

Webservices to Support Departmental Dashboards

CSC 2310 – Spring 2022

Abstract

Deans, Associate Deans, Department chairs and others across the university are constantly in need of the data necessary to help drive decision making. In order to use that data, effective visualization is needed to facilitate interpretation of the vast amount of information available to these administrators. In this project we will create a number of webservices suitable for supporting the creation of dashboards that can be used to access valuable institutional data.

Data is critical for making both short-term and long-term decisions. Data scientists are often called upon to create tools for providing insight into the knowledge embedded in data by fusing information from various sources into a single cogent collection of information. For instance, the following table shows basic information related to courses offered by the CS Department:

	CRN	Subject	Course	Section	Credits	Time	Where	Instructor ▲	Enrollment	Title
1	12038	CSC	1200	001	3	1500pm - 1615pm TR	BRUN 327		25/25	Principles of Computing
2	10625	CSC	1200	002	3	0800am - 0850am MWF	BRUN 228		28/40	Principles of Computing
3	13519	CSC	1200	600	3	1500pm - 1615pm TR	BRUN 327		3/3	Principles of Computing
4	11360	CSC	1300	002	4	1600pm - 1650pm MWF	BRUN 119		43/75	Intro/Prob Solving-Comp Prog
5	11050	CSC	1300	101	0	1300pm - 1350pm M	BRUN 209		31/26	Intro/Prob Solving-Comp Prog
6	12121	CSC	1300	102	0	1300pm - 1350pm W	BRUN 209		33/32	Intro/Prob Solving-Comp Prog
7	10566	CSC	1300	104	0	0800am - 0915am T	BRUN 209		32/32	Intro/Prob Solving-Comp Prog
8	11682	CSC	1310	001	4	0800am - 0915am TR	BRUN 119		70/60	Data Structures and Algorithms
9	10830	CSC	1310	002	4	1330pm - 1445pm TR	BRUN 119		71/60	Data Structures and Algorithms
10	11467	CSC	1310	101	0	1300pm - 1350pm M	BRUN 210		24/25	Data Structures and Algorithms
11	11785	CSC	1310	102	0	1300pm - 1350pm W	BRUN 210		23/25	Data Structures and Algorithms
12	10324	CSC	1310	103	0	0930am - 1045am T	BRUN 210		26/25	Data Structures and Algorithms
13	11034	CSC	1310	104	0	0930am - 1045am R	BRUN 210		24/25	Data Structures and Algorithms
14	11834	CSC	1310	105	0	1630pm - 1745pm T	BRUN 210		23/25	Data Structures and Algorithms
15	10454	CSC	1310	106	4	1630pm - 1745pm R	BRUN 210		21/25	Data Structures and Algorithms
16	11355	CSC	2400	002	3	1430pm - 1520pm MWF	BRUN 228		27/30	Design of Algorithms
17	10238	CSC	2400	003	3	0800am - 0850am MWF	BRUN 327		18/25	Design of Algorithms
18	10911	CSC	2500	002	1	0930am - 1045am R	BRUN 410		9/16	Unix Laboratory

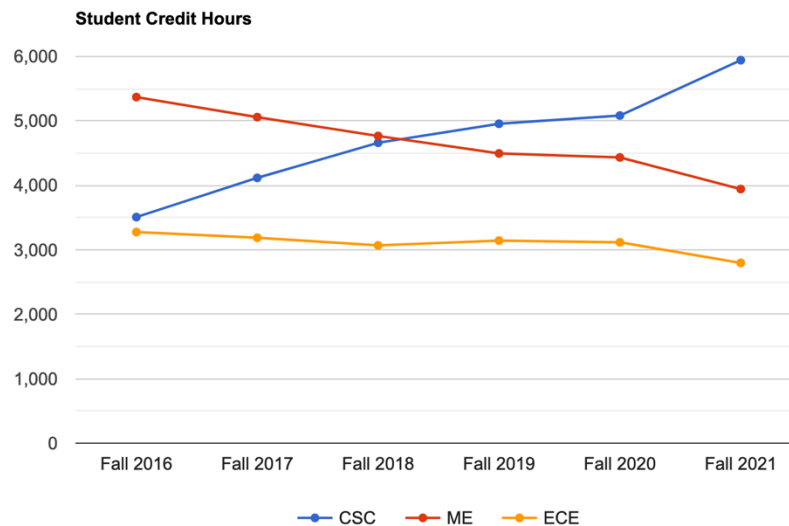
The knowledge embedded in this table includes both explicit and implicit information such as:

