Лабораторная работа №2.

Техники тестирования Black-box

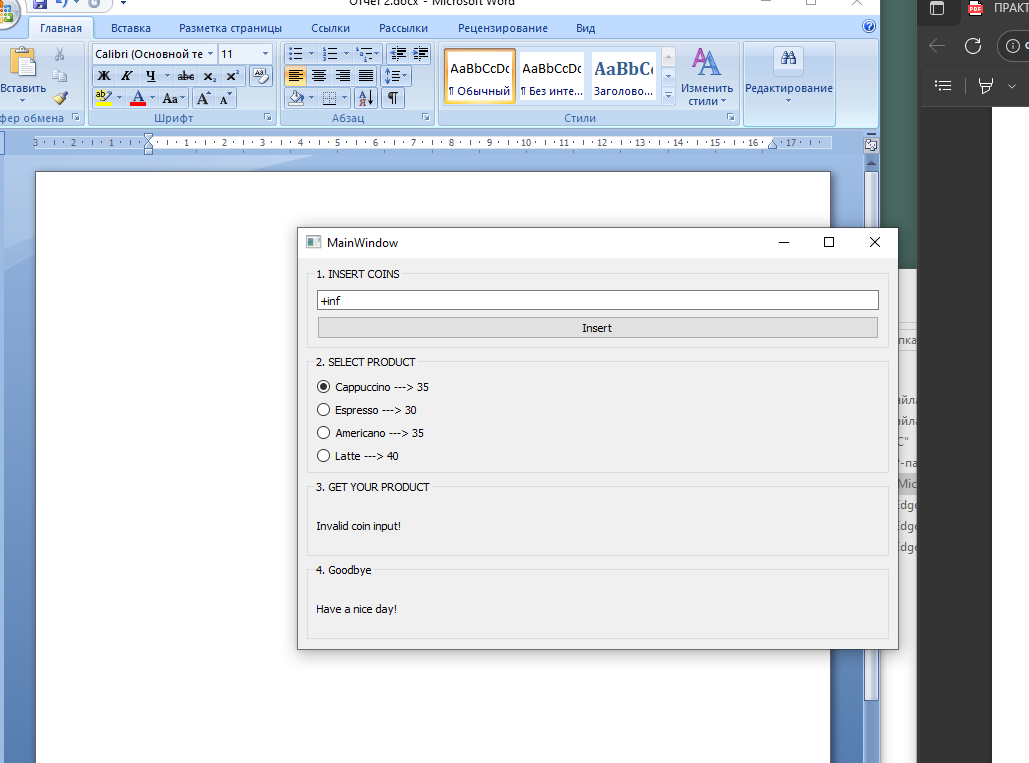


Рисунок 1 – Ввод информации +inf

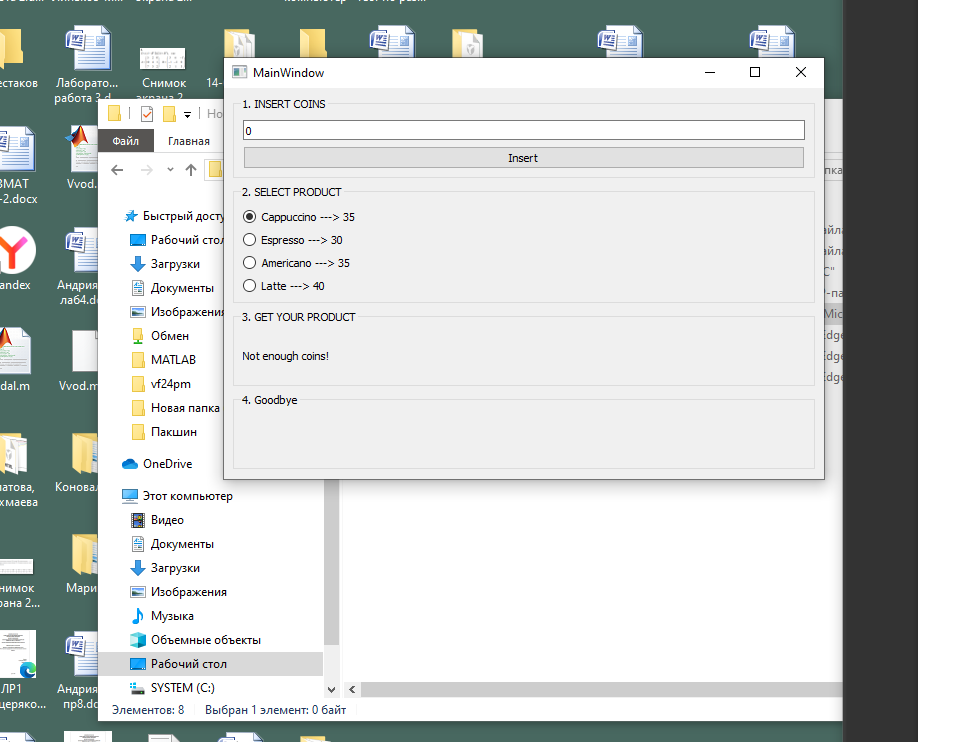


