Nora Allison (efa91)
Stephen LaPlante (laplansk)

CSE 403 Spring 2014
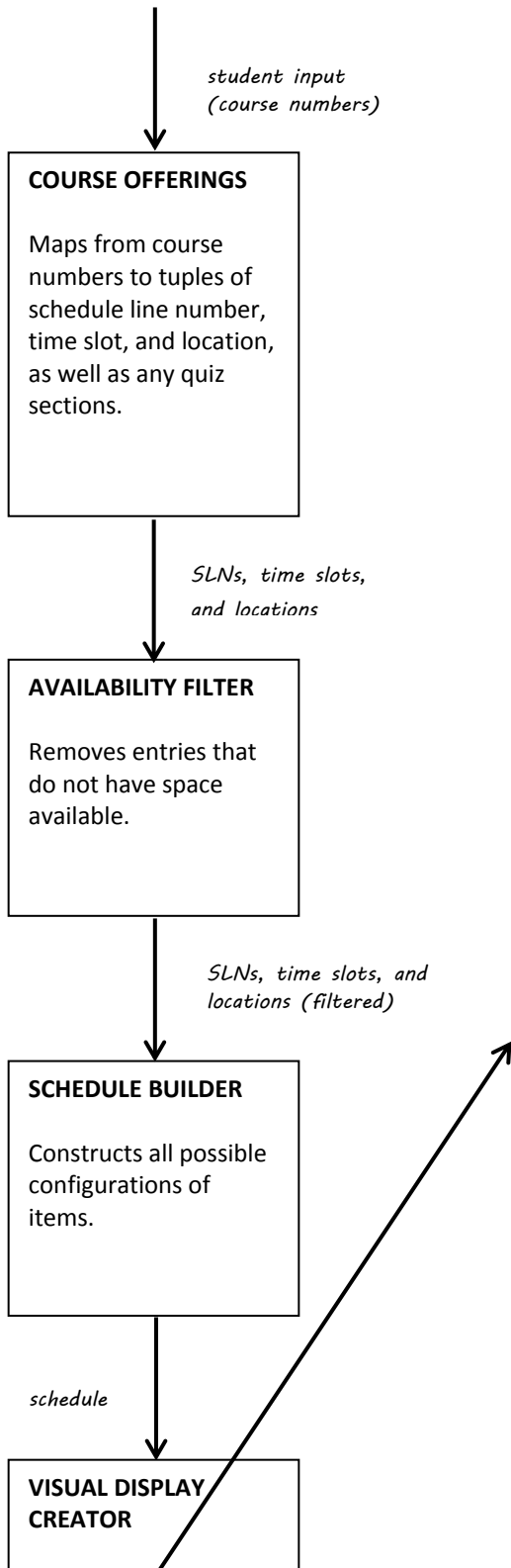Assignment 1 – Project Proposal

For most UW students, registration is a stressful process, often involving lots of loading of separate web pages and writing out potential schedules by hand. A student who wants to take courses X, Y, and Z in a given quarter must look up the schedule for each (which will likely have multiple possible lecture sections and quiz sections from which to choose), choose a combination of sections (each with its own schedule line number), and attempt to register for each separately. The process is error-prone and tedious.

Our proposed software would allow a student to enter just the course numbers (i.e. CSE 333, CSE 341, and HCDE 231), then see several (perhaps all) possible schedule configurations that would allow them to take all courses entered. If it is not possible to take all courses entered in the specified term, the student could view the possible schedules that allow them to take as many as possible and then choose their priorities, or to adjust their entries and try again, although this conflict-resolution should be under the category of additional features. Also a possible add-on feature would be functionality such that, upon selecting a specific schedule configuration, the student could register for those classes (assuming prerequisites had been met) with a single additional click. Additional filters or checks early in the process for things such as prerequisite fulfillment would be added once basic functionality had been achieved.

Much of the architecture necessary to implement this software already exists in the UW registration system, so a solution would involve mostly accessing this existing infrastructure and applying additional logic, then feeding registration information about the selected schedule options back into the system automatically.

Probably the biggest "wild card" and potential challenge here revolves around access to and ability to integrate with the school's existing systems. We don't know what channels we'll have to go through to access the information or what its format will be or how easy working with it will be. Another looming obstacle lies in the limits of the registration system itself (e.g. system going down or being overloaded, or glitches hindering the ability of our software to be effective and helpful).

Below is a simple diagram illustrating the process by which the software would operate, and a sample display of what a potential student might see.

student input
(course numbers)

COURSE OFFERINGS

Maps from course numbers to tuples of schedule line number, time slot, and location, as well as any quiz sections.

SLNs, time slots, and locations

AVAILABILITY FILTER

Removes entries that do not have space available.

SLNs, time slots, and locations (filtered)

SCHEDULE BUILDER

Constructs all possible configurations of items.

schedule

VISUAL DISPLAY CREATOR

Sample program output:

Option 1:

	Monday	Tuesday	Wednesday	Thursday	Friday
9:30	CSE 333A EEB 045		CSE 333A EEB 045	CSE 341AB EEB 125	CSE 333A EEB 045
10:00					
10:30	CSE 341A SMI 103		CSE 341A SMI 103		CSE 341A SMI 103
11:00					
11:30				CSE 333AC MGH 241	
12:00					
12:30	HCDE 231C MGH 238	HCDE 231C MGH 238	HCDE 231C MGH 238	HCDE 231C MGH 238	HCDE 231C MGH 238
1:00					
1:30					
2:00					
2:30					

Option 2:

	Monday	Tuesday	Wednesday	Thursday	Friday
9:30	CSE 333A EEB 045		CSE 333A EEB 045	CSE 341AB EEB 125	CSE 333A EEB 045
10:00					
10:30	CSE 341A SMI 103		CSE 341A SMI 103		CSE 341A SMI 103
11:00					
11:30	HCDE 231B MGH 238	HCDE 231B MGH 238	HCDE 231B MGH 238	HCDE 231B MGH 238	HCDE 231B MGH 238
12:00					
12:30				CSE 333AA SIG 226	
1:00					
1:30					
2:00					
2:30					