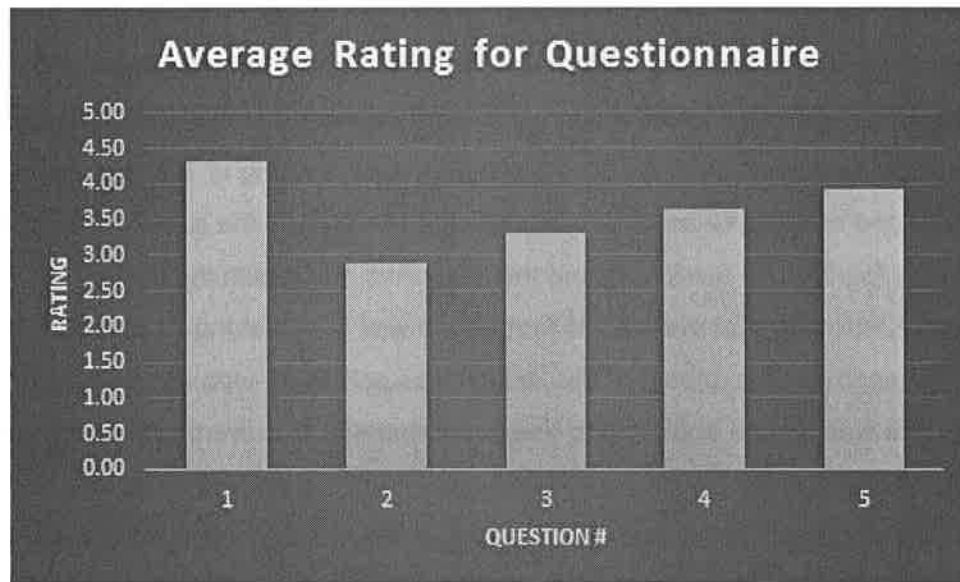


Figure 3. This graph shows the average rating obtained for each of the questions in the questionnaire.

As for the final questions the overall reactions/responses were:

Question 6

- No problem. (2)
- the popup for the time conflict wasn't detailed enough, would have been nice if it specified what was the course in my current schedule that was causing the time conflict rather than me looking at the time of the course I was trying to add and then scanning my current schedule to see which was the course it was conflicting with. Though, it's still better than what the current system offers. (1)

Question 7

- No. (3)

Question 8

- Yes. (1)
- The way the buttons changed color as well as the rows as I hovered over them felt very intuitive and self-explanatory, making it easy to know what was going on and what to do or click next. (1)
- Definitely. (1)

Figure 3 indicates that the dynamic search function of the interface wasn't very useful. Either the dynamic search function went unnoticed or its benefit over having all the information available (as such is the case with the current system design) was marginal. The tab switching function went also unappreciated/unnoticed with an average rating of 3.3, marginally useful at best. However, one of the core aspects of the design, which was the issue notification mechanism was found to be useful. Taking this response along with the performance on task as seen on figure 2 indicates that overall this feature did well in achieving its purpose. Lastly, figure 3 indicates that another core aspect of this design was achieved, with an average rating of 4.3, the new interface was simple and easy to use in comparison to current system design.

The responses to the open-ended questions indicate that overall the new design did not confuse the users and was rather self-explanatory. However, as noted by one user, the issue notification might not have been detailed enough. The users didn't feel overwhelmed by the amount of information, possibly indicating once again that the interface was simplified. Though it could have just been that the users are used to the amount of information due to the current system design. Lastly, the interface was deemed easier to navigate than the current system interface.

RTAQ: Design Implications

Given the concern raised by an user on how the time conflict notification wasn't detailed enough, I came to agree on the issue. The time conflict notification should contain detail on what course in the current schedule the time conflict is with so the user can more swiftly deal with it. Without a doubt having to look at the time blocks of every course in the current schedule has a toll on task performance. Therefore, this one thing that a new version of the prototype should address given more time to work on it. Though other than that, the results indicate that the design did well in accomplishing the goals of this project.

Critique and Summary:

The advantage of the evaluation conducted was to be able to get some measurements, but also feedback from the users themselves. Having just done measurements would have led to a conclusion that the design was a complete success. However, as the second technique indicates some improvements could have been made. This teaches the important lesson of not separating judgment on design from the actual users, user feedback is paramount. Measurements of performance alone aren't enough to determine how successful a design

solution is. A disadvantage of the evaluation however was that the questions asked to the users didn't seem to have done a good job in retrieving much information in regards to their thought process. Given more resources, such as time, more time should be spent on developing the questions asked to the users. This aspect teaches another lesson, in which that coming up with the proper questions to get feedback from users needs careful and thorough thinking as well as possibly reiteration and refinement after testing them with a couple of users. Overall however, the evaluation results positively indicate that the design did well in meeting the goals of improving speed of task performance and simplifying the registration process.