Рисунок 2 – Ввод информации сумма 0

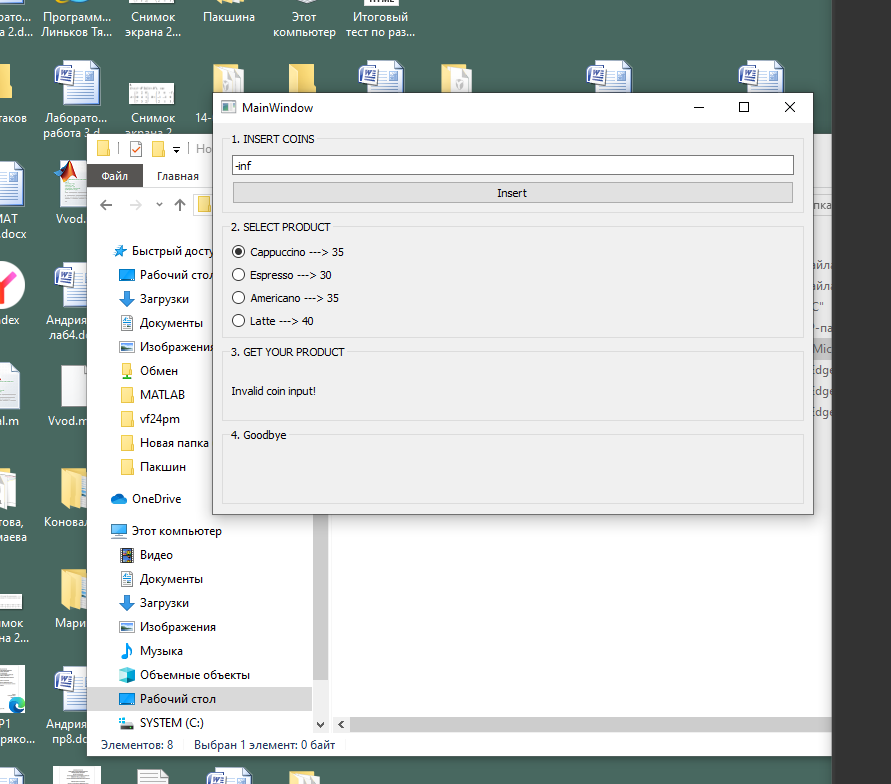


Рисунок 3 – Ввод информации –inf

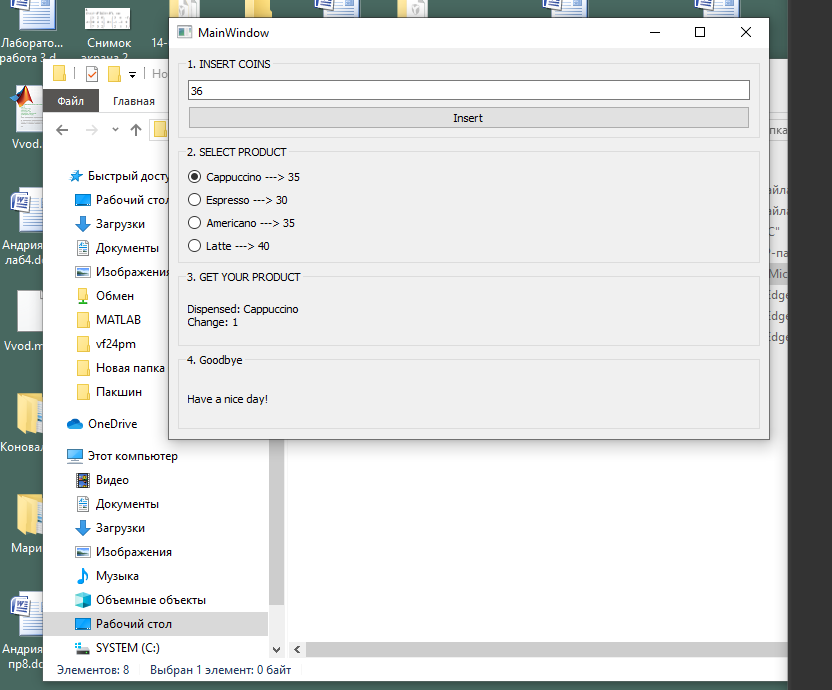


