H-0Y

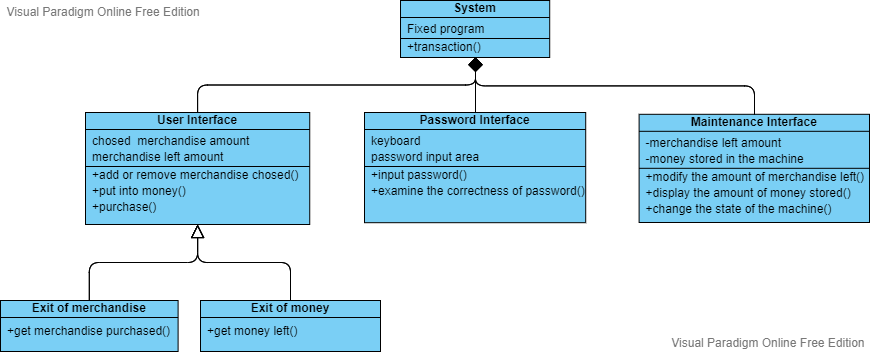
Team 7

sOFTWARE VALIDATION

Vending Machine

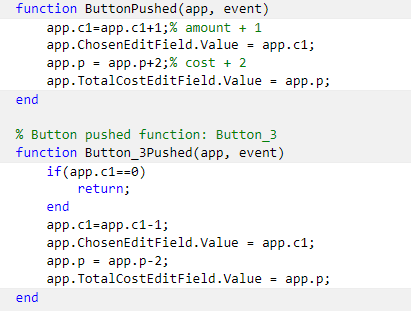
**System Architecture**

**The system architecture is shown below:**

****

**T1: Main Interface Test**

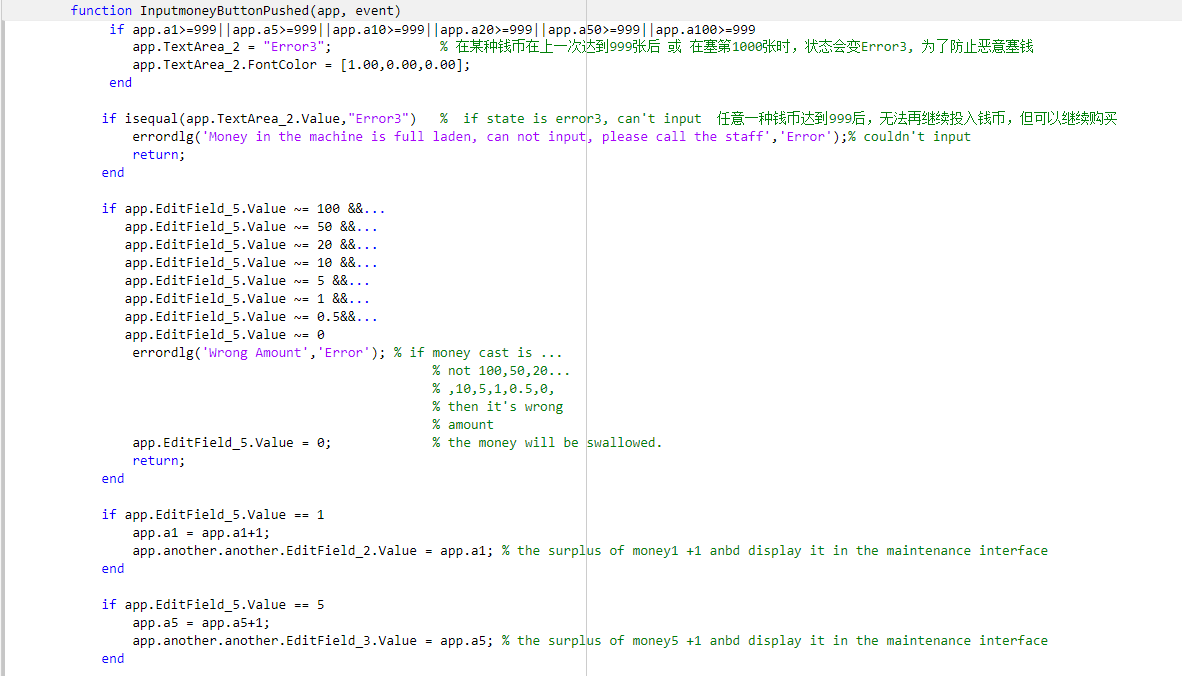
**T1.1: Test buttons for modifying amounts of chosen items**

