**Design Justification Group Assignment 4**

With functionality, the new updates administered to the vending machines will include an ability to pay through debit card, a graphical user interface, and a configuration panel for technicians. It will also include a locking and unlocking mechanism, which are designed to enhance safety protocols for technicians.

Our updates are facilitated through a number of classes that communicate amongst one another, thereby communicating with the respective logic class. The communication is carried between parts of the software, which are then administered, and will not rely heavily on the hardware. The lessened communicational burden between software and hardware allows for changeability of the program, while providing comprehensibility. In taking the advice that was provided by Mr. Client on keeping the software at optimal level of changeability without the burden of the aforementioned hardware, our design has been formulated according to efficiency. Hence, throughout the organization of the source code, each of the unique classes is observed to be communicating with one another.

This organization of the source is formulated in manner that would be easily understood by an individual with a Computer Science background. Organized in respective groups of major tasks, the source code has classes that all fall under the graphic interface, the configuration panel, and so forth. Although these major tasks are frequent to communicate with one another, its main task will remain acting as a closed system. Thus, making this process independent from the hardware, or the rest of the code for that matter. This allows for accessibility of new implementations in further updates. This manner of programing through closed systems would prove convenient, as it would not interfere with any recent changes to the software. When communicating with one another, via vending machine’s software and hardware, the two is akin to the likes of a TV communicating with a DVD. If in a case that an aspect of the DVD is flawed, the TV would not experience negative feedbacks from major bugs, but instead maintains its integrity by remaining as independent entities - despite the awareness of the other system’s presence.

In addressing changeability, the most realistic feature allowing this code to evolve would be the ways in which the user’s credit is incremented. Currently we are keeping track of the user credit in one variable, where the credit is incremented when a valid coin is inserted, or when a user pays with debit. With this method, where one variable keeps track of the credit, we can add features that allows a user to partially pay coins, and pay the remainder with any other form of payment, including debit. This method keeps the flexibility and evolve-ability of the code very high, thus making it easy to implement users paying with various forms of payment, including Bitcoin. This will allow the system to efficiently support any new currency that may be implemented at a later date; yet a majority of the original code will remain as it is. Once again, the idea of a closed systems applies here as the calculation of the user’s credit is done through a listener class. This as a result will not affect whether or not the user is inputting a coin, or paying with debit. In so forth there are sufficient funds, the pop-can will be delivered to the user.

If a feature allows accessibility to other forms of payments, the code would remain nearly stagnant, or within its intended originality. This will forward ease in building on top of the pre-existing software. The structure present in the software is designed in respect to support alternative hardware for further future uses. This can be observed through the style utilized to code the methods, such as: event logs, valid coin inserts, buttons pressed, and so forth. In methods that are used to determine what buttons are pressed, or how much a pop-can costs, it is simplified using an array list that benefits hardware changes. This array list has proven to be effective when, for example: the number of buttons transforms within the hardware, or when the cost of the pop-can is being updated. This will ensure that any buttons added to the vending machine hardware will fully be represented in the software. Finally, the update of the cost for a pop-can will be administered through the user-friendly technician panel.

The efficiency of the machine, in respective of a company’s perspective, are connected to its continuing ability to evolve, and future changes in levels of understandability. With a simplified software, the effort and time required in updating a software will be lessened, thereby benefiting the company in ways that saves efficiency and monetary income. With this idea of efficiency, it extends to the technician panel where the technician will be required to perform certain tasks (i.e., change the amount of money required for a pop-can). The performance of certain updates will be executed quickly due to the machine’s efficiency within the physical and environmental realm.

Wherefore approaching scalability within vending machines, the software that is being implemented is unquestionably scalable in relation to its saleability. By having the ability to support a new hardware through basic adjustments, or through its extensibility to function with multiple forms of payment, the machine’s scaling mechanism has a potential in functioning as a singular software that could be used with all basic structures.

As an entire entity, the structure of the vending machine’s software is comprised of interconnected closed systems. The groups of closed systems not only allow for high levels of changeability, it also has been proven through the examples we discussed above. In confirming this structure’s validity through levels of immense changeability, the simplicity in understanding the software is also observed. With comments to support one’s understanding, the code itself does not require a convoluted amount of tracing throughout the system. Instead, one will only be required to look at each closed system, and understand the modes of communication occurring throughout the machine. Conclusively, this design in itself will enable the machine to be easily updated, and bring an advance towards future innovate ideas for vending machines.