Рисунок 4 – Ввод информации сумма 36

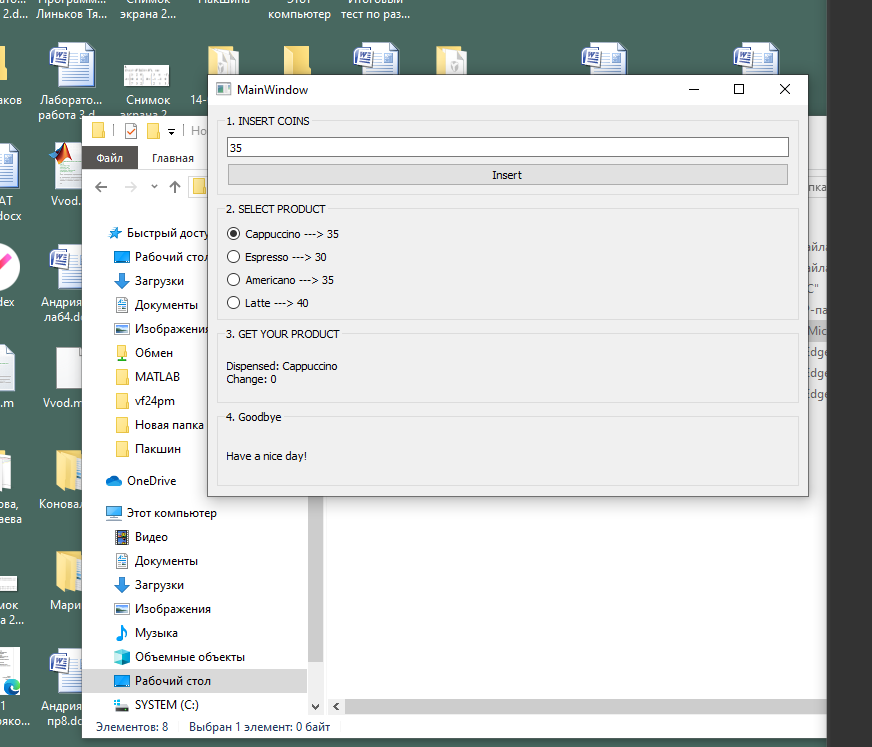


Рисунок 5 – Ввод информации сумма 35

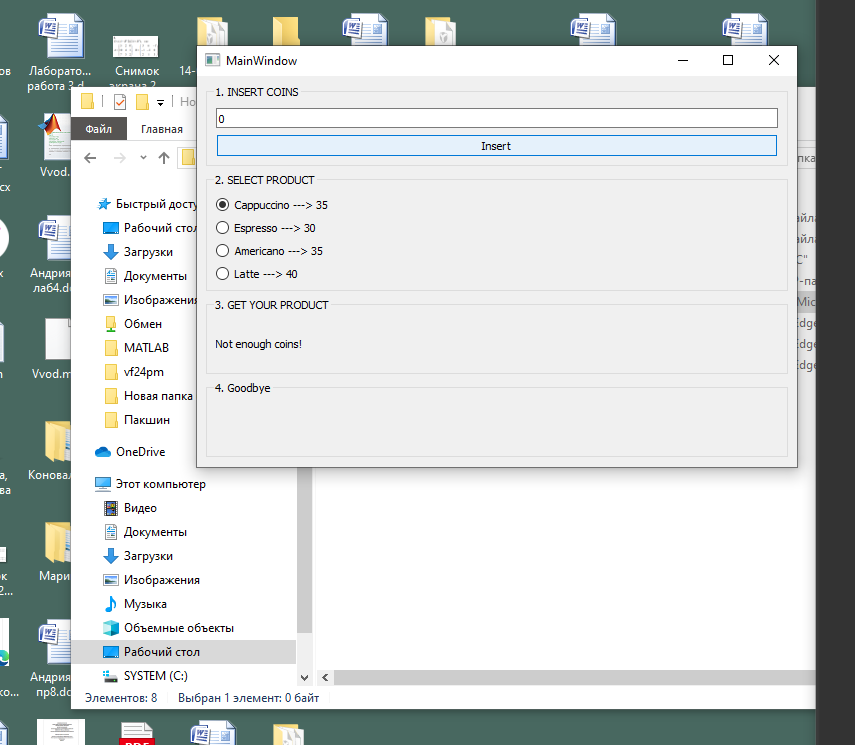


Рисунок 6 – Ввод информации сумма 0

