Ministerul Educaţiei, Culturii și Cercetării al Republicii Moldova

Universitatea Tehnică a Moldovei

Departamentul Informatică și Ingineria Sistemelor

**RAPORT**

Lucrare de laborator Nr.2

Disciplina: APSI

Tema: Analiza sistemului în baza metodologiei APOO şi elaborarea modelelor prin diagramele cazurilor de utilizare.

A efectuat: st.gr.

Ulmanu Cristian IA-182,

A verificat : lect.univ.   
 Sava Nina  
Melnic Radu

Chișinău 2020

**Scopul**: Studierea noțiunilor de actor, caz de utilizare, nota, pachet și relațiile între entități UML  
 **Sarcina:** de realizat 4 diagrame use-case pentru sistemul informațional ales

**Sistemul ales este: sistemul de operare.**

**Considerații teoretice**

**Actor**

Un actor modelează un tip de rol jucat de o entitate care interacționează cu subiectul (de exemplu, prin schimbul de semnale și date), dar care este extern subiectului (adică, în sensul că o instanță a unui actor nu face parte din instanța subiectului său corespunzător). Actorii pot reprezenta roluri jucate de utilizatori umani, hardware extern sau alte subiecte. Rețineți că un actor nu reprezintă neapărat o entitate fizică specifică, ci doar o fațetă anume (adică „rol”) a unei entități care este relevantă pentru specificarea cazurilor sale de utilizare asociate. Astfel, o singură instanță fizică poate juca rolul mai multor actori și, dimpotrivă, un anumit actor poate fi jucat de mai multe instanțe diferite.

Deoarece un actor este extern subiectului, acesta este de obicei definit în același clasificator sau pachet care încorporează clasificatorul subiectului

**Asociere**

O asociere specifică o relație semantică care poate apărea între instanțe tipizate. Are cel puțin două capete reprezentate de proprietăți, fiecare dintre ele fiind conectat la tipul capătului. Mai mult de un capăt al asociației poate avea același tip.

O proprietate finală a unei asociații care este deținută de o clasă finală sau care este un scop navigabil deținut de asociație indică faptul că asociația este navigabilă din capetele opuse; în caz contrar, asocierea nu este navigabilă din capetele opuse.

**Dependenţă**

O dependență este o relație care înseamnă că un singur sau un set de elemente de model necesită alte elemente de model pentru specificarea sau implementarea lor. Aceasta înseamnă că semantica completă a elementelor dependente este fie semantică, fie structural dependentă de definiția elementului (elementelor) furnizor

**Extinde**

Această relație specifică faptul că comportamentul unui caz de utilizare poate fi extins prin comportamentul unui alt caz de utilizare (de obicei suplimentar). Extensia are loc la unul sau mai multe puncte de extensie specifice definite în cazul de utilizare extins. Rețineți totuși că cazul de utilizare extins este definit independent de cazul de utilizare extins și este semnificativ independent de cazul de utilizare extins. Pe de altă parte, cazul de utilizare extins definește de obicei un comportament care nu poate fi neapărat semnificativ de la sine. În schimb, cazul de utilizare extins definește un set de creșteri modulare ale comportamentului care măresc execuția cazului de utilizare extins în condiții specifice.

Rețineți că același caz de utilizare extins poate extinde mai multe cazuri de utilizare. Mai mult, un caz de utilizare extins poate fi el însuși extins.

**Generalizare**

O generalizare este o relație taxonomică între un clasificator mai general și un clasificator mai specific. Fiecare instanță a clasificatorului specific este, de asemenea, o instanță indirectă a clasificatorului general. Astfel, clasificatorul specific moștenește trăsăturile clasificatorului mai general.

