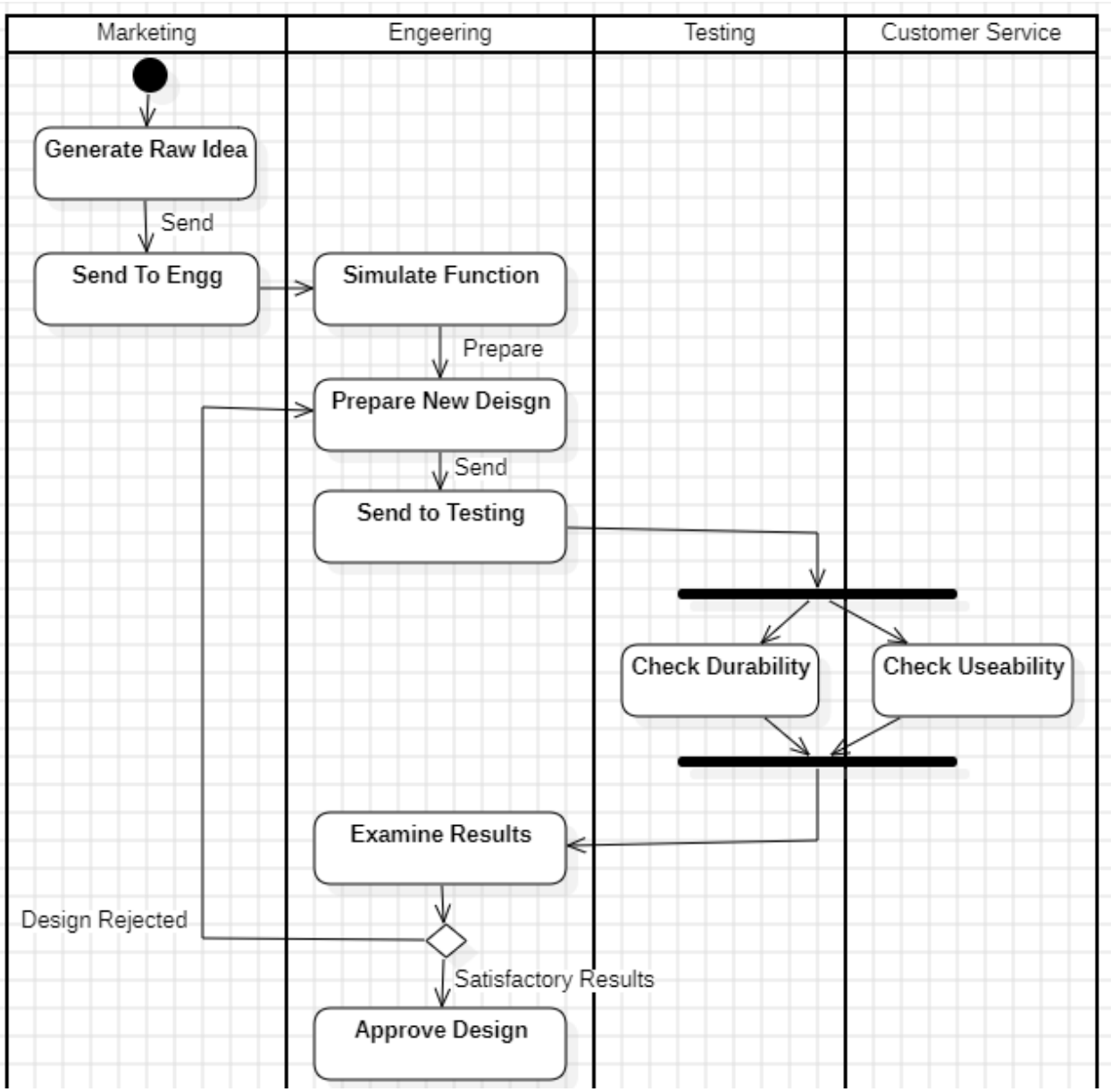
Abdullah Tahir 21L-5419 Assignment # 3

Question # 1

To approve the design for a company's new product, coordination is required amongst the different departments of the company. The marketing department first generates a raw idea which is then sent to the engineering department. The engineering department simulates the function of the product and then prepares a design. This design is then checked for durability by the testing department. At the same time, the customer service department checks the usability of the design. Once both of these departments have checked the design, the engineering department examines the results. If the results are satisfactory, the engineering department approves the design. Otherwise, the engineering department goes back to prepare a new design. The same steps are repeated until the design is finally approved. Construct a swim-lane activity diagram below for the product design approval process described above.

