**Software Development Engineer**

**Qualification Assignment**

**Name: Nadine Eideh**

**Date: Nov. 14th, 2022**

Thank you for your interest in joining Freightos team and congratulations on being shortlisted for this employment opportunity at Freightos.

This 24-hour assignment will test your skills relating to the professional execution of the Software Development Engineer position major job responsibilities. This assignment will test your level of knowledge mainly in Computer Science Fundamentals (coding, algorithms and data structures), OO, and problem solving among other skills.

**Introduction**

Vending machines can be be of different types. Some vending machines are dedicated to coffee, some are dedicated to drinks, and others are dedicated to snacks. For this assignment, you will implement a Snack Vending machine.

**Deliverables:**

1. Provide a UML diagram/s to illustrate your understanding of Object Oriented design techniques and concepts.
2. Provide a code skeleton to the design.
3. Implement the SnackMachine class
4. Provide a test suite to the main class.

**Specifications of the Snack Machine**

The Snack Machine has the following characteristics:

* **Money Slots:** the machine accepts money of the following types:
  + CoinSlot: There are four denominations: • 10c • 20c • 50c • $1
  + CardSlot : all cards accepted
  + Notes Slot :20$ and 50$ only
  + Machine only accepts USD currency
* **Snack Slots**
  + The machine has five rows to display snack items.
  + Each row has 5 columns to pile the items.
  + Each column has a number.
* **Keypad**

Users can select the items to be purchased using a keypad.

**Purchase a Snack Use Case**

**Basic Flow**

1. This use case begins when the customer wants to purchase snacks.
2. The customer selects a number by pressing on the keypad.
3. The VM displays a message that the snack is available for the selected number and displays its price.
4. The customer inserts the money.
5. The VM validates the money.
6. The VM accepts the money.
7. The VM displays the accumulated amount of money each time a new money is entered.
8. The VM monitors the amount of the accepted money, If the money is enough, the VM dispenses the selected snack to the customer.
9. The VM determines if any change should be sent back to customer.
10. The VM displays the change at panel.
11. Then, the VM dispenses change.

**[ P.S You are required to come up with alternative scenarios to this basic flow.]**

**Alternative Flow:**

1. This use case begins when the customer wants to purchase snacks.
2. The customer selects a row and column by pressing on the keypad.
3. If the snack is available, the customer is asked to insert the payment method number from menu.
4. The customer chooses the payment method.
5. The customer inserts the money.
6. The VM validates the money.
7. The VM accepts the money.
8. The VM displays the required amount left of money each time a new money is entered.
9. The VM monitors the amount of the accepted money, If the money is enough, the VM dispenses the selected snack to the customer.
10. The VM asks the customer if they want anything else. If yes, then the menu is displayed again (step 1).
11. The VM determines if any change should be sent back to customer.
12. The VM displays the change at panel.
13. Then, the VM dispenses change.