- (Explicit) Day, time, and room for a course
- (Explicit) Enrollment for a course
- (Explicit) List of courses taught by a specific faculty member
- (Implicit) Total student credit hours generated for the course (i.e., credits * enrollment)
- (Implicit) Total student credit hours generated by a department
- (Implicit) Total room utilization of a location of campus
- (Implicit) List of rooms available at any given moment

Information such as that found in the table, when supplemented with other data (such as the capacity of a given room on campus, list of colleges and the departments in each, etc.) can be valuable when looking to solve various problems, identifying which degree programs are growing/retracting, identifying which building spaces are being used efficiently, etc.

The purpose of this project is to identify the needs of administrators and to develop the webservices needed to support the creation of a dashboard of widgets that can be used to visualize information related to enrollment, course offerings, building utilization, etc. For instance, by using course

enrollment data, a graph showing historical trends of student credit hour generation can be displayed. Such a graph can be varied by displaying information for a specific college (such as the College of Engineering), by selecting specific departments, selecting different date ranges, and so forth.



In this iteration, you will be doing the following:

- Identify all of the potential user types for this system
- Brainstorm the potential features of the system – for instance, some users might want to add information, others may want to view specific course information (by clicking on a row?)
- Specify user stories for each of the user types based on your problem analysis using the user story format of “As a <user>, I want <feature>, so that <reason>”.

As you begin the brainstorming process, recall that as a “developer” you have biases, and that you need to try adopt the mindset of different kinds of users. You may want to ask yourself questions such as:

- What other information should we display? What does each type of user need to see and do?
- Should a user be able to add or modify data? Which type of users can do this?
- What should a user be able to learn from the display of the data?

As you go through this activity, you should attempt, as much as possible, to avoid thinking about how any of this is going to be implemented. The most important aspect of this activity is identifying all of the *possible* features for this system (noting that identification of all of the possible features does not mean you will have to implement all of the features).

Project Phases

The project will be conducted in two major phases: *concept-initiate* and *iteration-construction*. In the *concept-initiate* phase you will identify project requirements, create models, and prep your development environment to support your implementation activities. In the *iteration-construction* phase of the project you will iteratively develop software for the project by adding support

according to the requirements developed during the *concept-initiate* phase. The schedule of the phases of the project and the individual iterations contained therein are defined below:

Date (Midnight)	Phase	Description
February 4	Concept Initiate 0	User stories
February 18	Concept Initiate 1	Use case diagrams
March 4	Concept Initiate 2	Initial Design
March 25	Iteration 0	Project environment setup and initial execution demonstration
April 8	Iteration 1	Tier 1 services
April 22	Iteration 2	Tier 2 services (collections)
April 29	Iteration 3	System Integration

Concept Initiate 0

User Stories

In the initial phase of the project, you will create user stories based on the description provided while also generating ideas of other potential features for such a system. The user stories that you create should be specified in the following form:

As a <user type>, I want <desired feature>, so that <outcome>

You must also include, with each user story, a description of how you would expect to be able to observe this behavior in the system. Your source of information for the user stories you will identify are given in the description provided above as well as any discussion sessions we conduct in class. You may also reach out to faculty or other students to generate ideas on potential stories. **However, the work you turn in must be your own. You may not copy other students' user stories, descriptions, or other similar work products.**

A template for your user stories is provided below.

User Story Template

Use the following user story template for *concept-initiate 0*. Replicate this as many times as is necessary.

User Story Name: <Short phrase for the user story>	
As a:	<user type>
I want:	<feature>
So that:	<outcome or reason>
Description	<description of how the user story / feature would be observed and tested in the system>

Submission

Submit your user stories as a PDF document only. *Microsoft Word, LibreOffice, or any other native word processing format will not be accepted.*

Rubric

Completeness: You will be graded on the completeness of your turn-in. In particular, you must analyze the description and attempt to identify as many features as you can that are relevant for

the project. Note, there is no upper bound on the number of user stories you can define, but there is a lower bound that *will not* be specified.

Correctness: You will be graded based on your adherence to the format of the user story template and on the accuracy of the user stories with respect to relevance to the described project, including your description of how you might expect to observe the user story / feature in the system.

Points: The assignment is worth 20 project points.