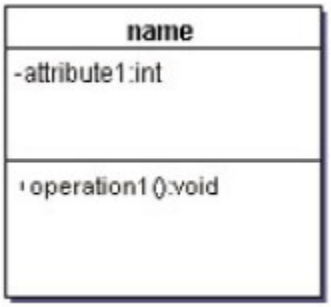
LIYANA ARACHCHIGE RANIL

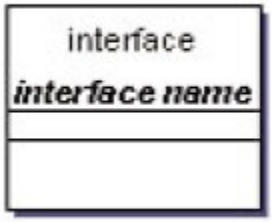
SHORT NOTES / UML

1. MODEL is a simplification of REALITY
2. There are mainly THREE building blocks of UML
3. **ELEMENTS** , **RELATIONSHIPS** and **DIAGRAMS [ERD]**
4. ELEMENST are the abstractions that are FIRST CLASS citizens in a MODEL
5. RELATIONSHIPS tie these ELEMENTS together
6. DIAGRAMS groups collection of related elements by means of RELATIONSHIP

ELEMENTS

1. ELEMENTS – There are FOUR types of ELEMENTS
    a. STRUCTURAL – Used to create the static part of the MODEL
    b. BEHAVIOURAL – To model the behavior of the system
    c. GROUPING – To organize the structural and behavioral elements
    d. ANNOTATIONAL – Explanatory parts of the model
2. STRUCTURAL – class , interface , collaboration , use case , active class , component , node
3. CLASS
    e. 

4. INTERFACE

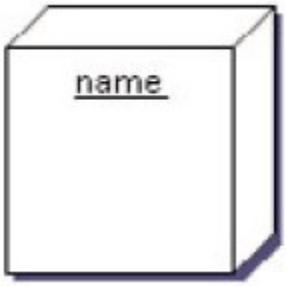


f.

5. COMPONENT

g.

6. NODE

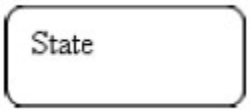


h.

7. BEHAVIOURAL ELEMENTS – Used to model the system behavior
8. There are TWO major types , INTERACTION and STATE MACHINE
9. Interaction – a set of MESSAGES exchanged among set of OBJECTs within a particular CONTEXT
10. INTERACTION

i.

11. state machine – specifies a SEQUENCE of STATES of an OBJECT
12. STATE MACHINE

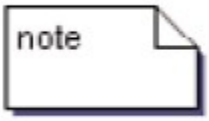


j.

13. GROUPING ELEMENTS – There is only ONE type of GROUPING elements
14. that is PACKAGE
15. PACKAGE

k.

16. ANNOTATIONAL ELEMENTS – There is only ONE type
17. NOTE is an annotation element



l.

RELATIONSHIPS

1. There are FOUR standard RELATIONSHIPS
2. DEPENDENCY , ASSOCIATION , GENERALIZATION and REALIZATION
3. DEPENDENCY – semantic relationship between two elements, if one changed that effect the other. [Dashed line with an arrow]

a. 

b. class1 has a REFERENCE to class2, either passed as a METHOD parameter or defined as a METHOD VARIABLE

4. AGGREGATION – Aggregation is a SPECIAL kind of ASSOCIATION
    a. Aggregation represents WHOLE PART relationship and this is a WEAK “HAS A” relationship

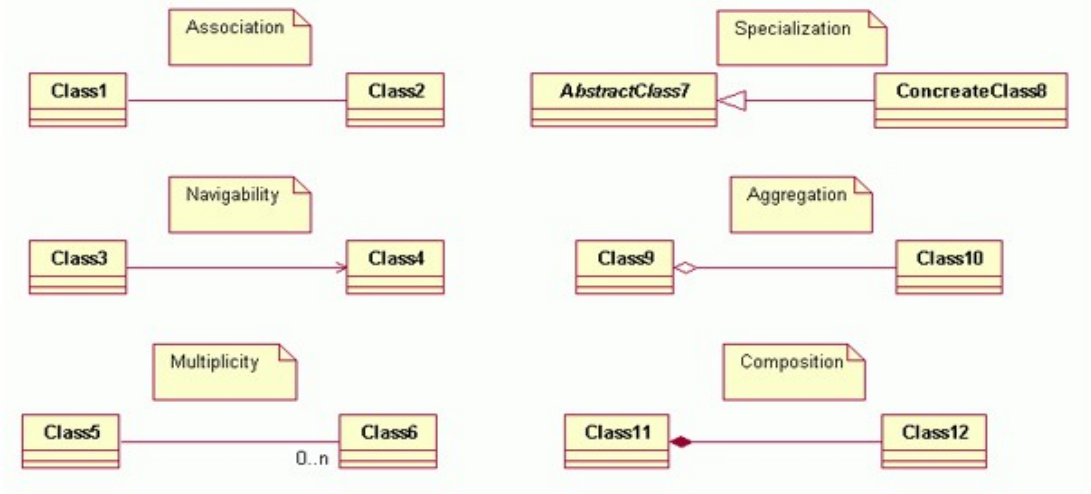
b. 