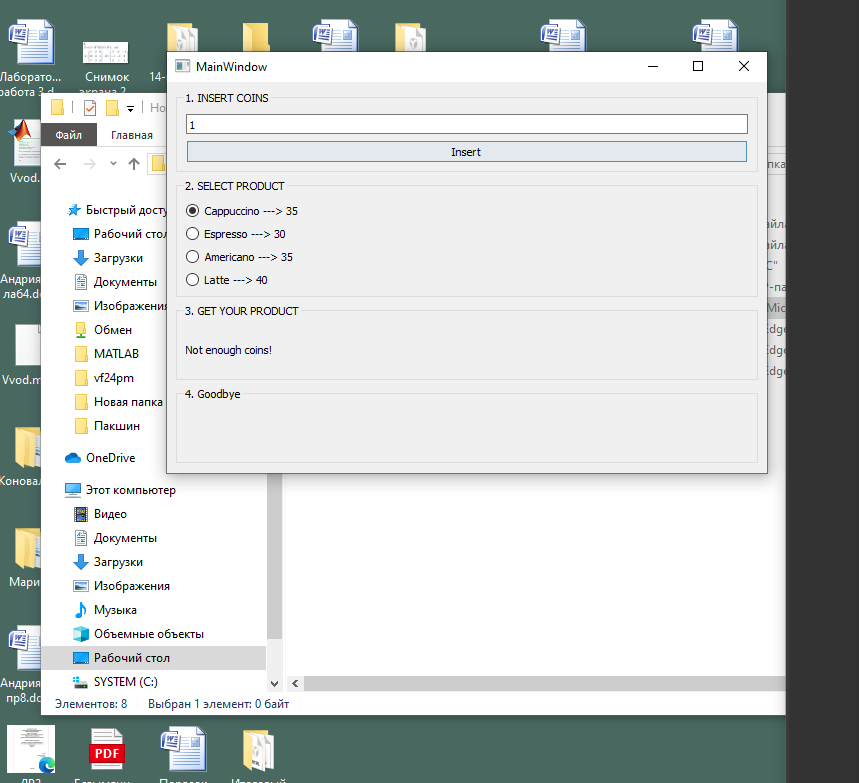


Рисунок 7 – Ввод информации сумма 1

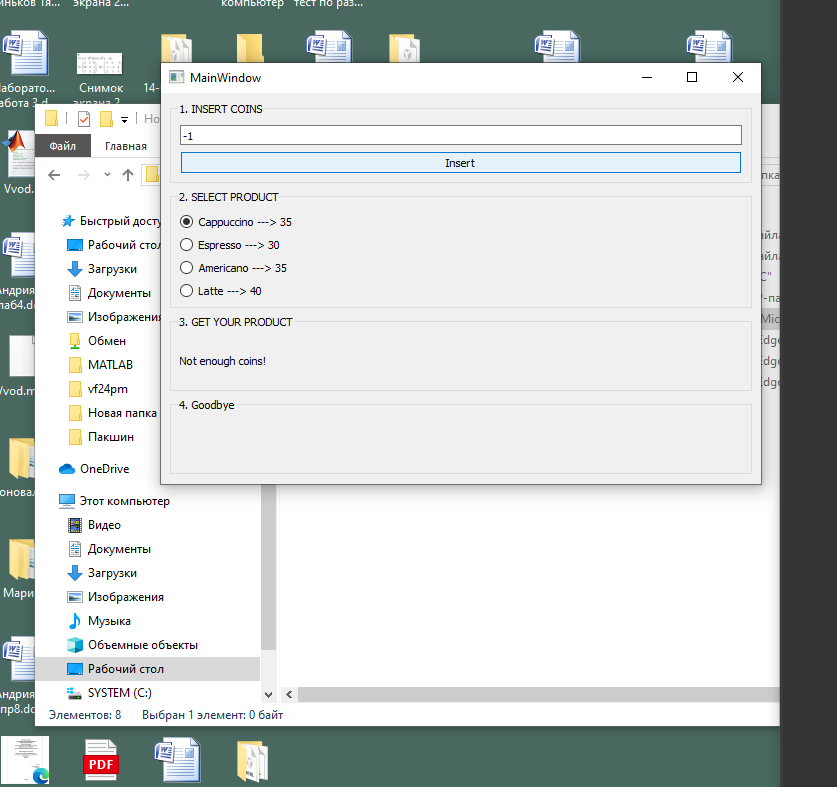


Рисунок 8 – Ввод информации сумма - 1

