Laborator 1 – IS

Pop David, gr. 7

2. UML este un limbaj de modelare generalizat standardizat în domeniul ingineriei

software. Este utilizat pentru a specifica, a vizualiza, a construi și a documenta sisteme de software.

3. Modelele sunt reprezentari abstracte ale realitatii utilizate pentru a intelege, analiza si comunica concepte sau sisteme complexe. In domeniul ingineriei software, modelele sunt utilizate pentru a descrie structura, comportamentul si interactiunea componentelor unui sistem software.

6. Class diagram - Instrumentul de modelare UML iti permite sa modelezi structura sistemului prin modelarea claselor sale, a atributelor si operatiunilor acestora intr-o diagrama de clase UML. Diagrama de clase UML este un plan al claselor (la nivel de cod) necesare pentru a construi un sistem software. Programatorii implementeaza un sistem software cu ajutorul atat al diagramei de clase, cat si al specificatiei claselor.

Use case diagram - Capteaza cerintele functionale cu ajutorul instrumentului de diagrama de cazuri de utilizare UML. Fiecare caz de utilizare dintr-o diagrama de cazuri de utilizare reprezinta un obiectiv de business la nivel inalt, care produce un rezultat masurabil cu valoare pentru business. Actorii (UML) sunt conectati la cazurile de utilizare pentru a reprezenta rolurile care interactioneaza cu functiile.

Activity diagram - Foloseste diagrama de activitate UML, o diagrama bazata pe fluxuri, pentru a modela fluxul de control. Partitioneaza actiunile in functie de tipul de participant implicat.

Sequence diagram - Vizualizeaza interactiunile dintre utilizatori, sisteme si subsisteme in timp, prin transmiterea de mesaje intre obiecte sau roluri. Daca diagrama de clase reprezinta scheletul claselor, afisand atributele si metodele acestora, diagrama de secventa UML completeaza clasele prin reprezentarea logicii de programare ce urmeaza sa fie implementata in corpul metodelor.

State machine diagram - Diagrama masinii de stari este un model de proiectare esential pentru sistemele bazate pe evenimente. O masina de stari bine proiectata reprezinta cu exactitate starile esentiale ale obiectelor, precum si declansatorii schimbarilor de stare, facilitand astfel dezvoltarea unei masini de stari fara erori.

5. Structure diagrams si Behavior diagrams.

Asemanari

Fac parte din UML – Ambele categorii sunt tipuri de diagrame utilizate in Unified Modeling Language (UML) pentru modelarea sistemelor software.

Ajuta la intelegerea sistemului – Ambele tipuri de diagrame sunt folosite pentru a reprezenta diferite aspecte ale unui sistem software, facilitand proiectarea, dezvoltarea si documentarea acestuia.

Pot fi utilizate impreuna – Intr-un proiect software, atat diagramele de structura, cat si cele de comportament sunt folosite complementar pentru a oferi o viziune completa asupra sistemului.

Deosebiri

Structure diagrams - Reprezinta arhitectura si organizarea statica a unui sistem.

Behavior diagrams - Modeleaza dinamica si comportamentul sistemului in timp.

Structure diagrams - Modelare statica

Behavior diagrams - Modelare dinamica

Structure diagrams - Se concentreaza pe componentele sistemului si relatiile dintre ele.

Behavior diagrams - Se concentreaza pe interactiunile dintre componente si fluxul de executie.

