

2. (UML) Unified Modeling Language este un limbaj standard pentru descrierea de modele și specificații pentru software. Acesta oferă o reprezentare grafică a structurii și comportamentului programului descris.

3. În dezvoltarea software, modelele UML sunt utilizate pentru a reprezenta diferite aspecte ale aplicației și a asigura o dezvoltare clară și bine structurată.

5. În [1] sunt prezentate 2 tipuri principale de diagrame și anume Structure Diagrams și Behavior Diagrams.

Asemănări: Ambele diagrame au rolul de a oferi utilizatorului o reprezentare grafică în acord cu programul pentru care sunt proiectate pentru a facilita înțelegerea mai bună a acestuia.

Deosebiri: În timp ce Structure Diagram prezintă aspect static precum relațiile componente, obiecte etc, Behavior Diagram nu oferă înțelegere despre ce se întâmplă în sistem și despre cum componentele interacționează între ele în sistem sau cu utilizatorul.

6. Așa cum scrie în [1] la capitol 2.4 avem de a face cu use case diagram, class diagram (care include și object diagram), state machine diagram, sequence diagram, and activity diagram.

Use case diagram reprezintă interacțiunea dintre sistem și "actor"/utilizator și descrie ce face sistemul în funcție de cum este utilizat.

Class diagram este folosit pentru modelarea conceptuală a structurii programului la nivel de clase.

State machine diagram descrie răspunsul componentelor la evenimente în funcție de starea în care se află pentru ca cititorul să vadă direcțiile în care programul se poate duce în funcție de starea entităților sale.

Sequence diagram arată operațiile care au loc în sistem în ordine cronologică pentru a înțelege drumul pe care programul îl urmează de-a lungul executării operațiilor sale