**Assignment 4**

**Due, Monday, July 14, 2014 for 100% credit**

**Tuesday, July 15, 2014 for 90% credit**

**Wednesday, July 16, 2014 for 80% credit**

**Thursday, July 17, 2014 for 70% credit**

**Deliverables**

To complete this assignment you must submit your compressed Netbeans project to Webcourses.

**Introduction**

This assignment is to develop a User Interface using classes from the javax.swing package.

**Tasks and Rubric**

|  |  |  |
| --- | --- | --- |
| Activity | | Points |
| Products.java | Instantiate a minimum of 16 instances of class JLabel with text in each to identify the product position in the products grid and include the item cost (Example: A0, B2, C3 ~ $1.00) |  |
|  | Add the JLabel instances to the JPanel using GridBagLayout and GridBagConstraints |  |
| OR | Update each JLabel that an ImageIcon was instantiated with by   1. Setting the text to the product location and the item price 2. Setting the horizontal text position to JLabel.CENTER 3. Setting the vertical text position to JLabel.BOTTOM |  |
| ProductBin.java | Create an inner class that implements interface ActionListener and override function actionPerformed(ActionEvent e) |  |
|  | Register the ActionListener to the JButton |  |
| Keypad.java | Create an inner class that implements interface ActionListener and override function actionPerformed(ActionEvent e) single class for all of the JButtons with a letter, number, or dot |  |
|  | Register the ActionListener to each of the JButtons with a letter, number, or dot |  |
|  | In the actionPerformed function determine which button was clicked using a conditional statement and evaluate e.getActionCommand(), based on the result output to the user which button was clicked on using System.out.println(); |  |
|  | Create an inner class that implements interface ActionListener and override function actionPerformed(ActionEvent e) for the Enter Money JButton |  |
|  | In the actionPerformed function determine which button was clicked using a conditional statement (you can use and **if/else if/else** OR **switch**) and evaluate e.getActionCommand(), based on the result output to the user which button was clicked on using System.out.println(); |  |
|  | Register the ActionListener to the JButton with Enter Money |  |
|  | Create a JButton for the character period/dot (.) |  |
|  | Add the JButton instance to the JPanel using GridBagLayout and GridBagConstraints |  |
| Compiles | Source compiles with no errors | 10 |
| Runs | Source runs with no errors | 10 |
| Comments | Source includes comments | 5 |
| Total |  | **100** |