c. COMPANY as a whole comprised of its PARTS
d. COMPONENT may survive without the aggregate object
e. The COMPONENT object may be accessed without going through the AGGREGATE object
f. The AGGREGATE object DOES NOT TAKE PART in the lifecycle of the COMPONENT object
g. Example – History class object that has Students as aggregates , but if the History class object is destroyed , still the STUDENT would exist

h. 

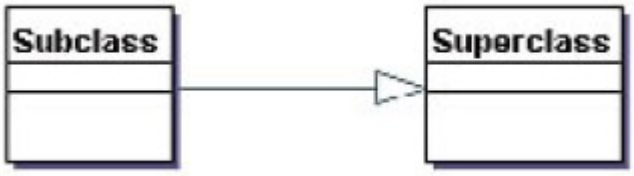
5. COMPOSISTION – Composition is also a special kind of ASSOCIATION
    a. Very similar to AGGREGATION , but composite object has sole responsibility for the DISPOSITION of the COMPONENT PARTS
    b. The RELATIONSHIP between the COMPOSITE and the COMPONENT is a “strong” HAS A relationship
    c. 

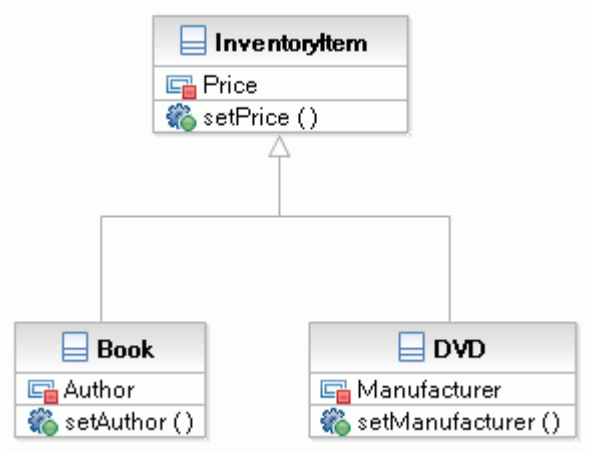
    d.



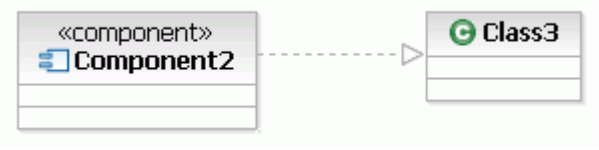
{Comp – Strong – COS}

6. GENERALIZATION – Parent/super class , child/subclass
    a. GENERALIZATION

b. 

c. 