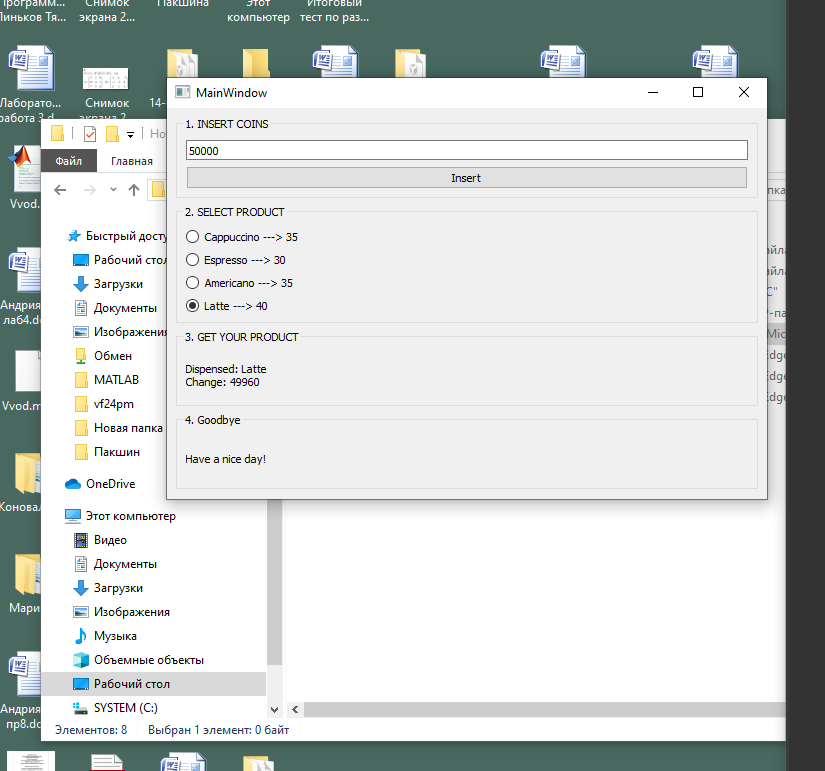


Рисунок 9 – Ввод информации сумма 50000

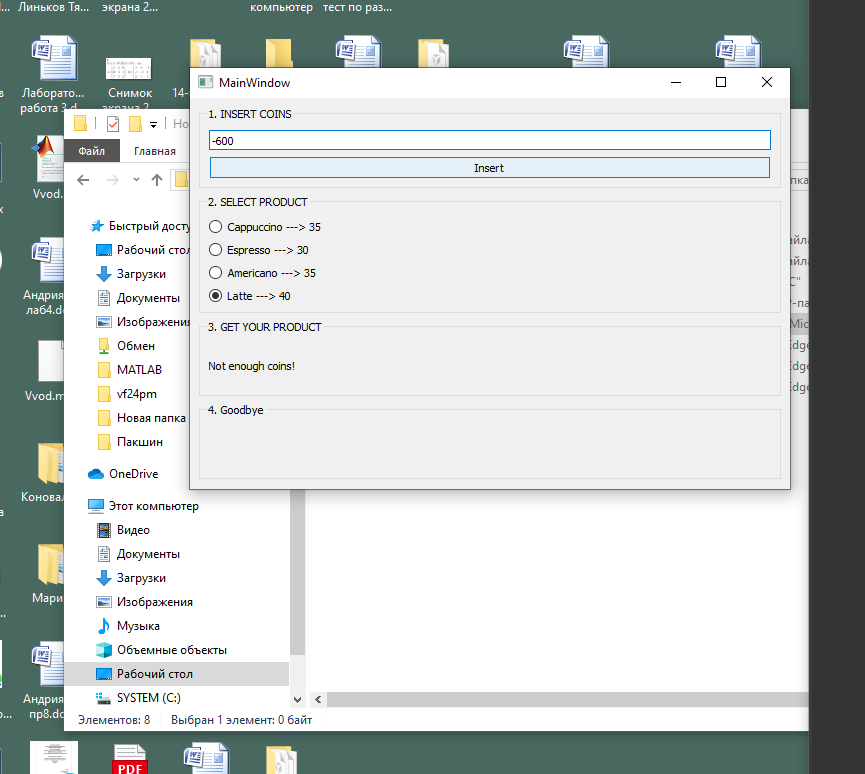
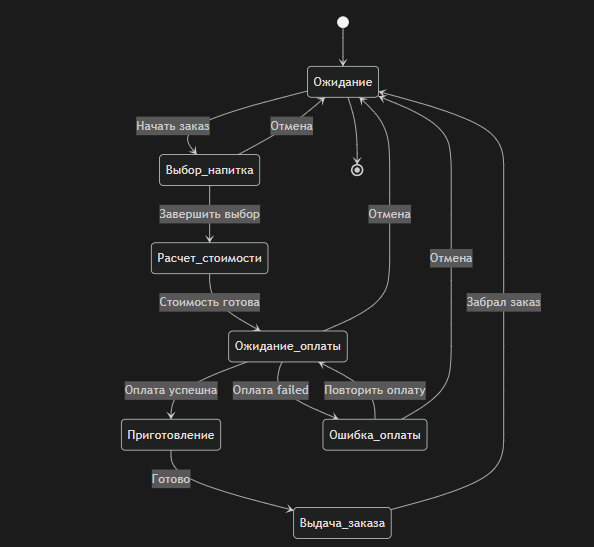