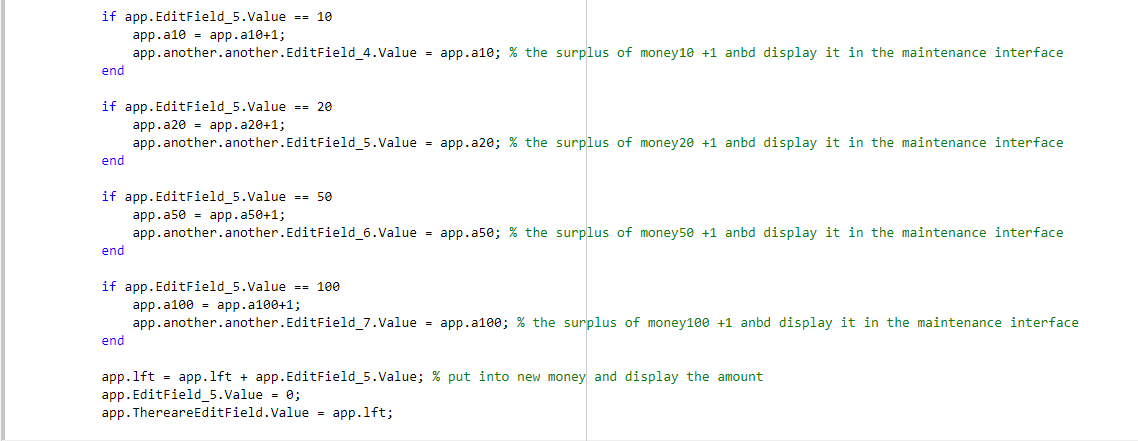
****

**For item1, when pressing button +, the parameter representing the amount of chosen item1 will be added by one. And when pressing button -, the parameter representing the amount of chosen item1 will be subtracted by one. Then these changes will be shown on the pane. No need to show every figure presenting its changes, which will be too tedious.**

**It’s the same for other items.**

**T1.2: Test buttons for inputting money in certain denomination.**

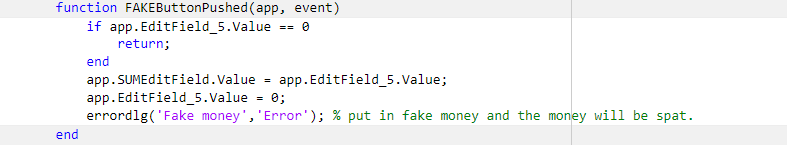
****

****

**Typing certain numbers (0, 1, 5, 10, 20, 50, 100) representing money in different denomination. And then press the “Input money” button to input money. Or it will report error.**

**And as we write in the codes, there will be a parameter in the system to record the money users put in and also record the remaining quantity of their money. And there are 6 numbers to record the amounts of money stored in the machine.**