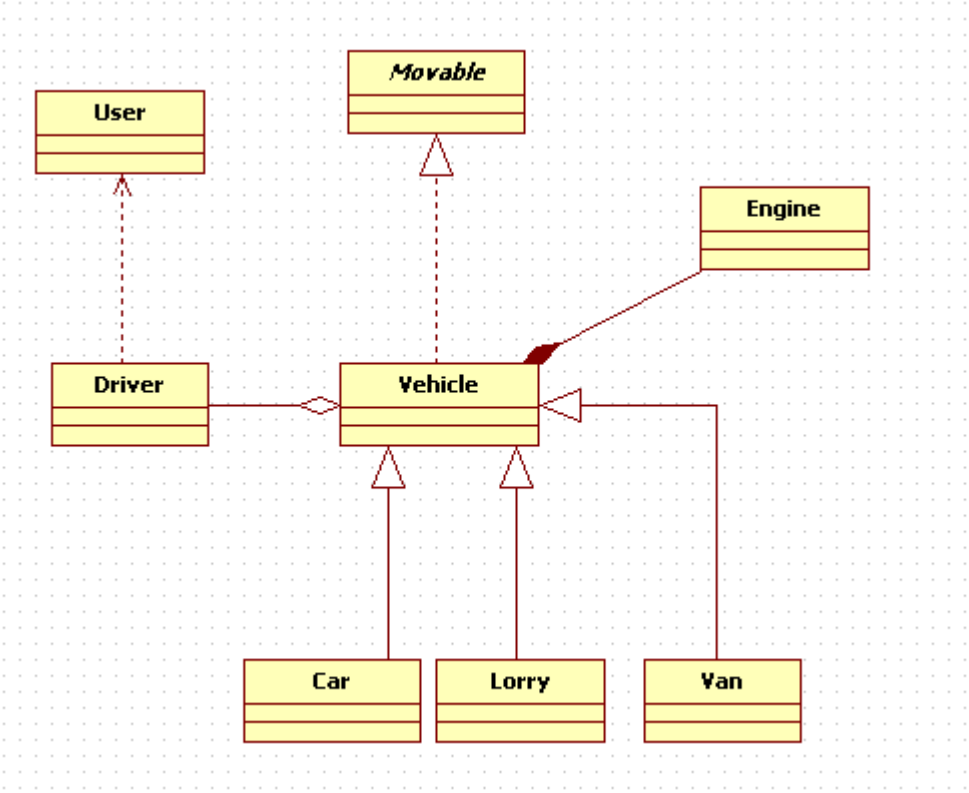
7. REALIZATION – implementation of interfaces

a. 

UML DIGRAMS

1. there are TWO main categories
    a. STRUCTURAL and BEHAVIORAL
    b. BEHAVIORAL diagrams has a sub category called INTERACTION DIAGRAMS
2. STRUCTURAL DIGRAMS – Describe the components that make up the system. Used to communicate the OVERALL structure of the system to the developers
3. BEHAVIORAL DIGRAM – describe the processing of the system

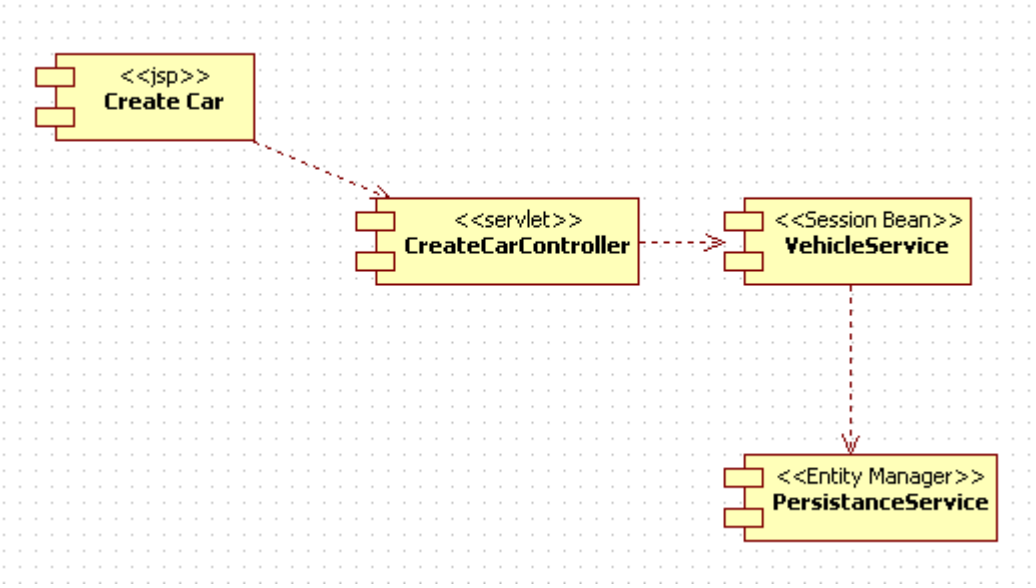
4. INTERACTION DIAGRAMS – describe the FLOW of CONTROL and DATA among the system components
5. STRUCTURAL DIAGRAMS
    a. CLASS DIGRAM