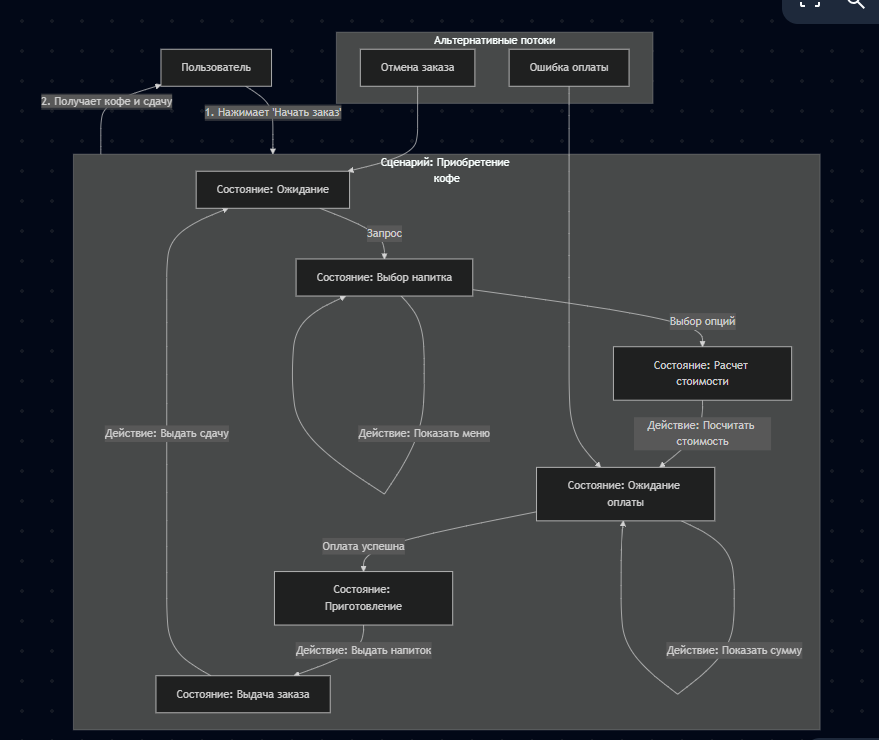
Рисунок 10 – Ввод информации сумма -600

Обнуление информации

|  |  |  |
| --- | --- | --- |
|  | Правило 1 | Правило 2 |
| Условие 1. Выбрать кофе | Y | Y |
| Условие 2. Выбрать количество напитков | Y | Y |
| Условие 3. Количество сахара | Y | N |
| Условие 4. Выбрать сироп | N | N |
| Условие 5. Выбрать молоко | N | N |
| Условие 6. Выбрать объем | Y | Y |
| Условие 7. Выбрать тип кофе | Y | Y |
| Условие 8. Узнать стоимость | Y | Y |
| Условие 9. Оплатить кофе | Y | Y |
| Условие 10. Взять сдачу | Y | N |
| Действие 1. Показать меню | X | X |
| Действие 2. Выдать нужное количество | X |  |
| Действие 3. Добавить сахар | X |  |
| Действие 4. Показать разновидности сиропа | X |  |
| Действие 5. Показать разновидность молока |  |  |
| Действие 6. Посмотреть объемы | X | X |
| Действие 7. Посчитать стоимость напитка | X | X |
| Действие 8. Принять оплату | X |  |
| Действие 9. Оплата не прошла |  | X |
| Действие 10. Выдать сдачу | X |  |

| Текущее состояние | Событие | Следующее состояние | Действие |
| --- | --- | --- | --- |
| Ожидание | Начать заказ | Выбор напитка | Показать меню |
| Выбор напитка | Отмена | Ожидание | Сбросить выбор |
| Выбор напитка | Завершить выбор | Расчет стоимости | Посчитать стоимость |
| Расчет стоимости | Стоимость готова | Ожидание оплаты | Показать сумму |
| Ожидание оплаты | Внес деньги | Ожидание оплаты | Принять оплату |
| Ожидание оплаты | Оплата успешна | Приготовление | Начать приготовление |
| Ожидание оплаты | Оплата failed | Ошибка оплаты | Показать ошибку |
| Ожидание оплаты | Отмена | Ожидание | Вернуть деньги |
| Ошибка оплаты | Повторить оплату | Ожидание оплаты | - |
| Ошибка оплаты | Отмена | Ожидание | Отменить заказ |
| Приготовление | Готово | Выдача заказа | Выдать напиток |
| Выдача заказа | Забрал заказ | Ожидание | Выдать сдачу |





Тестовый сценарий 1: Успешная покупка капучино с сахаром

| Параметр | Значение |
| --- | --- |
| Идентификатор: | TC-COFFEE-001 |
| Название: | Успешная покупка капучино с сахаром и получение сдачи |
| Предусловия: | 1. Автомат включен и находится в состоянии "Ожидание" 2. Автомат исправен, все компоненты работают 3. В автомате достаточно ингредиентов и денег для сдачи |
| Входные значения теста: | Шаг 1: Нажать кнопку "Начать заказ" Шаг 2: Выбрать "Капучино" Шаг 3: Выбрать объем "300 мл" Шаг 4: Добавить 2 порции сахара Шаг 5: Нажать "Готово" Шаг 6: Внести купюру 200 рублей |
| Ожидаемый результат: | ОР1: Отображается меню с видами кофе ОР2: Выбор зафиксирован, цена обновляется ОР3: Цена увеличивается на стоимость выбранного объема ОР4: Цена увеличивается на стоимость сахара ОР5: Отображается итоговая стоимость (например, 150 руб.) ОР6: Напиток готовится, затем выдается сдача 50 рублей |
| Постусловия: | 1. Автомат возвращается в состояние "Ожидание" 2. На экране отображается приветственное сообщение 3. В лотке для выдачи находится готовый напиток |

