Name CHEN-YUAN HO Username: N10912051 Date: November 23, 2017
Section: 001

Assignment 6-2 (Programming)

Assignment Layout (25%)

o Assignment is neatly assembled on 8 1/2 by 11 paper.
o Cover page with your name (last name first followed by a comma then first name), username and section number with a signed statement of independent effort is included.
o Program and documentation submitted for Assignment #6-2 are satisfactory. o File name is correct.

Total in points _____

Professor's Comments:

Affirmation of my Independent Effort: _____
(Sign here)

Project Report

I. Requirements

1. FSM.jar: given by the professor.
2. Java SE 8.

II. Entry Files

FiniteStateMachine_HW.java

III. Usage Information

1. Unzip the java source code from FSM.jar
2. Javac -cp .;./Fsm.jar Main.java
3. java -cp .;./Fsm.jar Main
4. To exit the program, input EOF or an empty string, for input from files, the program will exit when the read buffer reaches the end of file.

IV. Classes

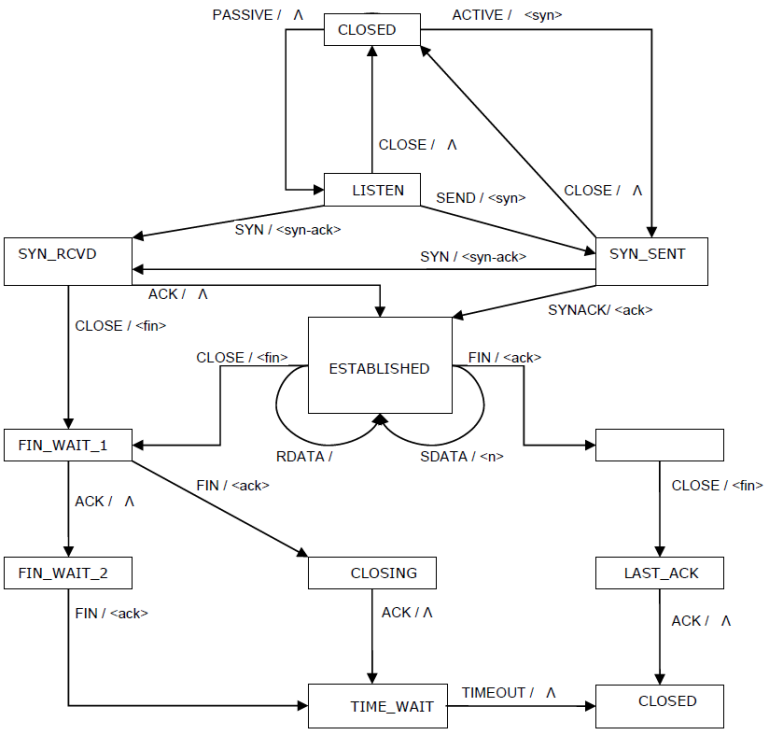
According to the project requirement, I designed three classes which extend the base classes from Fsm.jar.

1. State Class

   The state classes extend the State class from Fsm.jar which simulate the state in the network communication. In this project, the class which simulate state are listed as follows:

Class Name	Base Class
Transition	State
Timeout	
Time_Wait	
Syn_Sent	
Syn_Received	
Last_Ack	
Fin_Wait_1	
Fin_Wait_2	
Closing	
Closed	

Close	
Blank	



2. Event Class

   The event classes extend the Event class from Fsm.jar which simulate the event in the network communication. In this project, the class which simulate event are listed as follows:

Class Name	Base Class
TCPEvent	Event

3. Action Class

   The event classes extend the Event class from Fsm.jar which simulate the event in the network communication. In this project, the class which simulate event are listed as follows:

Class Name	Base Class
Ack	Action
Active	
Sdata	
Rdata	
TCPAction	

SynAck	
Syn	
Passive	
No_Action	
N	
Fin	

V. Termination

Initially, I set quit (termination) is false Boolean value. Then I keep requesting for actions until the user quits. For requesting events, I keep request it until a valid event for the current state is entered.