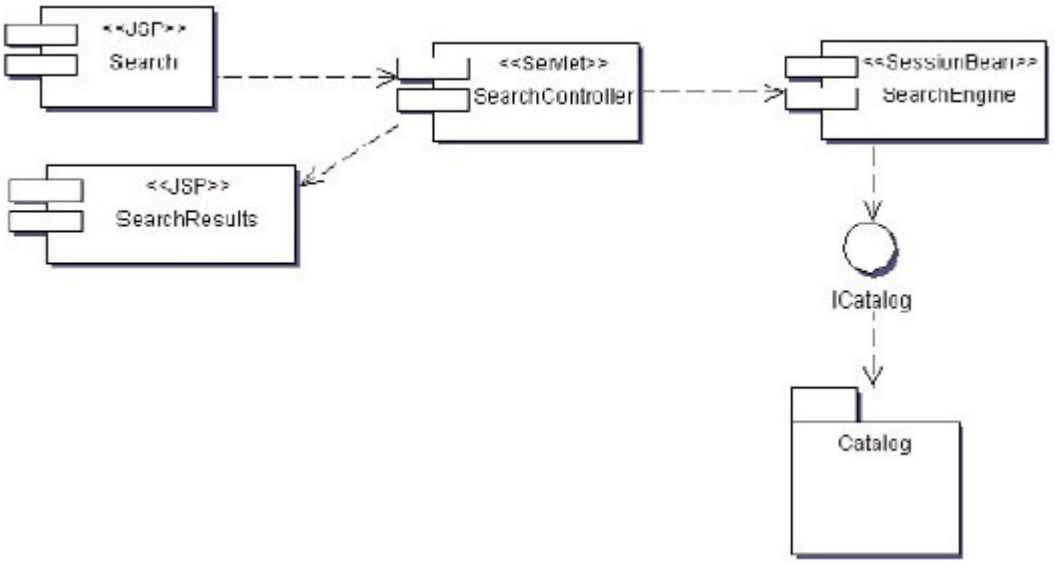


b. COMPONENT DIAGRAM – shows the organization and dependencies among a set of components. Component diagrams address STATIC IMPLEMENTATION view of the system. Think how you are going to code those , as a