Тестовый сценарий 2: Отмена заказа во время выбора

| Параметр | Значение |
| --- | --- |
| Идентификатор: | TC-COFFEE-002 |
| Название: | Отмена заказа на этапе выбора напитка |
| Предусловия: | 1. Автомат включен и находится в состоянии "Ожидание" 2. Пользователь начал выбор напитка |
| Входные значения теста: | Шаг 1: Нажать кнопку "Начать заказ" Шаг 2: Выбрать "Латте" Шаг 3: Выбрать объем "200 мл" Шаг 4: Нажать кнопку "Отмена" |
| Ожидаемый результат: | ОР1: Отображается меню с видами кофе ОР2: Выбор зафиксирован ОР3: Цена обновляется ОР4: Автомат возвращается в состояние "Ожидание", все выборы сброшены |
| Постусловия: | 1. Автомат находится в состоянии "Ожидание" 2. На экране отображается приветственное сообщение 3. Не списаны денежные средства |

Тестовый сценарий 3: Неудачная оплата картой

| Параметр | Значение |
| --- | --- |
| Идентификатор: | TC-COFFEE-003 |
| Название: | Обработка неудачной оплаты банковской картой |
| Предусловия: | 1. Автомат включен и готов к работе 2. Пользователь выбрал напиток и перешел к оплате 3. Итоговая стоимость: 120 рублей |
| Входные значения теста: | Шаг 1: Выбрать "Американо" и завершить выбор Шаг 2: Приложить заблокированную карту к терминалу Шаг 3: После ошибки нажать "Повторить" Шаг 4: Приложить карту с недостаточным балансом |
| Ожидаемый результат: | ОР1: Отображается сумма к оплате 120 руб. ОР2: Сообщение "Оплата не прошла" ОР3: Возврат к экрану оплаты ОР4: Повторное сообщение "Оплата не прошла" |
| Постусловия: | 1. Автомат остается в состоянии "Ошибка оплаты" 2. Напиток не приготовлен 3. Деньги не списаны |

Тестовый сценарий 4: Выбор напитка с дополнительными опциями

| Параметр | Значение |
| --- | --- |
| Идентификатор: | TC-COFFEE-004 |
| Название: | Выбор кофе с сиропом и альтернативным молоком |
| Предусловия: | 1. Автомат включен и готов к работе 2. В автомате есть все виды сиропов и молока |
| Входные значения теста: | Шаг 1: Начать заказ Шаг 2: Выбрать "Латте" Шаг 3: Выбрать "Карамельный сироп" Шаг 4: Выбрать "Кокосовое молоко" Шаг 5: Завершить выбор и оплатить |
| Ожидаемый результат: | ОР1: Отображается меню выбора кофе ОР2: Отображается меню сиропов (Действие 4) ОР3: Отображается меню видов молока (Действие 5) ОР4: Цена корректно рассчитана с учетом доп. опций ОР5: Напиток приготовлен с выбранными опциями |
| Постусловия: | 1. Автомат вернулся в состояние "Ожидание" 2. Пользователь получил капучино с карамельным сиропом и кокосовым молоком |

Тестовый сценарий 5: Таймаут при бездействии пользователя

| Параметр | Значение |
| --- | --- |
| Идентификатор: | TC-COFFEE-005 |
| Название: | Автоматическая отмена заказа по таймауту |
| Предусловия: | 1. Автомат включен и готов к работе 2. Пользователь начал выбор напитка |
| Входные значения теста: | Шаг 1: Начать заказ Шаг 2: Выбрать тип кофе Шаг 3: Не производить действий в течение 2 минут |
| Ожидаемый результат: | ОР1: Отображается меню выбора ОР2: Через 2 минуты бездействия система автоматически отменяет заказ ОР3: Автомат возвращается в состояние "Ожидание" ОР4: На экране отображается сообщение "Заказ отменен по таймауту" |
| Постусловия: | 1. Автомат находится в состоянии "Ожидание" 2. Все временные данные и выборы пользователя сброшены 3. Денежные средства не списаны |

### Баги кода **Баг 1: Отсутствует кнопка подтверждения выбора продукта**

**Проблема:** В интерфейсе нет кнопки для подтверждения выбора продукта, только обработка Enter.  
**Решение:** Добавить кнопку "Buy" в группу выбора продукта.

### **Баг 2: Некорректная обработка клавиши Enter**

**Проблема:** Код 16777220 - это магическое число, которое может быть нечитаемым и непереносимым.  
**Решение:** Использовать константы из Qt для лучшей читаемости.

### **Баг 3: Не сбрасывается состояние монет после покупки**

**Проблема:** После успешной покупки монеты не обнуляются, что позволяет делать multiple purchases без повторного внесения денег.  
**Решение:** Сбросить coins после успешной покупки.

### **Баг 4: Нет валидации ввода монет**

**Проблема:** Пользователь может ввести отрицательные числа или нечисловые значения.  
**Решение:** Добавить валидацию ввода.

### **Баг 5: Не обрабатывается случай, когда продукт не выбран**

**Проблема:** Если ни один продукт не выбран, функция все равно пытается проверить условия.  
**Решение:** Добавить проверку на выбор продукта.

