Walking Traffic Counter

The Rec Center opened a new South Entrance not to long ago. The project costed millions with all of the research, simulation and construction that went into the design. But anecdotally, the usage has not been as had hoped. The University doesn't want to spend more money to determine if indeed the new entrance is being under-utilized, so, they have asked for a cross university collaboration on developing instruments that can count foot traffic that can be used anywhere inside on campus.

With the arrival of Prof. Lupoli, (and his big mouth), it was determined that this project could be managed with the material you will learn inside and outside of this class. The overall design will use an Arduino interface to measure the traffic, and send data to a web accessible page and database for further research. Your group is not the only people that will work on this. Your work will be passed to another group. It is essential that you remember this, and build your work knowing someone will be finishing it. Since it will be passed on, there will be a coding standards that must be adhered to.

The project will consist of five parts, described below.

Arduino "box"

The Arduino board and interface will be the device that is placed in the traffic area to count the walkers. The Arduino will use a wireless "shield" (an item connect to the board) to connect to a web accessible database to add when a walker has passed the "box". Your team's Arduino must be placed inside a box:

1. Cardboard box(s) is fine
2. Completed closed, but cutouts in the box for sensors is authorized
3. The outer cover of each box must have:
    a. TAMU sticker(s)
    b. Your groups' contact information of email and cell phone
    c. Instructor's name
    d. Class name
    e. A notice describing the box's purpose
        i. Just in case, campus police come by…

Python/Arduino

The code used to program the Arduino must be entirely Python. Code that is found outside your own creation must be documented within your own Python code and in the deliverables.

PHP/HTML

A bare bones but complete web interface should interact with the same database the Arduino is dumping data into. Please notice the web interface:

1. does not need to be "pretty", but should be very complete with basic commands and data retrieval.
2. does not connect to the Arduino in any fashion
3. overall "look" of your web interface will be loosely graded, what features it supports and implements correctly will be looked at with great scrutiny.
4. can use several HTML/PHP pages for ease of use

While you are all given server space by CSEE, there could be the issue of sharing sensitive passwords. In that case, an outside server can be created and used for the web portion. But the DB must use the one given by CSEE, which can be a unique but shared password among your team. The code to be used for this portion is PHP, HTML and CSS (if needed).

PHP/MySQL

PHP pages will be used to interact with the MySQL database and show the user data from that DB. Again, for privacy purpose, creating a PHP server will be allowed, but remember:

1. Make sure to define where your web interface (URL) can be found within your documentation, and very easy to determine(find)
2. Make sure all on team have access to edit the code on that server
3. All DB queries from your PHP pages will connect to the TAMU CSEE DB given.

Deliverables

There will be many portions of this project that are graded but will not contain code. Here is a list a description of each. Please note, some of the deliverables will be due either midway, and the end of the project, or both (initial and then final)

Should have them by name, then format (pdf, word, etc…) then description

In the Midway project deliverables, notice that not much of your code is required other than preliminary test cases.

Midway Project 1 Deliverables	
Deliverable	Format
Description	
Initial UML Structure Design	.dia (Dia)
This needs to cover who all aspects will interact with each other.	
Initial Use Case Diagram	.dia (Dia)
Description here	
TimeLine	.xls (Excel)
This a deliverable that shows what your team has planned for a timeline to complete the project. Please be as specific as possible on which aspects you are completing, even though it	

might change as time goes on	
Initial Test Cases (DB interaction with PHP)	.txt
This should simple be a copy and paste of code used to test for validation. Ity will be graded on overall completeness.	
Initial Test Cases (DB interaction with Python)	.txt
This should simple be a copy and paste of code used to test for validation. Ity will be graded on overall completeness.	
Team Checkup	(google form)
A Google form will be distributed by your instructor. Simply fill out the form.	

Remember, your code, documentation, etc… will be given to another class. Your names should not be ANYWHERE on them to avoid… awkward situations. The list below should be everything you develop for this project.

Final Project 1 Deliverables	
Deliverable	Format
Description	
Arduino wiring diagram	.fzz (Fritzing)
Using Fritzing, create an Arduino wiring diagram of your project so the interface can be mimicked by another group after your group's time on this project has passed.	
Final UML Structure Design	.dia (Dia)
This needs to cover who all aspects will interact with each other. Make sure it accurate reflects your current version of your project.	
Presentation	.pptx (In-Person, as a group)
A Doodle poll will be given within 1 week of your project 1 due date to schedule your group presentation Overall description here	
Final Use Case Diagram	.dia (Dia)
Description here. Make sure it accurate reflects your current version of your project.	
Final Test Cases (DB interaction with PHP)	.txt
This should simple be a copy and paste of code used to test for validation. It will be graded on overall completeness. Make sure it accurate reflects your current version of your project.	
Final Test Cases (DB interaction with Python)	.txt
same as above	
Final Test Cases (Arduino interaction with Python)	.txt
same as above	

Teammate Evaluation	(CATME form)
CATME will email a survey for you to complete that asks you to describe working with your teammates.	
Arduino/Python Code	.py (Python)
All code, zipped together	
Webserver Code	.html/.php/.css (various web)
All code, zipped together	
Arduino/Python related API	.docx (Word)
While the contents and setup are up to your team, the reason for an API (will be) is clear. You are making this for the next team to take this project. Remember to have your test cases or functions with as well. It should be easy to read, follow and organized. The API should also contain where any outside code was lifted and when.	
PHP/MySQL related API	.docx (Word)
same as above	
Arduino/Python related API	.docx (Word)
same as above	
MySQL Table Setup	**.txt**
Within PHPMYAdmin, there is a way to export the code needed to create the same tables in your project. The next group will need this.	

Project 1 Grading

The project will be graded by the various aspects below. We reserve the right to add and change point values as the project progresses and revelation come to light.

Topic	*Percentage*
Adherence to Coding and Documentation standards	10%
Arduino Wiring diagram	2.5%
API – DB/PHP(HTML) Interface	5%
API – Arduino/Python Interface	5%
API – DB/Python Interface	5%
Project 1 Midway Deliverables	15%
Final UML (Structure)	10%
Final UML (Use-Case)	5%
DB overall design	5%
Test Cases – DB/PHP(HTML) Interface	2.5%
Test Cases – Arduino/Python Interface	2.5%
Test Cases – DB/Python Interface	2.5%
Teammate evaluation (CATME)	10%
Presentation	10%
Project completeness	10%

Turning in Work

Except as specified otherwise (e.g. emails), you are to submit work and reports via ECampus system. Individual reports (e.g. the final report regarding teamwork) will be submitted by individuals using CATME. The team reports and code should be submitted by one team member.