CS 474: Object-oriented Programming languages and environments

Smalltalk Project #2

Due time: 3/20/2014 at 9:00 pm
Instructor: Ugo Buy
TA: Venkat Srinivasan

You have been asked to code an *electronic restaurant order system* (EROS) for use by the infamous *Expensive foods "R" us* restaurant chain. EROS allows the restaurant chain to set up and maintain the lists of available food and beverage selections. Food selections are classified as follows: (1) appetizers, (2) main dishes, and (3) desserts. Appetizers are defined by a name and a list of ingredients. Main dishes are characterized by a name, ingredient list, and a side dish. Finally, desserts are characterized by a name and a short description of the dessert. Also, your selection display should show, for each food item, an approximate calory count and the price of the food item.

Beverage selections fall under the categories of (1) soda selections and (2) long drinks. Each soda selection has a name, a price, and a calory count. Each long drink has a name, an ingredient list, a calory count and a price.

All interactions with the user are through a graphical user interface. You must use Cincom Smalltalk to implement the whole system. You are free to choose the general layout of the interface and the operations provided by your system. You can assume that a restaurant owner will be working with just one list at a time. (This means that your program should not support multiple food and beverage lists). You should use inheritance in your code, however. The following is a plausible list of operations that your system must support:

- *Create_List.* This operation allows a user (i.e., the restaurant owner) to create a new list of foods and beverages. An empty list is displayed on a the user's screen.

- *Add_List_Entry.* This operation allows a user to add a new food or drink selection to the list. Food and drink selections must conform to the categories described earlier. The fields of the new item are entered interactively in the Smalltalk system. Upon completion of this interaction, the updated list is displayed on the user's screen.

- *Remove_List_Entry.* This operation allows a user to remove a food or drink selection from the list. Food and drink selections must conform to the categories described earlier. The fields of the new item are entered interactively in the Smalltalk system. Upon completion of this interaction, the updated list is displayed on the user's screen.

- *Save_List.* The current list of food and beverages is saved to a named file. (Hint: use the BOS capability of Cincom Smalltalk—described in the web page for the class—for this function).

- *Retrieve_List.* This operation prompts an interactive user for a file name. A previously-saved list of foods and beverages is retrieved from the file and becomes the current list. The new list is displayed. (Hint: Again use BOS.)

- *Suggest_Full_Meal.* This feature allows the restaurant owner to enter a total calory count and price. Given these two items EROS searches the food and beverages available in the system in order to put together a meal whose total calories and price do not exceed the total entered by the user. The meal must consist of the following items: (1) a main dish, (2) an appetizer or a dessert, (3) a soda drink, and (4) a long drink. If a meal cannot be created within the calory and price totals specified by the user, an appropriate message is displayed. Otherwise, the various items in the meal are displayed.

- *Suggest_Light_Meal.* This feature allows the restaurant owner to enter a total calory count and price. Given these two items EROS searches the food and beverages available in the system in order to put together a meal whose total calories and price do not exceed the total entered by the user. The meal must consist of the following items: (1) an appetizer, (2) a dessert, and (3) a soda drink. If a meal cannot be created within the calory and price totals specified by the user, an appropriate message is displayed. Otherwise, the various items in the meal are displayed.

- *Quit.* EROS is exited.

You must work alone on this project. Submit this project using the digital dropbox feature of the courseinfo web page. No late submissions will be accepted.