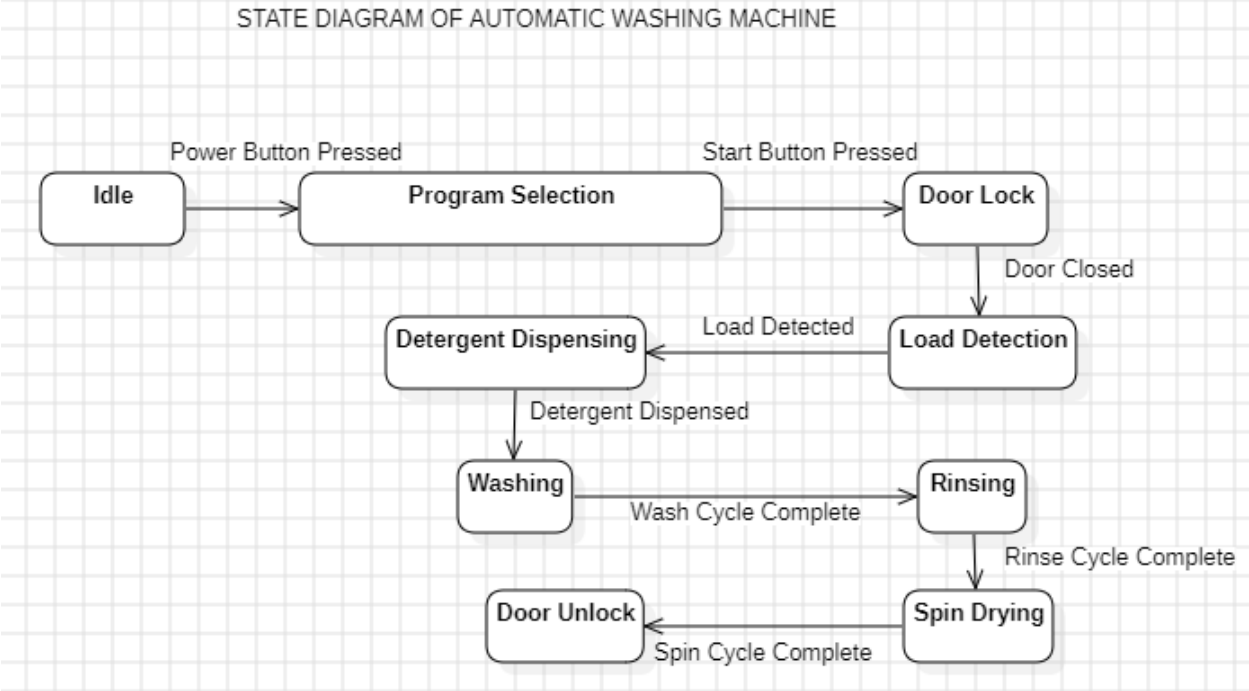
Solution:



Question # 2

Develop a state diagram for any automatic washing machine near you.

Solution:

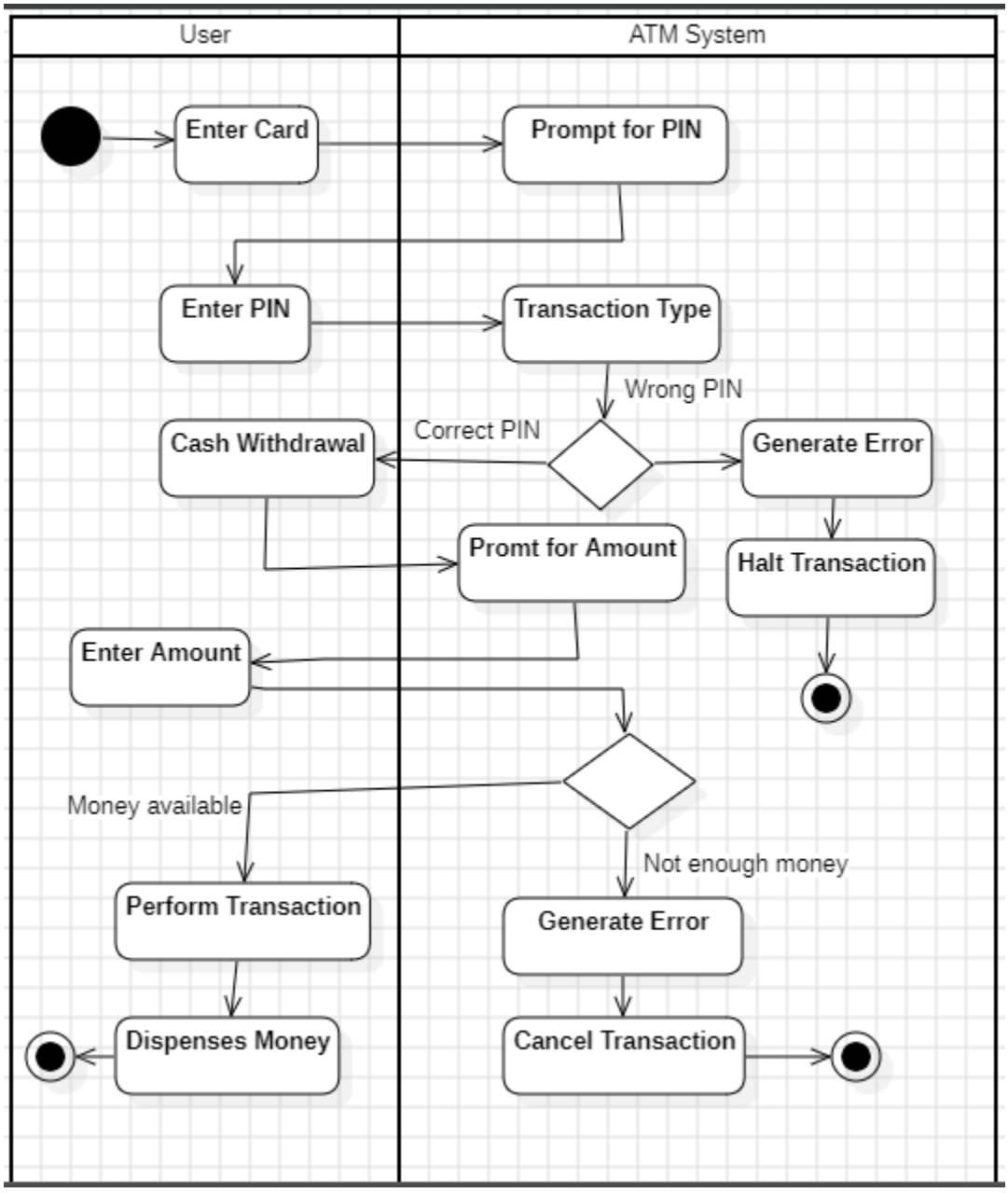


Question # 3

Consider the following use case description:
Use case: Withdraw cash
System: ATM
Steps:
i. User enters the card
ii. System prompts for the PIN code
iii. User enters the PIN code
iv. System asks for type of transaction
v. User selects "Cash withdrawal" option
vi. System asks for amount of money
vii. User enters a number
viii. System performs the transaction and dispenses cash
Alternate paths:
iv (b) If the PIN code is incorrect the system generates an appropriate error, and halts the transaction
viii (b) If there is not enough money in the user's account the system shall display an error and shall cancel the
transaction

Now your task is to give an Activity diagram to graphically depict the above use case. You may like to use swim-lanes.

Solution:

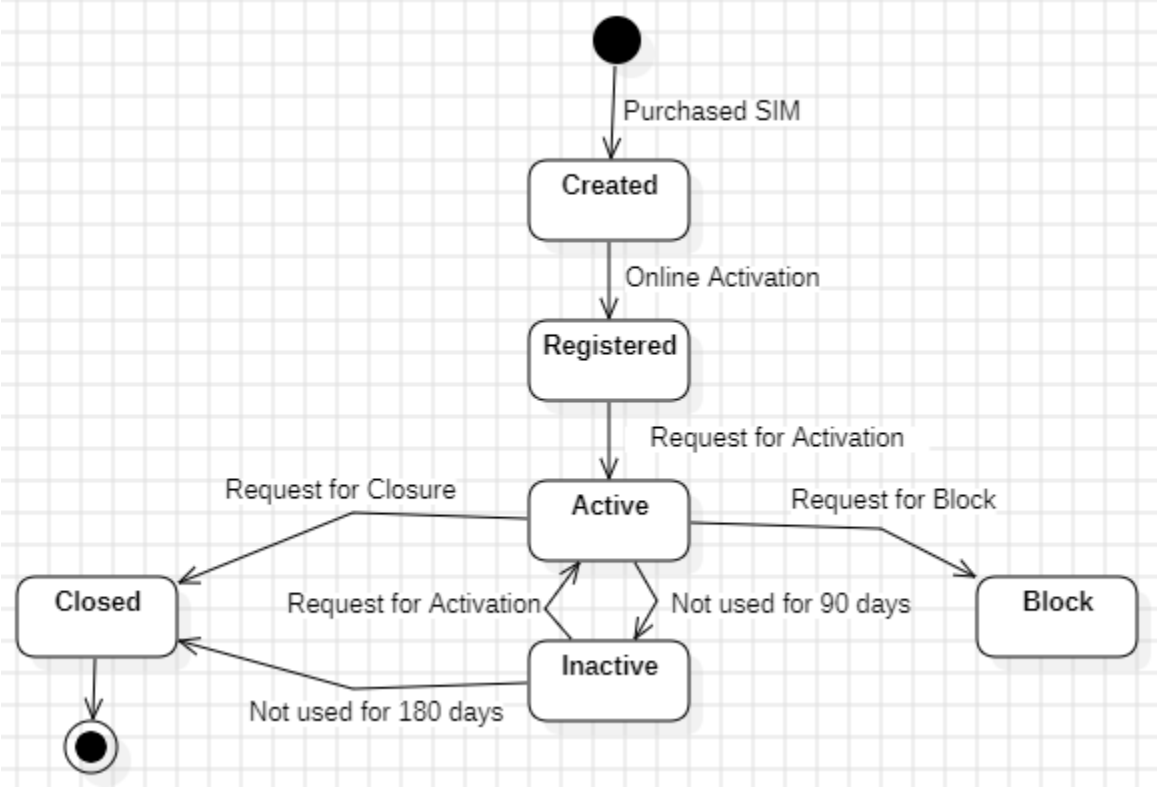


Question # 4

You want a software module to be developed that manages mobile SIM cards. The software module will be developed by a vendor to which you have to communicate your business rules. The business rules must capture details of all the states of the SIM card and on what events/actions the state of

the SIM card can be changed. Develop a state diagram showing all the states of the SIM card and transitions between them. You must show at least 5 states [Created, Registered, Active, Inactive, Closed] The SIM card can transition between these states based on certain rules that you need to define and capture in the state diagram. The vendor will use this state diagram to correctly implement the module. Consider using events/actions like SIM-purchase, online-activation, not-used-for-90-days, not-used-for-180-days, request for activation, request for block, request for closure etc.

Solution:



Question # 5

Write C++ code required for the following program. Use virtual inheritance to resolve the diamond problem.

```
int main() {
   Student s1("Aslam", "BSCS");
   Teacher t1("Zahid", "Assistant Professor");
   TA ta1("Nasir", "BSEE", "TA", 1000);
   s1.print();
   t1.print();
   ta1.print(); // prints all the data members (including inherited ones)
```

```
    return 0;
}
```

Solution:

```
#include <iostream>
#include <string>

using namespace std;

class Person {
protected:
    string name;

public:
    Person(const string& _name) : name(_name) {}

    void print() {
        cout << "Name: " << name << endl;
    }
};

class Student : public virtual Person {
private:
    string program;

public:
    Student(const string& _name, const string& _program)
            : Person(_name), program(_program) {}

    void print() {
        Person::print();
        cout << "Program: " << program << endl;
    }
};

class Teacher : public virtual Person {
private:
    string designation;

public:
    Teacher(const string& _name, const string& _designation)
            : Person(_name), designation(_designation) {}

    void print() {
        Person::print();
```

```
        cout << "Designation: " << designation << endl;
    }
};

class TA : public Student, public Teacher {
private:
    int salary;

public:
    TA(const string& _name, const string& _program, const string& _designation, int _salary)
        : Person(_name), Student(_name, _program), Teacher(_name, _designation), salary(_salary) {}

    void print() {
        Person::print();
        Student::print();
        Teacher::print();
        cout << "Salary: " << salary << endl;
    }
};

int main() {
    Student s1("Aslam", "BSCS");
    Teacher t1("Zahid", "Assistant Professor");
    TA ta1("Nasir", "BSEE", "TA", 1000);

    s1.print();
    t1.print();
    ta1.print();

    return 0;
}
```