LIYANA ARACHCHIGE RANIL

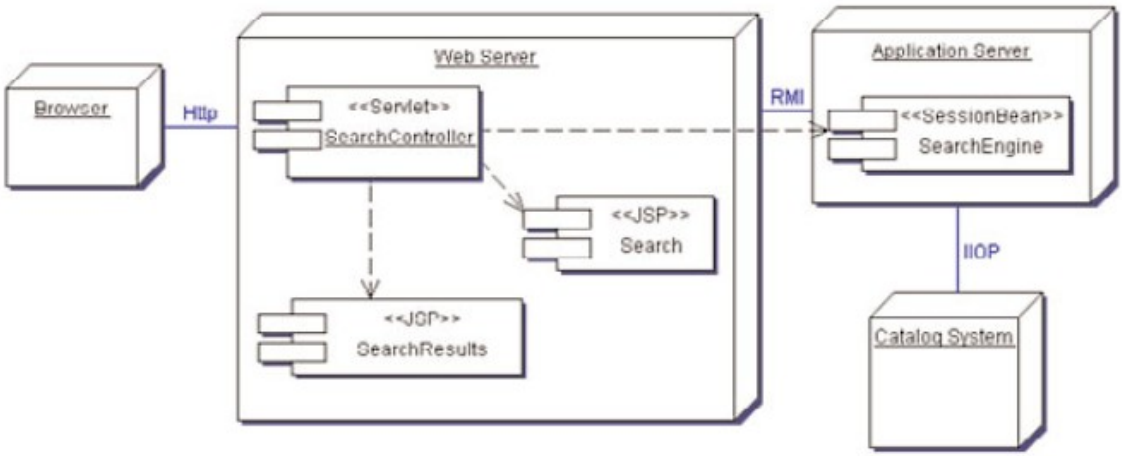
JSP , Servlet or Bean or what

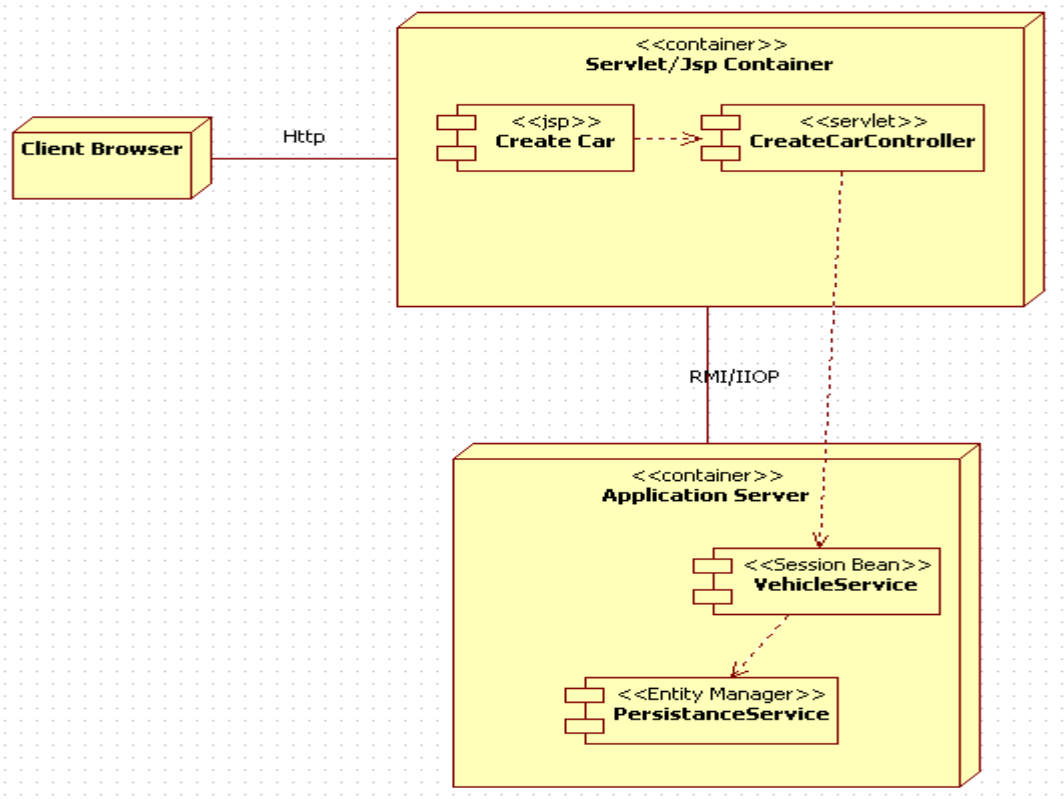




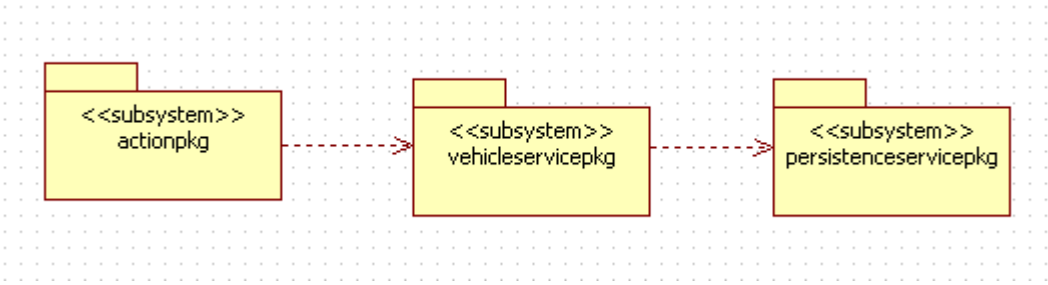
c. DEPLOYMENT DIGRAM – Shows the configuration of RUN-TME processing nodes and the components that live on these nodes. Deployment diagrams address the STATIC DEPLOYMENT VIEW of an architecture