**Example output**

**Figures**



Figure 1 Vending Machine



Figure 2 Vending Machine



Figure 3 Keypad



Figure 4 Keypad

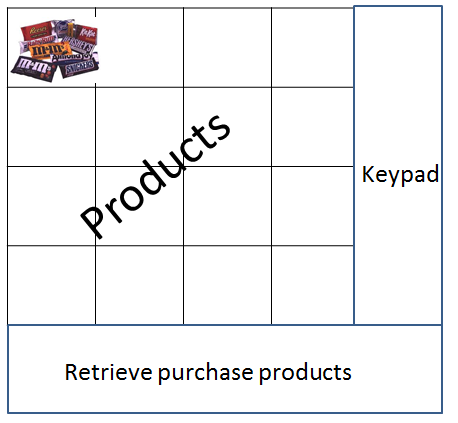


Figure 5 User Interface Layout

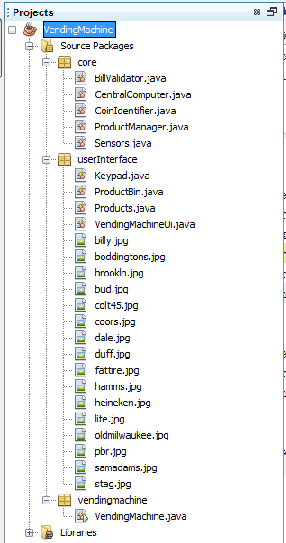


Figure 6 Project View

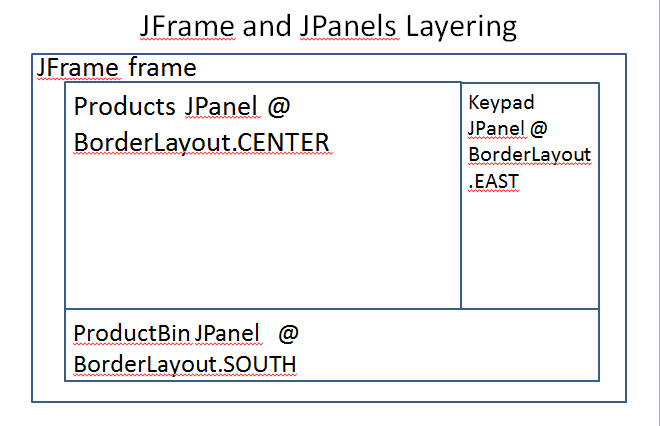


Figure 7 JFrame and JPanel

# Step-by-Step Process of How a Vending Machine Works

### Definition

A vending machine is an, "electronic machine used to disperse a product to a consumer after a certain amount of money has been put into the machine," as defined by Business Dictionary. A vending machine can give small business owners real extra income without cost. Most vending companies do not charge for placing machines in a business, and they split the proceeds with the small business owner.

### Step 1

Load the vending machine. Prior to operation, a vending machine must be stocked with the products it will dispense to consumers. These products are often supplied by the vending machine supplier, who buys them wholesale and then sells the product at a retail price. Vending machines typically have specialized keys that a small business owner can use to access the inside. Fill each display with product and make sure the products do not become inter-tangled with the vending mechanism.

### Step 2

Test the vending machine. To ensure the machine is properly loaded and ready to dispense, select one or several products and use both coin and paper money to pay for the items. Vending machines measure coins and read paper money. Each machine differs slightly, but in general, it identifies each coin with an electromagnetic field which measures thickness and diameter. Dollar bills are scanned with a laser, which transmits to a small computer that deciphers each denomination. It then calculates a balance and allows the consumer to make a choice of product. When the consumer chooses, electronic signals are relayed to a motor through an electronic console board which turns the spiral dispenser of the selected product. The machine then deducts from the balance the amount of the purchase.

### Step 3

Collect the money. Vending machine providers generally do not allow a small business owner access to the money collection vault and reserve access to approved vending machine mechanics. These individuals visit on a weekly, biweekly or monthly basis. The vending machine mechanic inspects the inside and outside of the machine for damage and regular maintenance. He then opens the money collection area and reconciles the money with the transaction log. Once accounted, the mechanic splits the proceeds from the sales with the business owner.

# Vending Machine Components

### Keypad

The keypad is not only the vending machine’s main input device but also the home for its central computer. When you press a key combination on the pad, it tells the central computer what it should do next.

### Bill Validator

When you use a bill, it goes through a validator. A treadmill-like path pulls your bill into the vending machine and through the optical scanners. Each optical scanner is a small camera that takes pictures of your bill. These pictures are then sent to the main computer, which runs a program looking for specific markers of worth and authenticity.

### Coin Identifier

A device like this one determines the type of coins you insert thanks to a series of electromagnets. Different coins affect the electromagnets differently due to their metal composition and thickness. This is how the vending machine figures out if you're using a quarter, dime, nickel, penny or even a plastic coin.

### Product Manager

Many vending machines are equipped with metal spirals to hold products in place. Of course, those spirals are easy to observe if you're peering through a machine with a glass front. The metal spirals that hold products in place are controlled by a motor, which turns the coils once the main computer has determined you've put in the correct amount of money and your selection has been entered. This allows for your product to drop to the bottom where you can finally retrieve it.

### Sensors

At the bottom of the vending machine, a line of laser beams determines if your product has been released by the metal spirals. Each laser beam is paired with an electronic light sensor. When your product falls it breaks this path, telling the computer that it was a successful transaction.

While most use metal spirals, vending machines that dispense beverages like coffee and tea must instead depend on plastic containers to house their products.

### User Interfaces

Many vending machines are now equipped with touch screen interfaces. They allow for the vending machine to convey much more information, as well as different kinds of information.

Need more than a small snack to hold you over? No problem. You can also buy hot dogs and hamburgers, as a vending machine in Spain displays. Different countries offer different foods; you can buy ramen similarly in Japan.

A Japanese vending machine takes these convenient devices a step further; you can pay simply by scanning your mobile phone and directly charging your account.

Renting movies at your local grocery store or mall has become more common due to these movie rental vending machines. Unlike traditional vending machines, you must return what you purchase!

German-made vending machine Gold To Go allows you to purchase gold pieces on the run. More products are available through vending machines today. You may find Apple iPods, jewelry, cigarettes or even groceries.

**Credit Card**

Many newer vending machines also allow for purchase through credit card.