**T1.3: Test Fake money input button**

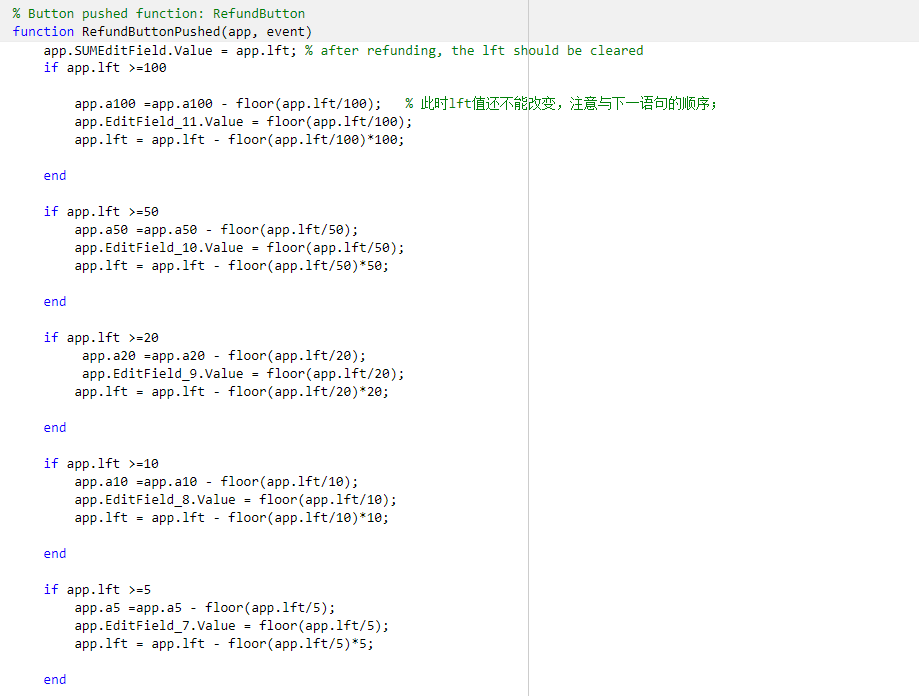
****

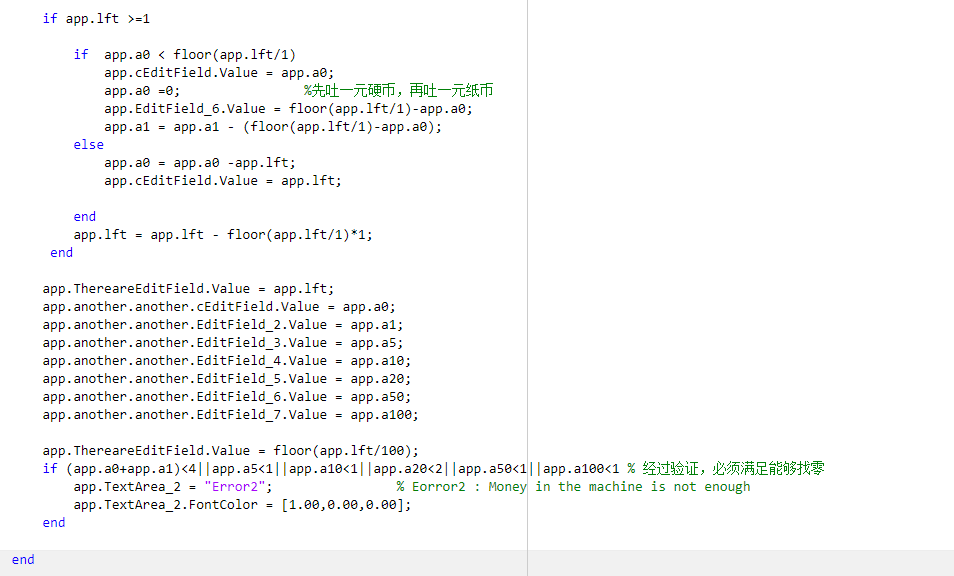
**Users could simulate putting in fake money by press the “FAKE” button.**

**Users could input any amount of fake money they want in the numerical box. Then press the “FAKE” button.**

**The system will judge this money as fake money, the parameter in the machine won’t change, just like nothing is putting in. And naturally, the fake money cannot be retrieved by “Refund” button.**