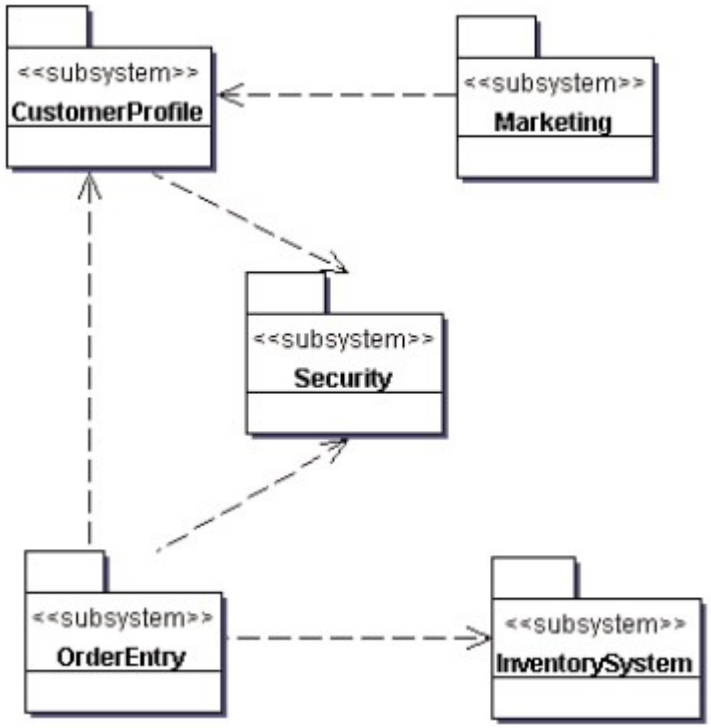
Typically a NODE encloses ONE or MORE component



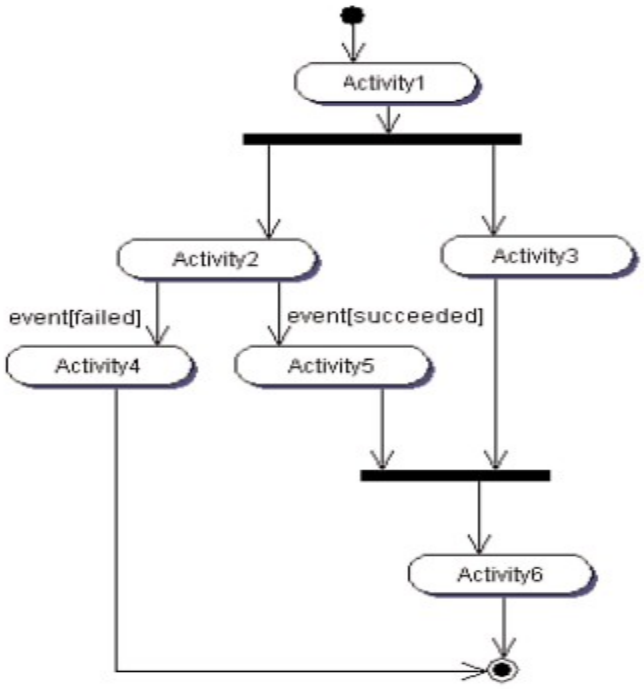


d. PACKAGE DIAGRAM – physical packages you expect within the system. These are used to communicate the PACKAGING of the software for BUILD and DEPLOYMENT

