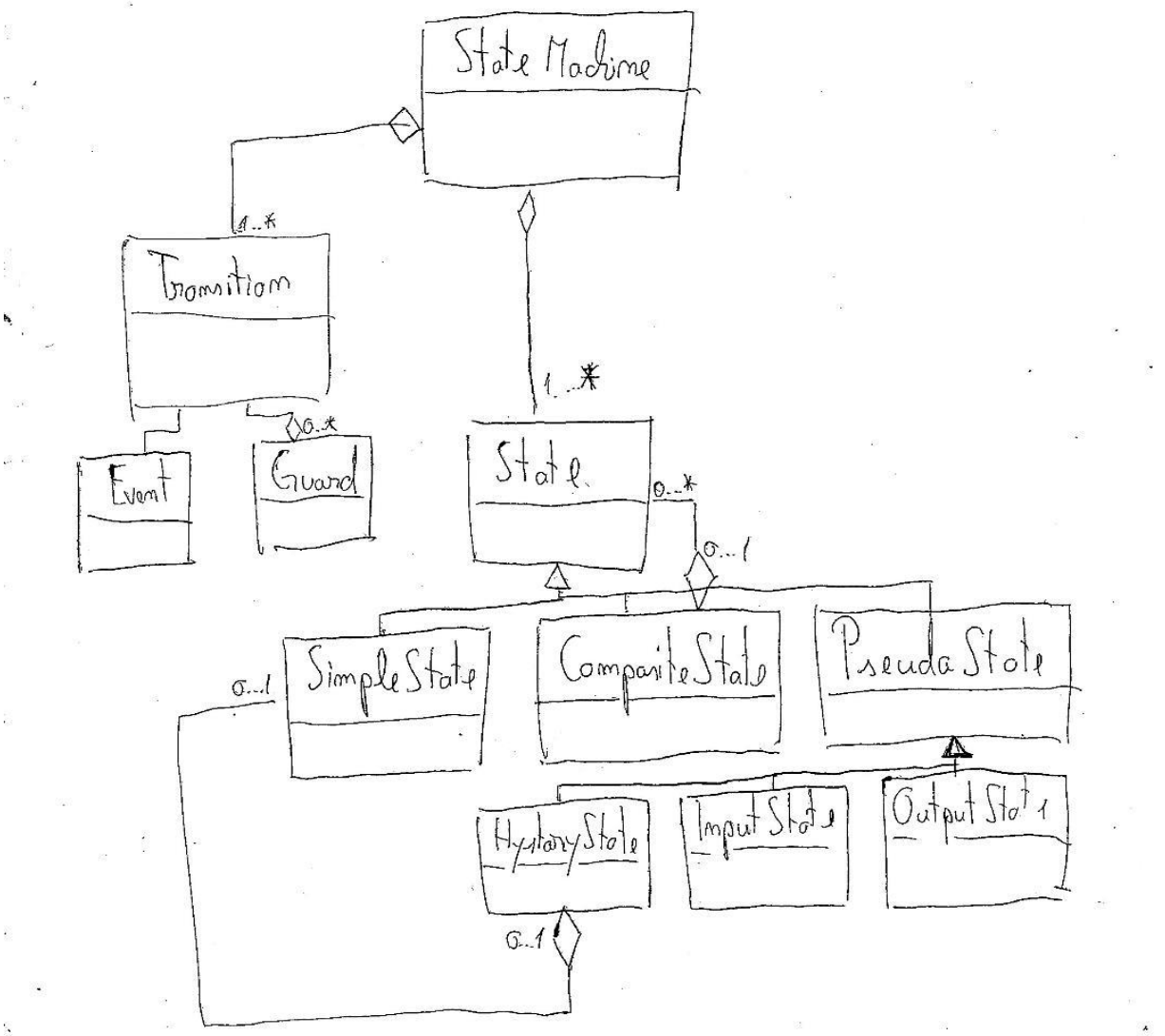
i.


State Machine
1..*
Transition
1..*
0..*
Event
Guard
State
0..*
0..1
0..1
SimpleState
CompositeState
PseudoState
HystoryState
InputState
OutputState
0..1

a)

ii. In this state machine diagram we have a composite state CS1 and 4 other states A, A1, A2, C and a pseudo state H. 4 events e1, e2, interrupt and resume.
If the system is A state then if an e1 event is triggered the system will transition to state A1. Then from A1 to A2 if event e2 is triggered and again to A1 if e1 is triggered.
If the system is in composed state CS1 and an interrupt event will appear, then a transition will be executed towards state C. If the resume event is triggered the system will transition through the history state back to state A2

iii. In my opinion using this diagram we can't generate code that is 100%. Firstly, the name for the association ends between "Paper" and "Abstract" classes are "abstract" and "paper". The word "abstract" is a reserved keyword in java and when the code will be auto generated the field it will use a reserved keyword. The second problem is with the class "PCCo-Chair" it contains the character "-". In java "-" is used to represent the operator minus and it can't not appear in class names or other definitions outside arithmetic operations. In a similar way the name of the association end "pCCo-Chairs" between "PCCo-Chair" and "Conference" will not be valid code in java.

iv. The diagram is a communication diagram. In it we have lifeline names(product, selectionPolicy), classNames(Product, Company, Warehouse), messages(getPropertyBy(), assignStockLocation()), sequence expression(1,2…7) and links(the lines between objects)