### **Баг 6: Нет обработки множественных покупок**

**Проблема:** После покупки radio buttons остаются selected.  
**Решение:** Сбросить выбор после успешной покупки.

### **Баг 7: Плохая UX-практика - скрытая функциональность**

**Проблема:** Пользователь может не знать, что нужно нажать Enter.  
**Решение:** Добавить подсказку в интерфейс.

Исправленный код  
  
import sys

from PyQt5.QtWidgets import (QApplication, QMainWindow, QVBoxLayout,

QRadioButton, QPushButton, QLineEdit,

QLabel, QGroupBox, QWidget)

from PyQt5.QtCore import Qt

class VendingMachine(QMainWindow):

def \_\_init\_\_(self):

super().\_\_init\_\_()

self.setWindowTitle("Coffee Vending Machine")

self.setGeometry(100, 100, 600, 400)

# Основной виджет и макет

widget = QWidget(self)

self.setCentralWidget(widget)

main\_layout = QVBoxLayout(widget)

# 1. Ввод монет

self.insert\_coins\_group = QGroupBox("1. INSERT COINS")

self.insert\_coins\_layout = QVBoxLayout()

self.coin\_input = QLineEdit("0")

self.insert\_button = QPushButton("Insert")

self.insert\_button.clicked.connect(self.insert\_coins)

self.insert\_coins\_layout.addWidget(self.coin\_input)

self.insert\_coins\_layout.addWidget(self.insert\_button)

self.insert\_coins\_group.setLayout(self.insert\_coins\_layout)

# 2. Выбор продукта

self.select\_product\_group = QGroupBox("2. SELECT PRODUCT")

self.select\_product\_layout = QVBoxLayout()

self.cappuccino\_rb = QRadioButton("Cappuccino ---> 35")

self.espresso\_rb = QRadioButton("Espresso ---> 30")

self.americano\_rb = QRadioButton("Americano ---> 35")

self.latte\_rb = QRadioButton("Latte ---> 40")

self.buy\_button = QPushButton("Buy")

self.buy\_button.clicked.connect(self.select\_product)

for rb in [self.cappuccino\_rb, self.espresso\_rb,

self.americano\_rb, self.latte\_rb]:

self.select\_product\_layout.addWidget(rb)

self.select\_product\_layout.addWidget(self.buy\_button)

self.select\_product\_group.setLayout(self.select\_product\_layout)

# 3. Получение продукта

self.get\_product\_group = QGroupBox("3. GET YOUR PRODUCT")

self.get\_product\_layout = QVBoxLayout()

self.product\_label = QLabel("")

self.get\_product\_layout.addWidget(self.product\_label)

self.get\_product\_group.setLayout(self.get\_product\_layout)

# 4. Прощание

self.goodbye\_group = QGroupBox("4. Goodbye")

self.goodbye\_layout = QVBoxLayout()

self.goodbye\_label = QLabel("")

self.goodbye\_layout.addWidget(self.goodbye\_label)

self.goodbye\_group.setLayout(self.goodbye\_layout)

# Подсказка

self.hint\_label = QLabel("Press Enter or click Buy to purchase")

self.hint\_label.setStyleSheet("color: gray; font-style: italic;")

# Добавление в основной макет

main\_layout.addWidget(self.insert\_coins\_group)

main\_layout.addWidget(self.select\_product\_group)

main\_layout.addWidget(self.get\_product\_group)

main\_layout.addWidget(self.goodbye\_group)

main\_layout.addWidget(self.hint\_label)

self.coins = 0

def insert\_coins(self):

try:

coins = int(self.coin\_input.text())

if coins < 0:

self.product\_label.setText("Coin amount cannot be negative!")

return

self.coins = coins

self.product\_label.setText("")

self.goodbye\_label.setText("")

except ValueError:

self.product\_label.setText("Invalid coin input! Please enter a number.")

def select\_product(self):

product = None

cost = 0

if self.cappuccino\_rb.isChecked():

product = "Cappuccino"

cost = 35

elif self.espresso\_rb.isChecked():

product = "Espresso"

cost = 30

elif self.americano\_rb.isChecked():

product = "Americano"

cost = 35

elif self.latte\_rb.isChecked():

product = "Latte"

cost = 40

if not product:

self.product\_label.setText("Please select a product first!")

return

if self.coins >= cost:

change = self.coins - cost

self.product\_label.setText(f"Dispensed: {product}\nChange: {change}")

self.goodbye\_label.setText("Have a nice day!")

self.coins = 0

self.coin\_input.setText("0")

# Сбросить выбор

for rb in [self.cappuccino\_rb, self.espresso\_rb,

self.americano\_rb, self.latte\_rb]:

rb.setAutoExclusive(False)

rb.setChecked(False)

rb.setAutoExclusive(True)

else:

self.product\_label.setText("Not enough coins!")

def keyPressEvent(self, event):

if event.key() == Qt.Key\_Enter or event.key() == Qt.Key\_Return:

self.select\_product()

if \_\_name\_\_ == "\_\_main\_\_":

app = QApplication(sys.argv)

window = VendingMachine()

window.show()

sys.exit(app.exec\_())