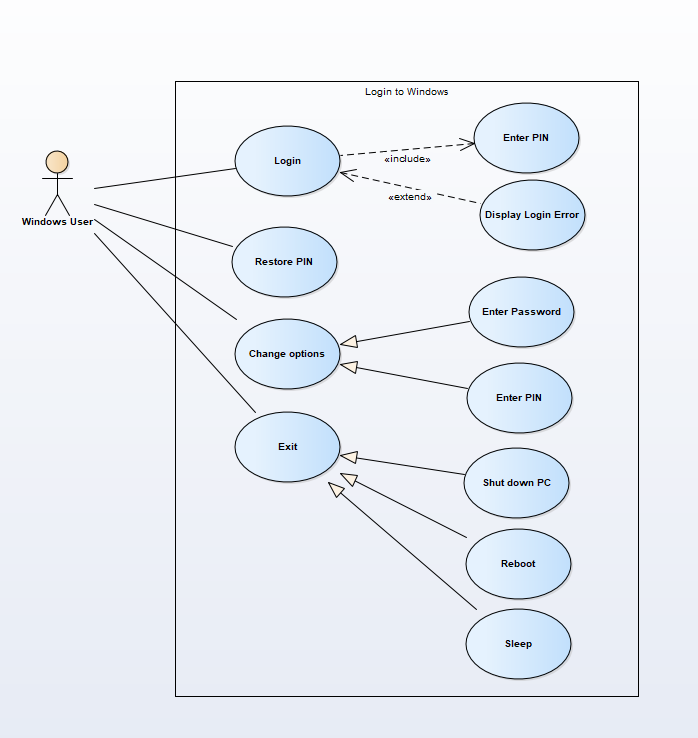
**Include**

Include este o relație direcționată între două cazuri de utilizare, ceea ce implică faptul că comportamentul cazului de utilizare inclus este inserat în comportamentul cazului de utilizare inclus. Este, de asemenea, un fel de NamedElement, astfel încât să poată avea un nume în contextul propriului caz de utilizare. Cazul de utilizare inclus poate depinde numai de rezultatul (valoarea) cazului de utilizare inclus. Această valoare este obținută ca urmare a executării cazului de utilizare inclus.

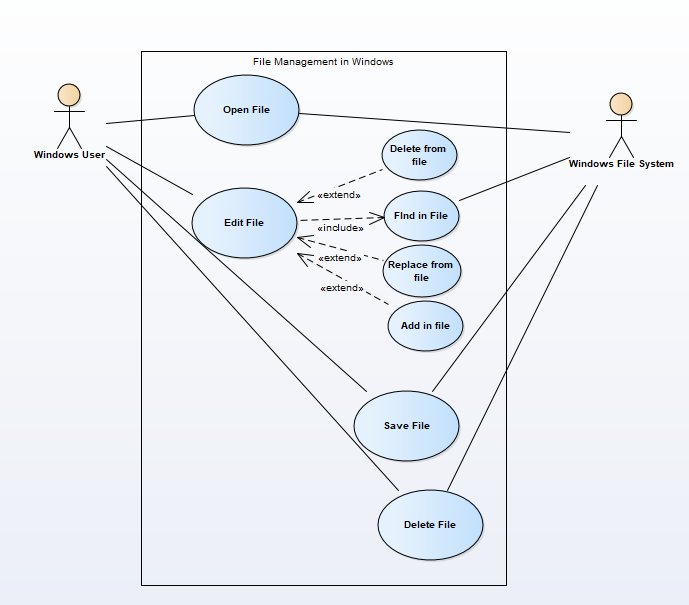
Rețineți că cazul de utilizare inclus nu este opțional și este întotdeauna necesar pentru ca cazul de utilizare inclus să se execute corect.

**Diagramele use case pentru sistemul de operare**

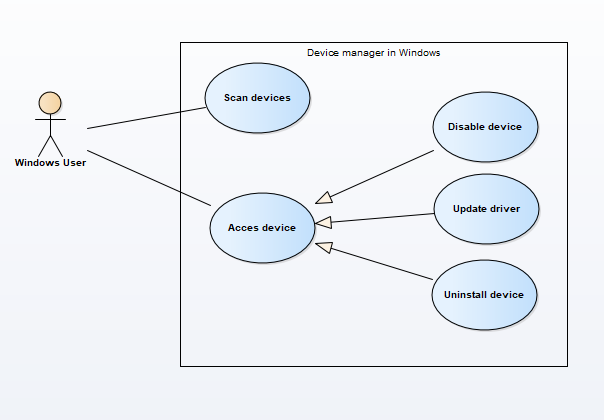
**Diagrama de logare în Windows**



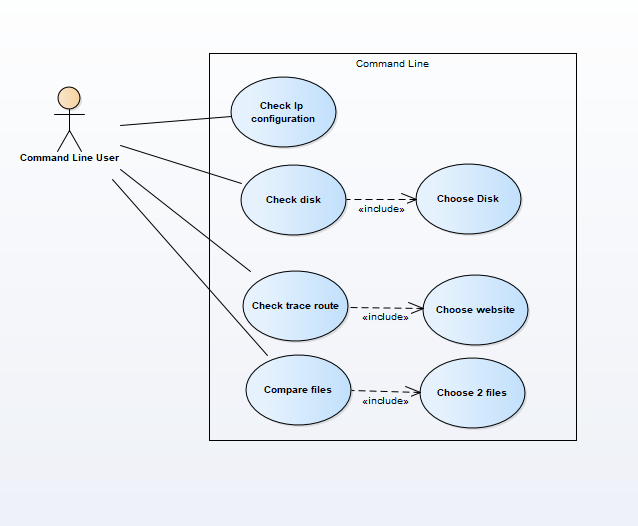
**Diagrama de creare și management a fișierelor**



**Managementul dispozitivelor în Windows**



**Operații în linia de comandă a Windows**



**Bibliografie**

1. **Melnic R., Sava N.** Indrumar metodic “Analiza si modelarea sistemelor informationale”.
2. **Моделирование бизнес процессов|CASE средства|Rational Rose**, [Электронный ресурс].-Режим доступа: <http://www.kpms.ru/Automatization/Rational_Rose.htm>
3. https://www.visual-paradigm.com/VPGallery/diagrams/UseCase.html
4. <https://sparxsystems.com/resources/user-guides/15.2/index.html#fundamentals>
5. https://www.youtube.com/watch?v=zid-MVo7M-E