CS 3360, Fall 2019
Professor Ward

PHP Assignment

Society today faces many great challenges. Technology can often help.

> In Albertsons, my daughter saw rambutans and decided to buy six to take to a party. She used the self-check-out, but couldn't find rambutans listed. A cashier saw her having trouble and came over to help. After a short conversation, the cashier punched in the PLU for something else entirely, and told my daughter that was the real name of the fruit, even though the picture was entirely different and the price was much lower. My daughter paid as she was told and left, but felt bad about it.

Maybe technology can help prevent future disasters of this type. In this assignment you will design and prototype a system for this purpose.

It will provide an interface with a list and two input fields, for "name" and for "PLU," and a submit button. After every shipment arrives the produce manager will look to see what new fruits and vegetables came in, and if necessary is the input fields to add them, one by one, to the list. (He will presumably do this for every fruit or vegetable that is sold loose and has a surface too rough for the PLU stickers to properly adhere to. For example, depending on the store and season, these may include rambutans, lotus root, rhubarb, quinces, chestnuts, jackfruit, aloe vera leaves, prickly pear cactus, and daikon.) The system will display an alphabetized list of all items entered and their PLUs, for cashiers to refer to via the web (instead of looking at the same information handwritten on a scrap of folded paper in their apron pocket).

Grading will be based on the points earned:

- 10 for the basic functionality described above.
- 5 if a user can move commonly-encountered items to the top of the list, by clicking checkboxs for the items and then a “move checked items to top of list” button.
- 5 points if users can add aliases to items, for example “nopales” as an alternative name for 4558. Aliases will be shown with a light green background. Aliases can only be added for items already on the list.
- 5 points if there is some way for users to add pictures of the items
- 5 points if the list shows only items that may currently be in the store. This is determined by looking at past invoices, and including only items which came in no more than two weeks before today. Sample invoices and a mapping from SKU to PLU are at the class homepage.
- 2 points if there is password control, so that only the managers, not cashiers, can add items.
- 0 points for usability (since that’s not the point of this class)
- up to 5 for other interesting features. Features involving AJAX or a database are discouraged.

To earn the full number of points for each feature, it must be clearly implemented and documented. You can aim to earn points proportional to the grade you want: 10+ C, 15+ B, and 20+ A. The maximum number of points we will record in the gradebook is 25.

Systems with interesting features or with exemplary code structure and style may earn you an invitation to present your code to the class, for extra credit.

Your report should be hardcopy, and include:

- The URL of your prototype. We plan to give you a place to host your system on a CS department server; details will be given later.
- Description of the features of your prototype
- Explanation of how to use them, unless it's obvious
- Screenshots showing that all the features work
- Explanation of the structure of your code, using UML or any diagramming style you like that would be helpful for people wanting to understand or extend your code, up to one page.
- Your code (literately written and appropriately commented)
- A example of how one of the unusual properties of variables in PHP caused problems or affected your code.

You may work with anyone on the interface design and implementation (html and css) or use any html and css you find, with appropriate acknowledgement of sources. However all the PHP code must be your own, done individually.

Helpful links: ifpsglobal.com, supermarketpage.com, github.com/ankane/plu

Due September 25.