**T1.4: Test Buy button**

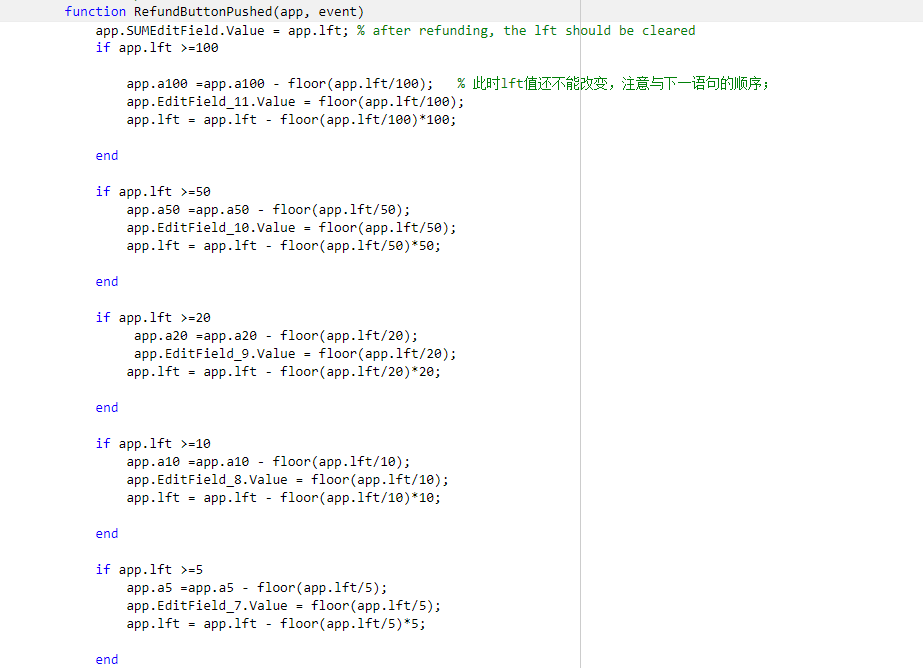
****

****

**On the premise of successful purchase, the parameter recording the remaining** **money will be subtracted by the total cost of all the merchandise purchased after pressing the “Buy” button.**

**On the premise of successful purchase, all the merchandise purchased will be dropped out from the “Exit of merchandise”. Precise number will be displayed in the pane.**

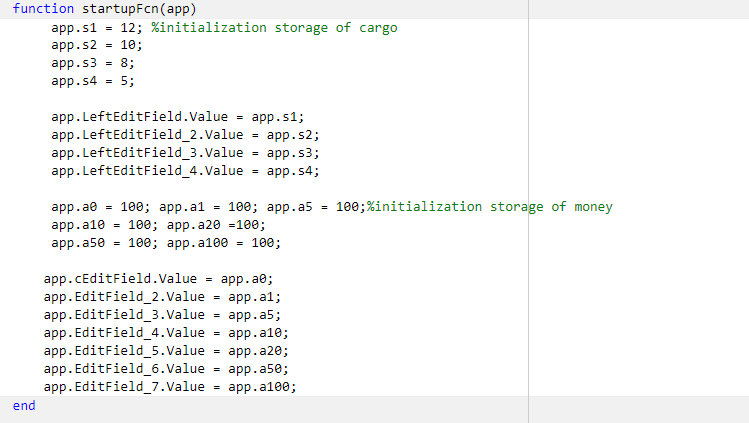
**T1.5: Test Refund button**

****

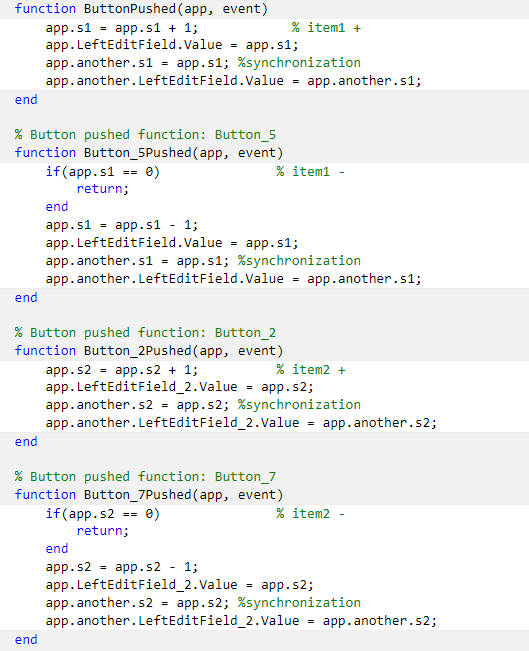
**The remaining money will be dropped put from the “Exit of money”. User can see the certain amount of money being displayed on the exit pane. The parameter recording the remaining will be cleared to 0. And many of the other related parameters will also be reset to 0.**

