Lab 1

2. UML este o modalitate standardizată de a vizualiza sisteme complexe, cum ar fi arhitectura software sau bazele de date, și de a face ușor de înțeles relațiile, caracteristicile și comportamentele componentelor.

3. Un **model** este o reprezentare simplificată a unui sistem, creată pentru a înțelege, analiza sau proiecta acel sistem. Modelele ajută la abstractizarea detaliilor complexe și oferă o imagine clară asupra structurii și funcționării sistemului. Utilitatea modelelor este redata de clarificarea cerintelor , facilitarea comunicarii si reducerea riscurilor.

5. In capitolul 2 sunt reprezentate structure diagram si behaviour diagram. Diferenta dintre ele este ca behaviour diagrams reprezinta modul in care starea unui obiect se poate schimba in timp prin anumite actiuni: "Behavior refers to the direct consequences of an action of at least one object. It affects how the states of objects change over time. Behavior can either be specified through the actions of a single object or result from interactions between multiple objects." in timp ce structure diagrams nu ia in considerare acest "behaviour" dinamic al obiectelor: "The dynamic behavior of the elements in question (i.e., their changes over time) is not considered in these diagrams.

6. 6. Class diagram : modeleaza clasele care apar intr-un sistem si relatiile dintre aceste clase.

Use case diagram: modeleaza functiile pe care un sistem trebuie sa le indeplineasca.

Activity diagram: prezinta actiunile necesare pentru implinirea unui proces.

Sequence diagram: modeleaza modul in care diverse obiecte trebuie sa ineractioneze pentr a implini o anumita actiune.

State machine diagram: prezinta starile prin care va trece un obiect pentru ca un proces sa se implineasca.