Project Updates

1. If you use database (mySql), have a class **db.php** or **database.php** where the mysql host, user, password, database name are specified. This file must include a method that creates the database where you will store your data. Your project must not use more than one database. When evaluating your work, that is the ONLY file that will be modified to set the actual values of the parameters (mysql host, user, password, database name) . ALL queries in mySql must be done through a call of a method implemented in that php file.
2. MySql allows you to export your databases and tables to a file (CSV, JSON, XML format). When submitting your project, include ALL data files (prerequisites, schedule, list of courses per program, data from databases …) that are required to use your application.
3. Have one file **install.php** which sets up the environment (create database, tables, read files to fill in the tables …). If you had exported files from mySql, ave your install.php read the file to import the data in the tables. This file will be executed first before your application is evaluated.
4. Have one file **index.php** or **index.html** in your project.
5. Have a readme.txt or readme.docx providing the information requested on the next page.
6. Submitting a piece of work (for instance js, php, css, jar files) that you and your project partner(s) did not write will result in a grade F for the project. You cannot include any library or framework (such as JQuery, Dojo …) in your project. The evaluation will use xampp. No additional library/package will be installed.
If two groups submit codes that are similar, both groups get a grade F.
7. Pick one Engineering program (for instance Communications Eng.) to test your implementation. The program you pick may not have a basic science elective component or a breadth elective component, BUT YOU MUST describe the changes that must be made when deploying your project (See last page).
Regarding the prerequisites : focus on SYSC, ELEC and ECOR and
You may assume that students meet the prerequisites of ALL complementary studies electives and all basic science electives.

Include the following information/answers in your readme.txt or readme.docx

1. The students in your group :

	Student number	Firstname Lastname	Email (@cmail.carleton.ca)
1			
2			
3			

2. Which program (CE, CSE, SE) did you use to test your application ?

3. Who is the TA who is assigned to your group ? (Mr. Kazi or Mr. Abaza) ?

4. Provide the contributions of each member of the group.

5. Provide a brief description of your folders/files. If your project includes several folders, give the content of each folder. If you had separate scripts to implement each task in the project, then provide the mapping tasks to html/php files.

6. Provide the instructions to execute when deploying your application to offer the service to all programs of the Faculty of Engineering. What is the format of data to provide to your application (for each academic program, for the prerequisite trees, for the complementary studies electives, for the basic science electives, for the (breadth) engineering electives ? What files in your project will need to be modified to include all Engineering programs ? Briefly, describe the changes that must be done in the code ?

7. Explain how you implement the prerequisite dealing with 3rd year status or 4th yea status in Engineering and how does your code checks whether a student has or not the proper status. Explain how your software processes the perquisites that can be taken concurrently with the course (eg. ECOR 4995 must taken at the same time as the 4th year project; STAT 2605 and STAT 3502 are prerequisite to SYSC 4602, but the STAT can be taken concurrently to SYSC 4602). Explain how your application deals with program transfer : For instance, SYSC 2006 requires either SYSC 1005 or ECOR 1606. Software Eng. requires SYSC 1005, while Communications Eng. requires ECOR 1606. A student starting in Software Eng. may switch to Communications Eng, or vice versa. How does your software determine if the student can take or not SYSC 2006 ?

8. Explain briefly the rules (or algorithms) you use when building the (conflict-free) timetable. For instance, a course has currently 2 lectures sessions (A and B) and 2 lab sessions each (A1, A2, B1, B2); if all are still available, how does your software select which lecture session and lab to suggest to the student ?

9. In your solution, which entity identifies the courses that can be taken based on the completed courses and the prerequisites ? The client process or the server process ? Same question for the determination of the conflict-free timetable ? Explain your choices.