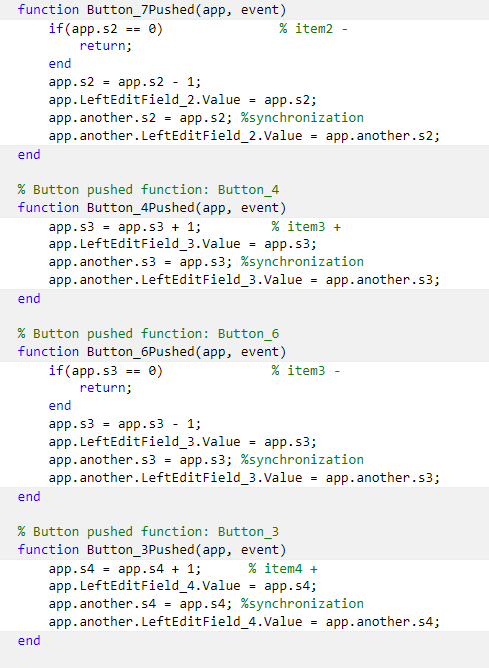
**T2: Main Interface Test**

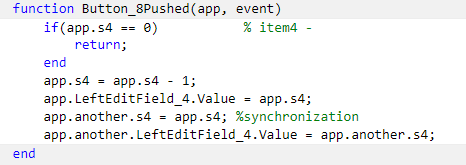
**T2.1: Test buttons for modifying amounts of items stored**

****

**Set the initial values.**

****

****

****

**Add the amount of the remaining merchandise**

1. **Press “+” button under the picture of the certain merchandise.**
2. **There is a parameter representing the amount the remaining merchandise which is displayed in the box and set to a certain number initially.**
3. **When the “+” button is put, we will add 1 to the parameter, indicating that the amount of this is added 1.**

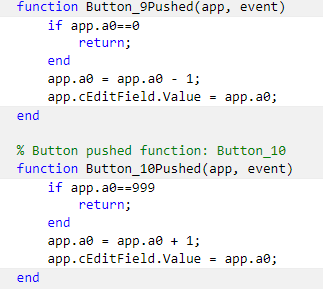
**Subtract the amount of the remaining merchandise**

1. **Press “-” button under the picture of the certain merchandise.**

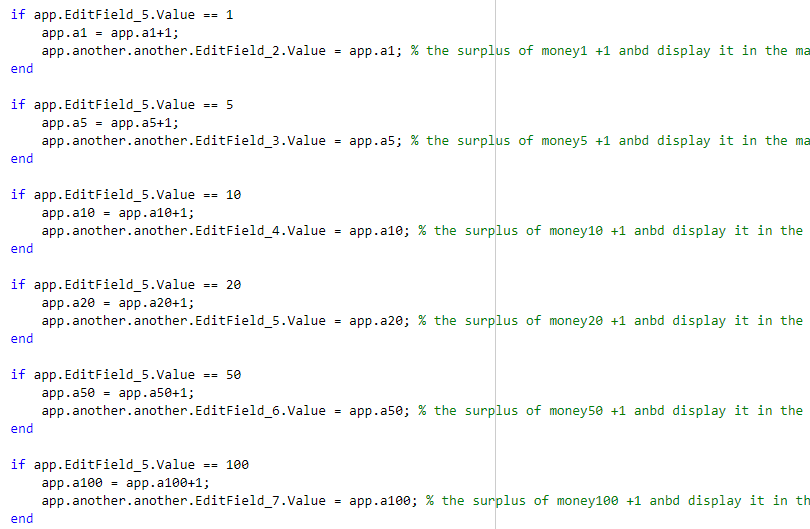
**There is a parameter representing the amount the remaining certain merchandise which is displayed in the pane.**

1. **When the “-” button is put, we will subtract this parameter by one, indicating that the amount of this is reduced by 1.**