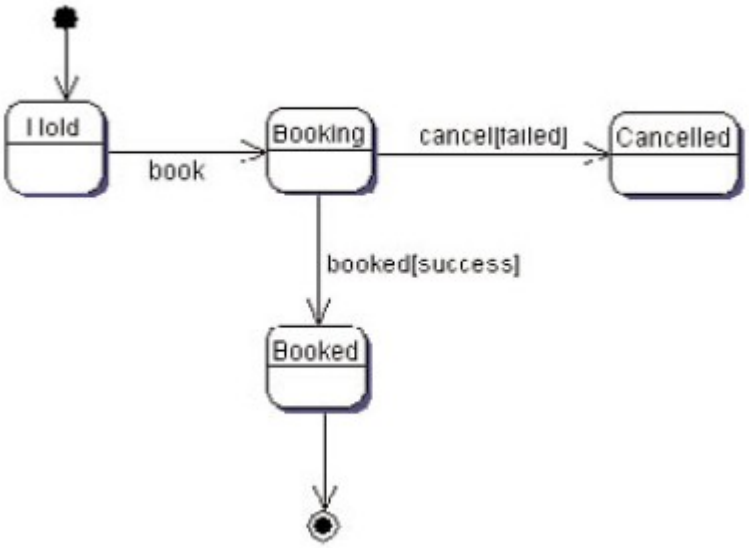




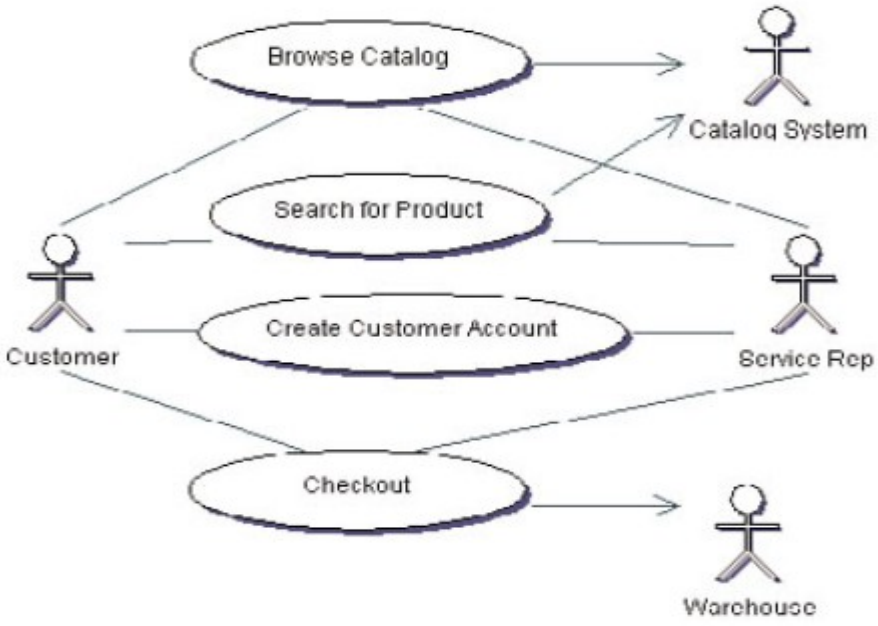
6. BEHAVIORAL DIAGRAM
    a. ACTIVITY DIAGRAM – shows FLOW from ACTIVITY to ACTIVITY within the system



b. STATE CHART DIAGRAM – shows a state machine consisting of states, transitions, events and activities. Define different states of an OBJECT during its lifetime. Useful in modeling REACTIVE systems. Behavior of a SINGLE object across MANY USE CASES

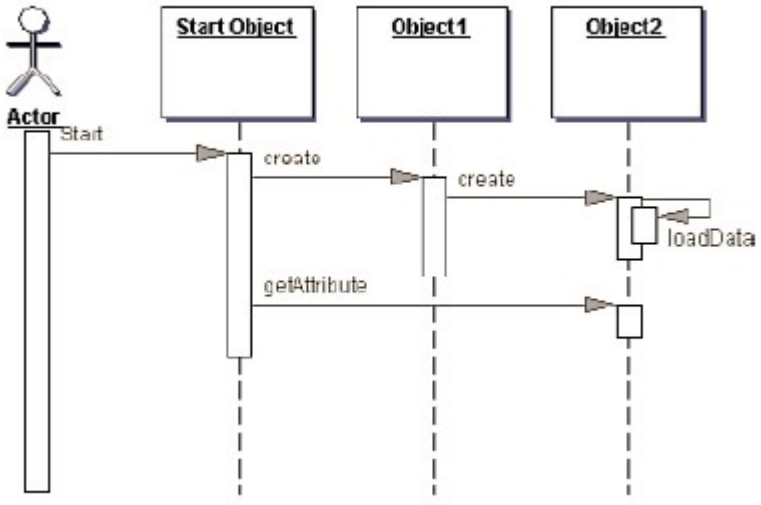


c. USE CASE DIAGRAM – shows set of USE CASES ,actors , and their RELATIONSHIPS. Addresses the STATIC use case view of the system

