

Lab1

1) Ce este UML si pentru ce este utilizat?

-UML= Unified Modeling Language, dezvoltat pentru integrarea diagramelor; create pentru a ajuta sistemul si inginerii software pentru specificare, vizualizare, construire si documentatie sistemul software, precum si pentru business modeling si a altor sisteme non-software;

-utilizeaza in mare parte notatii grafice pentru a exprima proiectarea proiectelor soft

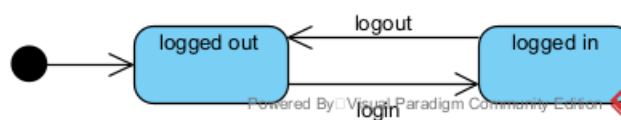
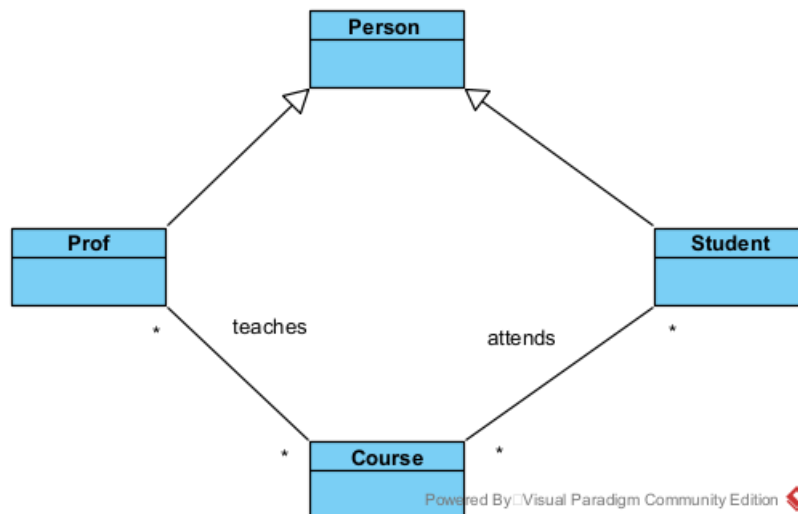
-utilizat pentru modeling language, object oriented modeling

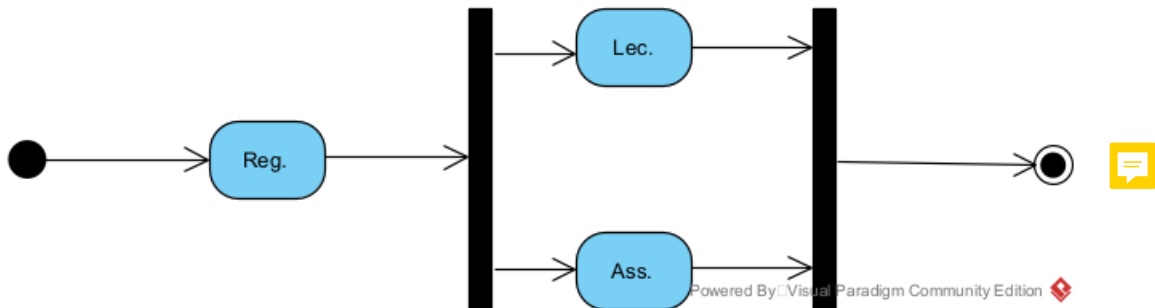
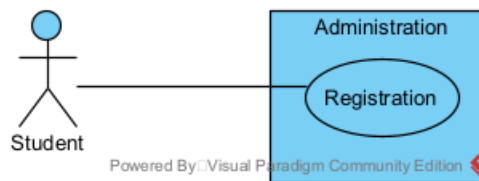
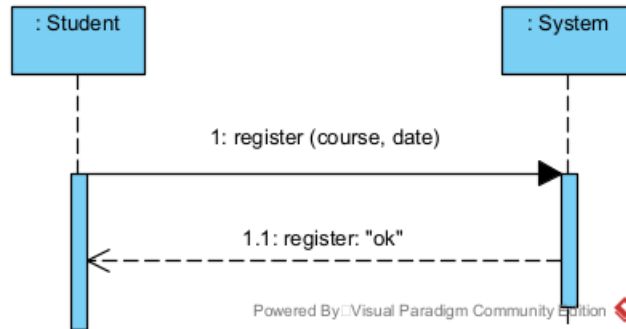
2) Ce sunt modelele si care este utilitatea lor?

-modelele descriu sistemele eficient si elegant;

-sunt create din componente care sunt legate cu altele si sunt influentate in modul in care

3) Familiarizarea cu mediul de lucru mai sus amintit prin crearea anumitor diagrame din sectiunile 2.3.1 s, i 2.3.2 din [1].





4) Asemanari:

- toate sunt diagrame UML
- reprezentarea vizuala ale anumitor aspecte de sistem pentru facilitarea intelegerii si comunicarii intre membrii echipei
- utilizata in dezvoltare software
- permite abstractia si focalizarea pe anumite aspecte ale sistemului

Deosebiri:

- fiecare diagrama se focalizeaza pe aspecte diferite ale sistemului (structura, comportament,interactiune)

- sunt utilizate in faze diferite ale dezvoltarii software; sunt utilizate in scopuri diferite (analiza, proiectare, implementare)

- elemente vizuale si modul in care sunt utilizate variaza de la diagrama la diagrama

5)Caracteristici

-Class diagram:

- include clase, interfete, asocieri si mosteniri

- arata relatiile dintre clase (Ex: asocieri si dependente)

- poate include detalii despre attribute si metode

- utilizata pt a define arhitectura de baza a unui sistem

-Use Case diagram:

- identifica functionalitatile sistemului din perspectiva utilizatorilor

- include cazuri de utilizare si actori

- evidentiaza relatii dintre actori

- utila pt a intelege cerintele functionale ale unui sist

-Activity diagram:

- reprezinta fluxuri de activitati si decizii dintr un proces

- include noduri de activitate, tranzitii, decizii si ramificari

- poate modela si concurenta in cadrul proceselor

- adekvata pt a arata logica operationala si fluxuri de lucru

-Sequence diagram:

- arata ordinea mesajelor schimbate intre obiecte pt a realiza o functionalitate

- include obiecte, mesaje, life line-uri, si puncta de sincronizare

- evidentiaza interactiunile bazate pe timp intre obiecte

- utila pt a analiza cerintele de detaliu ale sistemului si comportamentul acestuia

-State machine diagram:

- reprezinta starile unui obiect si tranzitiile intre aceste stari

- include stari, evenimente, tranzitii si actiuni
- poate modela comportamentul dinamic al sistemelor complexe
- adevata pt obiecte cu un comportament complex, care raspunde la evenimente sau conditii externe

6)