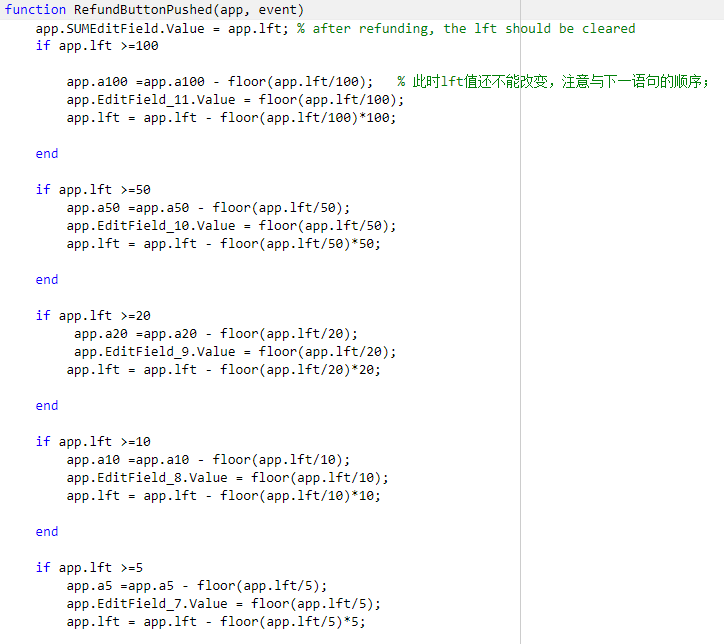
**T2.2: Test the storage of money in different denomination**

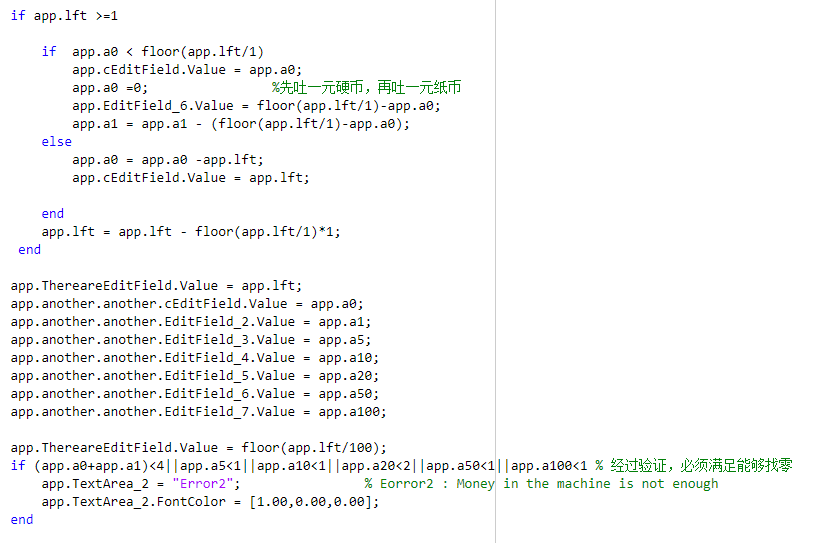
****

**Maintainers can modify the amount of money stored in the machine via the two buttons near the screen. But the amount couldn’t exceed the limit of [0, 99].**

****

**Meanwhile, when inputting money, the system will also count the amount of money being stored, and store it into correct container.**

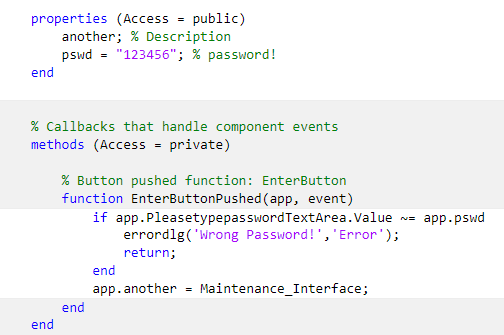
****



**When refunding, the system will start the process at 100￥, and then 50￥, and then 20￥…… In this case, we can guarantee the refund process efficient and correct. And the money being retrieved will also will be recorded in order to make the amount of money stored precise.**

**T3: Password Interface Test**

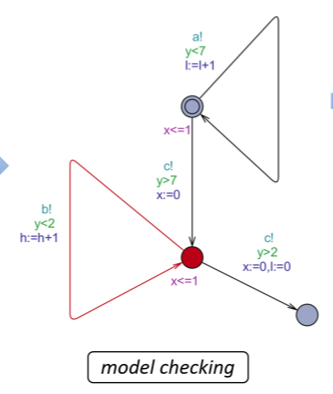
**T3.1: Test Correctness**

****

**If the numbers users input are the same with the right password, the system will open another interface, or the system will report error, and won’t do anything.**

**T4: Validation Test**

**T4.1: Test the correctness money inputting**



**T4.2: Test the machine error condition**