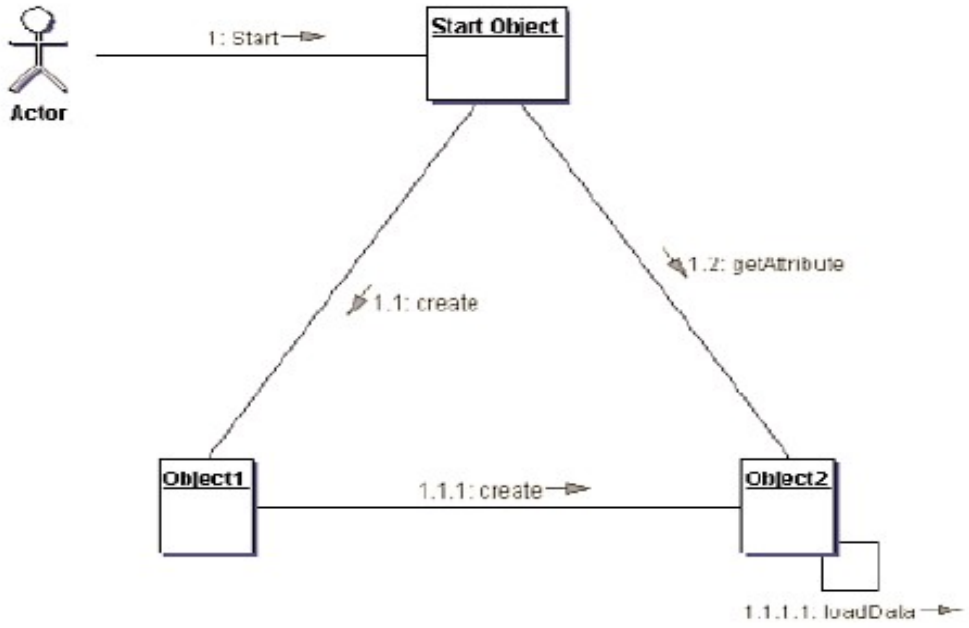


7. INTERACTION DIAGRAMS

a. SEQUENCE DIAGRAM – time ordering of messages. You should use sequence diagram when you want to look at the behavior of SEVERAL OBJECTS within a SINGLE USE CASE



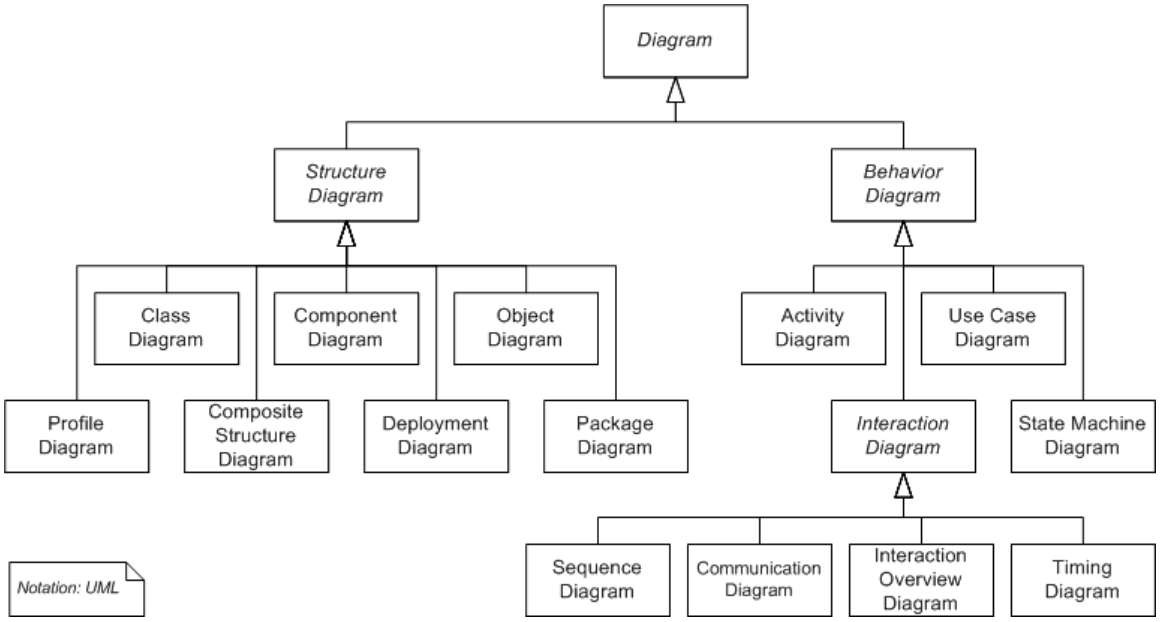
b. COLLABORATION DIAGRAM / COMMUNICATION DIGRAM– structural organization of objects. Messages are numbered to identify the sequence



c. Sequence and Collaboration diagrams are ISOMORPHIC , one can be transformed to the other
d. All INTERACTION DIGRAMS start with an ACTOR

LIYANA ARACHCHIGE RANIL

8.



LIYANA ARACHCHIGE RANIL